International Telecommunication Union

ITU-T FG-AI4H Deliverable

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

15 September 2023

PRE-PUBLISHED VERSION

DEL10.14

FG-AI4H Topic Description Document for the Topic Group on Symptom assessment (TG-Symptom)

ITU-T

International Telecommunication Union

Summary

This topic description document (TDD) specifies a standardized benchmarking for AI-based symptom assessment. It covers all scientific, technical and administrative aspects relevant for setting up this benchmarking.

Keywords

Artificial intelligence; benchmarking; health; topic groups; overview; ethics; regulations; data quality; data audit; clinical relevance; topic description; symptom assessment

Change Log

This document contains Version 1 of the Deliverable DEL10.14 on "*FG-AI4H Topic Description Document for the Topic Group on Symptom assessment (TG-Symptom)*" approved on 15 September 2023 via the online approval process for the ITU-T Focus Group on AI for Health (FG-AI4H).

Editor:

Henry Hoffmann
TG-Symptom
Ada Health GmbH
Germany
Tel: +49 177 6612889
Email: henry.hoffmann@ada.com

Martin Cansdale
TG-Symptom
Healthily
UK
Email: martin@livehealthily.com

Contributors:

Andreas Kühn
(Formerly Ada Health GmbH)
Germany

Jonathon Carr-Brown
Healthily
UK
Tel: +44 7900 271580
E-mail: jcb@livehealthily.com

Matteo Berlucchi
Healthily
UK
Tel: +44 7867 788348
E-mail: matteo@livehealthily.com

Jason Maude
Isabel Healthcare
UK
Tel: +44 1428 644886
E-mail: jason.maude@isabelhealthcare.com

Shubhanan Upadhyay
Ada Health GmbH
Germany
Tel: +44 7737 826528
E-mail: shubs.upadhyay@ada.com

Yanwu Xu
Artificial Intelligence Innovation Business, Baidu
China
Tel: +86 13918541815
E-mail: xuyanwu@baidu.com

Ria Vaidya
(Formerly Ada Health GmbH)
Germany

Isabel Glusman
(Formerly Ada Health GmbH)
Germany

Saurabh Johri
Babylon Health
UK
Tel: +44 (0) 7790 601 032
E-mail: saurabh.johri@babylonhealth.com

Nathalie Bradley-Schmieg
(Formerly Babylon Health)
UK

Piotr Orzechowski
Infermedica
Poland
Tel: +48 693 861 163
E-mail: piotr.orzechowski@infermedica.com

Irv Loh, MD
Infermedica
USA
Tel: +1 (805) 559-6107
E-mail: irv.loh@infermedica.com

Jakub Winter
Infermedica
Poland
Tel: +48 509 546 836
E-mail: jakub.winter@infermedica.com

Ally Salim Jr
Inspired Ideas
Tanzania
Tel: +255 (0) 766439764
E-mail: ally@inspiredideas.io

Megan Allen
Inspired Ideas
Tanzania
Tel: +255 (0) 626608190
E-mail: megan@inspiredideas.io

Anastacia Simonchik
Visiba Group AB
Sweden
Tel: +46 735885399
E-mail: anastacia.simonchik@visibacare.com

Sarika Jain
(Formerly Ada Health GmbH)
Germany

Yura Perov
Babylon Health
UK
E-mail: yura.perov@gmail.com

Tom Neumark
University of Oslo
Norway
E-mail: thomas.neumark@sum.uio.no

Rex Cooper
(Formerly Healthily)
UK

Martina Fischer
(Formerly Ada Health GmbH)
Germany

Lina Elizabeth Porras Santana
1DOC3
Colombia
E-mail: linaporras@1doc3.com

Juan Sebastián Beleño
1DOC3
Colombia
E-mail: jbeleno@1doc3.com

María Fernanda González Alvarez
1DOC3
Mexico
E-mail: mgonzalez@1doc3.com

Adam Baker
Babylon Health
UK
E-mail: adam.baker@babylonhealth.com

Xingxing Cao
Baidu
China
E-mail: caoxingxing@baidu.com

Clemens Schöll
(Formerly Ada Health GmbH)
Germany

Audrey Menezes
(Formerly Healthily)
UK

Francisco Cheda
Barkibu
Spain
E-mail: fran@barkibu.com

Ernesto Hernández
Barkibu
E-mail: ernesto@barkibu.com

Saddif Ahmed
Flo Health
London
E-mail: s_ahmed@flo.health

Anna Klepchukova
Flo Health
London/Minsk
E-mail: a_klepchukova@flo.health

Michal Tzuchman Katz
Kahun
Israel, Tel Aviv
E-mail: michal@kahun.com

András Meczner
Healthily
UK
E-mail: andras@livehealthily.com

Aleem Qureshi
(Formerly Healthily)
UK

Marta Lemanczyk
Hasso Plattner Institute
Germany
E-mail: Marta.Lemanczyk@hpi.de

Carolin Prabhu Riksrevisjonen Norway	E-mail:	carolin.prabhu@riksrevisjonen.no
Eva Weicken Fraunhofer HHI Germany	E-mail:	eva.weicken@hhi.fraunhofer.de
Milan Jovanovic Ada Health GmbH Germany	E-mail:	milan.jovanovic@ada.com
Mateusz Głód Infermedica Poland	E-mail:	mateusz.glod@infermedica.com
Dejan Hajdukovic Croatia	E-mail:	mailto:hajdukdejan@hotmail.com
Rohit Malpani Consultant, WHO, FG AI4H WG-Ethics Switzerland, Geneva	E-mail:	malpanir@who.int

CONTENTS

Page

1 Introduction 1

2 About the FG-AI4H topic group on AI-based symptom assessment 1
- 2.1 Documentation 2
- 2.2 Status of this topic group 2
- 2.3 Topic Group participation 3

3 Topic description 4
- 3.1 Definition of the AI task 4
- 3.2 Current gold standard 4
- 3.3 Relevance and impact of an AI solution 4
- 3.4 Existing AI solutions 5
  - 3.4.1 Topic Group member Systems for AI-based Symptom Assessment 5
  - 3.4.2 Other systems for AI-based symptom assessment 8
  - 3.4.3 Input data 9
  - 3.4.4 Output data 11
  - 3.4.5 Scope dimensions 15
  - 3.4.6 Additional relevant dimensions 16
  - 3.4.7 Robustness of systems for AI based symptom assessment 17

4 Ethical considerations 18
- 4.1 Protect autonomy 18
  - 4.1.1 Effect on decision-making in health 18
  - 4.1.2 Protection of privacy of personal health information 19
- 4.2 Promote human well-being, human safety and the public interest 19
  - 4.2.1 The ethical implications of introducing benchmarking 19
- 4.3 Ensure transparency, explainability and intelligibility 20
- 4.4 Foster responsibility and accountability 20
- 4.5 Ensure inclusiveness and equity 21
  - 4.5.1 Wider potential societal effects of AISAs 22
- 4.6 Promote artificial intelligence that is responsive and sustainable 22

5 Existing work on benchmarking 22
- 5.1 Self-Assessment 22
  - 5.1.1 Publications on benchmarking systems 22
  - 5.1.2 Benchmarking publications outside science 28
  - 5.1.3 Benchmarking by AI developers 29
  - 5.1.4 Relevant existing benchmarking frameworks 29
  - 5.1.5 Scores & metrics 31
  - 5.1.6 Metrics for symptom assessment 37

Page
5.1.7 Performance and accuracy ... 38
5.1.8 Putting it all together for clinicians ... 46
5.1.9 Additional clinical considerations and limitations ... 48
5.1.10 Conclusion ... 49

6 Benchmarking by the topic group ... 49
6.1 Benchmarking self-assessment systems ... 49
6.1.1 Benchmarking version MMVB 1.0 ... 50
6.1.2 Benchmarking version MMVB 2.0 - 2.2 ... 65
6.1.3 MMVB 3.0 - 3.1 ... 91

7 Overall discussion of the benchmarking so far ... 114
7.1 Overview of work done ... 114
7.2 Learnings ... 115
7.3 Next steps ... 117
7.4 A comment on the LLMs in context of symptom assessment ... 118
7.4.1 The possible place of LLMs in symptom assessment systems ... 118
7.4.2 Implications for benchmarking ... 121

8 Regulatory considerations ... 121
8.1 Existing applicable regulatory frameworks ... 122
8.2 Regulatory features to be reported by benchmarking participants ... 123
8.3 Regulatory requirements for the benchmarking systems ... 124
8.4 Regulatory approach for the topic group ... 125

9 References ... 125

Annex A Glossary ... 129

Annex B Declaration of conflict of interests ... 132

Annex C Topic Group status updates for the focus group meetings ... 137

Annex D MMVB 3.x case encoding tool manual ... 173

Annex E MMVB 3.x case annotation guideline ... 186

List of Tables

Page
Table 1 – Topic Group output documents ... 2
Table 2 – Symptom assessment systems inside the topic group ... 5
Table 3 – Symptom assessment systems outside the topic group ... 8

Page

Table 4 – Overview symptom assessment system inputs 9
Table 5 – Overview symptom assessment system outputs 11
Table 6 – Manchester Triage System levels 11
Table 7 - Ground truth approaches with their problems 34
Table 8 – Overview patient case metrics 40
Table 9 – Benchmarking iterations 50
Table 10 – MMVB 1.0 input data format 56
Table 11 – MMVB 1.0 API output encoding example 57
Table 12 – MMVB 1.0 AI output label encoding 58
Table 13 – An example of a MMVB 1.0 case set with a single case. 58
Table 14 – Case example for the London Model 61
Table 15 – MMVB 2.2 input data format 80
Table 16 – MMVB 2.2 AI output structure 82
Table 17 – MMVB 2.2 case with labels included. 82
Table 18 – MMVB 2.2 overall case set structure 83
Table 19 – MMVB 3.x annotation tool backend API interface. 96
Table 20 – MMVB 3.x AI API interface. 98
Table 21 – Example of the case without FHIR encoding – similar to the MMVB 2.2 case format. 101
Table 22 – Top-level structure of a FHIR encoded benchmarking case. 102
Table 23 – FHIR encoded input case 103
Table 24 – MMVB 3.x AI output structure for the FHIR and the non-FHIR benchmarking API endpoints 105
Table 25 – Health-check endpoint API responses 106
Table 26 – MMVB 3.x case set structure with label/annotation example 106
Table 27 – MMVB 3.x case with labels included 107
Table 28 – The list of metrics migrated to the OCI system together with the descriptions used 108
Table 29 – Case #1 of the cases prepared after workshop #3 109
Table 30 – Case #33 of the MMVB 3.x case corpus 111
Table 31 – Toy-AI benchmarking results for the “Official FG AI4H Meeting I Benchmarking test data set“. Please note that the toy-AIs have nothing to do with the company’s production AIs. 113
Table 32 – IMDRF AI system risk classification scheme 122
Table 33 – Candidates for AI regulatory metadata fields 124

List of Figures

Page

Figure 1 – Example “Model Facts” label for sepsis machine learning model from Sendak et al, 2020. (Nature) 47
Figure 2 – "London Model" used for sampling cases for MMVB 1.0 51
Figure 3 – MMVB 1.0 High-level architecture 53
Figure 4 – MMVB 1.0 case generation UI 55
Figure 5 – MMVB 1.0 screen for running a benchmarking session 55
Figure 6 – MMVB 1.0 result screen 55
Figure 7 – Abdominal Pain symptom with attributes inside the Berlin Model 65
Figure 8 – Factors with state details inside the Berlin Model 66
Figure 9 – Refined factor distributions for ectopic pregnancy inside the Berlin Model 66
Figure 10 – MMVB 2.2 High-level architecture 67
Figure 11 – MMVB 2.2 ai-implementations API 68
Figure 12 – MMVB 2.2 cases and case sets API 69
Figure 13 – MMVB 2.2 benchmarking-sessions API 70
Figure 14 – MMVB 2.2 metrics API 70
Figure 15 – 2.2 Version of the Benchmarking start page 72
Figure 16 – The AI implementations list now featuring the ability of adding new AIs and editing existing ones. 73
Figure 17 – MMVB 2.2 case sets overview page 74
Figure 18 – MMVB 2.2 case set creation page 75
Figure 19 – MMVB 2.2 benchmarking sessions overview page 76
Figure 20 – MMVB 2.2 benchmarking session creation page 77
Figure 21 – MMVB 2.2 benchmarking session runner 78
Figure 22 – MMVB 2.2 benchmarking result page 79
Figure 23 – MMVB 2.2 General input case structure 80
Figure 24 – MMVB 2.2 case set raw viewer 85
Figure 25 – MMVB 2.2 case set statistics view 87
Figure 26 – MMVB 2.2 example of a case defined by a doctor using the case annotation tool 88
Figure 27 – MMVB 2.2 benchmarking results for test set with 100 cases sampled from the Berlin model 90
Figure 32 – Some of the case vignettes created by the doctors after workshop #3 149
Figure 33 – Attributes of symptom "abdominal pain" collected from the workshop #3 case vignettes 150
Figure 34 – Example of workshop #3 symptoms phrases mapped to SNOMED CT (ignoring attributes) 151
Figure 35 – Experimental first simple SNOMED CT based case creation tool 152

Page

Figure 36 – Case symptoms separated by category 155

Figure 37 – SNOMED search results for "headache" (left side) and the ancestors and children for the selected "Headache (finding)" concept. 156

Figure 38 – Editor that describes attributes severity, clinical course and finding site of an abdominal pain finding. 159

Figure 39 – SNOMED concept browser with the new feature showing how often concepts have been used in cases and if they are appropriate for use in case vignettes. 160

Figure 40 – Screenshot of the first benchmarking results in the audit benchmarking system 163

Figure 41 – Main roadmap items for the remaining time of the topic group. 164

Figure D.1 – Case Encoding Tool will ultimately be used by clinicians with the goal of creating new clinical case vignettes to be used for the independent benchmarking of symptom assessment tools globally. 174

Figure D.2 – The "Comment" box is there to allow a user (case annotator) to report any issues observed during the case creation, e.g., issues with using the Case Encoding Tool, describing the case etc., while the "Case description" box serves the purpose of storing all evidence relevant for the case as reference while encoding the case. 174

Figure D.3 – The case encoding process starts already at this stage by entering the clinical vignette title, the disease the case is describing, the triage level for the expected disease and the age and sex of the virtual patient. 175

Figure D.4 – At this point, the evidence stored in the "Case description" box is being encoded into corresponding evidence type fields. 176

Figure D.5 – The encoding of evidence starts by clicking on the "+" button. 177

Figure D.6 – Snomed Concept Browser allows a user to quickly find evidence of interest, e.g., fever, abdominal pain, fatigue etc. 177

Figure D.7 – SNOMED is a systematically organized computer-processable collection of medical terms used in clinical documentation and reporting. 179

Figure D.8 – User should search for the term that describes the evidence from a case most precisely. 180

Figure D.9 – In case that the evidence of interest is not precisely defined by the selected search result, a user should try to look for it by reviewing Ancestor and Children concepts related to it. . 180

Figure D.10 – Once a user finds an evidence of interest this part of the process should be finalised by clicking on the "Add" button. 181

Figure D.11 – To define time since onset, severity, clinical course and/or finding site of an evidence of interest a user clicks the blue pen icon. 182

Figure D.12 – Only those fields that will help to define an evidence of interest as precisely as possible should be completed. 183

Figure D.13 – User defines attributes by clicking on the Select State box located on the right hand side of a respective Attribute field and by selecting the time since onset unit and time since onset value in case of the "time since onset" attribute. 183

Figure D.14 – Another part of the encoding process could now be finalised by clicking the "Save" button. 184

Page

Figure D.15 – A user is now able to see the changes made during the previous step of the process.. 184

Figure D.16 – Once all the evidence from the "Case description" box is encoded a user is ready to finalise the case creation process by clicking the "Submit new case" button. 185

Figure E.1 – Example of a case list showing pre-created annotation task cases..... 187

ITU-T FG-AI4H Deliverable 10.14

FG-AI4H Topic Description Document for the Topic Group on Symptom assessment (TG-Symptom)

1 Introduction

This document describes the work towards the specification of a standardized benchmarking for AI-based symptom assessment systems. It serves as deliverable No. DEL10.14 of the ITU/WHO Focus Group on AI for Health (FG-AI4H).

As the COVID-19 pandemic recently underlined, providing everyone with the health care they need is still a challenge. The 2017 Global Monitoring Report by the WHO and the World Bank reported that half of the world's population lacks access to basic essential health services [1].

One reason for the limited access to proper health care, reduced doctor time and worsen patient journeys to a correct diagnosis and proper treatment is the shortage of health workers. In 2013 the World Health Organization estimates the shortage of global health workers to increase from 7.2 million in 2013 to 12.9 million by 2035 [2]. While more recent data showed improvements, the COVID-19 pandemic is expected to have increased the shortage [3]. This shortage is driven by several factors including growing population, increasing life expectancy, higher health demands and an unequal global distribution of work force in healthcare.

In recent years, one promising approach to meet the challenging shortage of doctors has been the introduction of AI-based symptom assessment applications that have become widely available. These systems, also called “symptom-checkers”, allow their users to enter presenting complaints they seek advice for. The systems then follow-up with a conversation collecting further evidence on other symptoms the user might have experienced to then provide advice on relevant next steps ranging from self-care, over see a pharmacy to seek emergency care, diseases that might have caused the symptoms and explanations on how the symptoms and these suggestions are related.

By navigating users to the right care at the right time such systems help using the resources of the health systems more efficient. On the doctor’s side such systems help to save time by allowing for an automated collection of relevant information before seeing the doctor and to reduce the risk of misdiagnosis.

While systems for AI-based symptom assessment have great potential to improve health care, the lack of consistent standardisation makes it difficult for organizations like the WHO, governments and other key players to adopt such applications as part of their policies to address global health challenges.

The specification of a standardized benchmarking for AI based symptom assessment applications in this document as part of the ITU/WHO AI4H Focus Group will therefore be an important step towards closing this gap.

2 About the FG-AI4H topic group on AI-based symptom assessment

The introduction highlights the potential of a standardized benchmarking of AI systems for AI-based symptom assessment to help solving important health issues and provide decision-makers with the necessary insight to successfully address these challenges. To develop this benchmarking framework, FG-AI4H decided to create the TG-Symptom at the meeting C in Lausanne, Switzerland, 22-25 January 2019. It was based on the "symptom checkers" use case, which was

accepted at the November 2018 meeting B in New York building on proposals by Ada Health GmbH:

- A-020: Towards a potential AI4H use case "diagnostic self-assessment apps"
- B-021: Proposal: Standardized benchmarking of diagnostic self-assessment apps
- C-019: Status report on the "Evaluating the accuracy of 'symptom checker' applications" use case and on a similar initiative by Healthily (formerly Your.MD):
- C-025: Clinical evaluation of AI triage and risk awareness in primary care setting

The focus group assigns a *topic driver* to each topic group who coordinates the collaboration of all topic group members on this document. During meeting C in Lausanne, Switzerland, 22-25 January 2019, Henry Hoffmann from Ada Health GmbH, Berlin, Germany was nominated as topic driver for the TG-Symptom. Between meeting O and meeting P Martin Cansdale from Healthily joined as second topic driver.

2.1 Documentation

This document is the Topic Description Document (TDD) for the topic group on AI-based symptom assessment. It introduces the health topic including the AI task, outlines its relevance and the potential impact that the benchmarking will have on the health system and patient outcome and provides an overview of the existing AI solutions for symptom assessment. It describes the existing approaches for assessing the quality of such systems and provides the details that are relevant for setting up standardized benchmarking. It specifies the actual benchmarking methods at a level of detail that includes technological and operational implementation. Finally, it summarizes the results of the topic group's benchmarking initiative and benchmarking runs. In addition, the TDD addresses ethical and regulatory aspects.

The TDD was developed cooperatively by the members of the topic group and updated versions have been submitted and presented at all FG-AI4H meeting. Table 1 shows the documents submitted for each meeting.

Table 1 – Topic Group output documents

Number	Title
FGAI4H-x-021-A01	Latest update of the Topic Description Document of the TG-Symptom
FGAI4H-x-021-A02	Latest update of the Call for topic group Participation (CfTGP)
FGAI4H-x-021-A03	The presentation summarizing the latest update of the Topic Description Document of the TG-Symptom

2.2 Status of this topic group

At the end of its live time as part of the focus group, the members of the topic group included 22 companies and 9 independent contributors. Of which in 2023 3 companies and 2 independent contributors have been active.

The members contributed with changing engagement based on their resources and priorities. With the start of the pandemic many companies in the topic group had to focus on contributing initiatives for dealing with the health crisis and could not participate in the same way.

After meeting D the topic group identified the potential need for introducing two distinct sub-topic-groups for "self-assessment" and "clinical symptom assessment". The first sub-topic was proposed to address symptom-checker apps used by non-doctors while the second group was supposed to focus on symptom-based diagnostic decision support systems for doctors. Since both topics would build on the same foundation, the topic group decided to postpone a separation until the “self-assessment” topic would reach a point there the common parts reached the necessary maturity. This document only describes the self-assessment use case.

With the focus group reaching the end of its lifetime, the TG-symptoms aims to transition into the new WHO AI for health global initiative to continue the work there since the advent of large language models in mainstream public use underlines the need to have reliable benchmarking for symptom assessment system which needs to be continued to be developed and extended to take new systems into account.

During the lifetime of the focus group this document contained in this section the updates for each focus group meeting. For this final submission the history updates have been moved to Annex C Topic Group status updates for the focus group meetings.

2.3 Topic Group participation

The participation in both, the focus group on AI for Health and in the topic group was open to any individual. For this topic group, the corresponding final version of the ‘Call for TG participation’ describing the topic group and how to join it can be found here:

- https://extranet.itu.int/sites/itu-t/focusgroups/ai4h/docs/FGAI4H-Q-021-A02.docx

Each topic group also had homepage ad the ITU collaboration site. The subpage for this topic group can be found here:

- https://www.itu.int/en/ITU-T/focusgroups/ai4h/Pages/symptom.aspx

For participation in this topic group, interested parties could also join the regular online meetings. All relevant administrative information about FG-AI4H - like upcoming meetings or document deadlines - have been announced via the general FG-AI4H mailing list fgai4h@lists.itu.int which could be joined by following the instructions found at https://itu.int/go/fgai4h/join.

In addition to the general FG-AI4H mailing list, the topic group also had an *individual mailing list:*

- fgai4htgsymptom@lists.itu.int

Regular FG-AI4H workshops and meetings proceed about every two months at changing locations around the globe and remotely. More information can be found on the official FG-AI4H website:

- https://itu.int/go/fgai4h

3 Topic description

This section contains a detailed description of AI-based symptom assessment and how they can help to solve a relevant ‘real-world’ problem.

3.1 Definition of the AI task

This section provides a detailed description of the symptom-assessment AI task. It does not cover the existing work on benchmarking of symptom assessments as this will be discussed in chapter 5. This section corresponds to DEL03 *“AI requirements specifications*,” which describes the functional, behavioural and operational aspects of an AI system.

The exact definition of Artificial Intelligence (AI) is controversial. In the context of this document, it refers to a field of computer science working on machine learning and knowledge-based technology that allows the user to *understand* complex (health related) problems and situations at or above human (doctor) level performance and providing corresponding insights (differential diagnosis, triage) or solutions (next step advice).

The available systems can be divided into consumer facing tools sometimes referred to as "symptom checkers" and professional tools for doctors sometimes described as "diagnostic decision support systems". In general, these systems allow users to state an initial health problem, usually medically termed as the presenting complaint (PC) or chief complaint (CC). Following the collection of PCs, the collection of additional symptoms is performed either proactively - driven by the application using an interactive questioning approach - or passively, allowing the user to enter additional symptoms. Finally, the applications provide an assessment containing different output components, which can include a general classification of severity (triage), possible differential diagnoses (DD) and advice on what to do next.

3.2 Current gold standard

The gold standard for correct differential diagnosis, next step advice and adequate treatment is the evaluation of a medical doctor who is an expert in the respective medical field, which is based on many years of university education and structured training in hospitals and/or in community settings. Depending on context, steps such as triage preceding diagnosis are responsibilities of other health workers. Decision making is often supported by clinical guidelines and protocols or by consulting literature, the internet or other experts.

In recent years, individuals have increasingly begun to use the internet to find advice. Recent publications show that one in four Britons use the web to search their symptoms instead of seeing a doctor [4]. Meanwhile, other studies show that internet self-searches are more likely to incorrectly suggest conditions that may cause inappropriate worry (e.g., cancers for innocuous symptoms).

3.3 Relevance and impact of an AI solution

Whilst the shortage of health workers in low- and middle-income countries (LMICs) is worse, in more developed countries health systems face challenges such as increased demand due to increased life expectancy. Additionally, available doctors have to spend considerable amounts of time on patients that do not always need to see a doctor. Up to 90% of people who seek help from primary care have only minor ailments and injuries [5]. The vast majority (>75%) attend primary

care because they lack an understanding of the risks they face or the knowledge to care for themselves. In the United Kingdom alone, there are 340 million consultations at the GP every year and the current system is being pushed to do more with fewer resources.

The challenge is to provide high-quality care with prompt and adequate treatment where necessary and to develop mechanisms to avoid overdiagnosis and focus the health system resources on the patients in need.

Very few high quality papers on AI-based symptom assessment exist and those that do are usually based on limited retrospective studies or use case vignettes instead of real cases. Therefore, there is a lack of scientific evidence available that assesses the impact of applying such technologies in a healthcare setting.

AI-powered symptom assessment applications have the potential to improve patient and health worker experience, deliver safer diagnoses, support health management and save health systems time and money. This could be by empowering people to navigate to the right care, at the right time and in the right place or by enhancing the care that healthcare professionals provide.

Reliable benchmarking of AI solutions will give stakeholders the numbers and metrics required for decision making, building trust and paving the way for wider adoption of AI-based symptom assessment. This wider adoption could potentially enable outcomes such as earlier diagnosis of conditions, more efficient care-navigation through the health systems and ultimately better health as it is currently pursued by UN's sustainable development goal (SDG) number 3 [6].

3.4 Existing AI solutions

This section presents the AI providers currently available and known to the topic group. The tables summarize the inputs and outputs relevant for benchmarking. They also present relevant details concerning the scope of the systems that will affect the definition of categories for benchmarking reports, metrics and scores.

3.4.1 Topic Group member Systems for AI-based Symptom Assessment

Table 2 provides an overview of the AI systems of the topic group members. The initial benchmarking started with the AI developers who joined the topic group and hence focussed on the benchmarking relevant technical dimensions found in this group before increasing the complexity to cover all other aspects.

Table 2 – Symptom assessment systems inside the topic group

Provider	System(s)	Input	Output	Scope/Comments
1DOC3	1DOC3 platform	• Age, sex • Free text • Complementary information about signs, symptoms and medication related to the main topic.	• Pre-clinical triage • Possible Causes – differentials.	• Worldwide • Spanish • More than 750 conditions • Web and App for iOS and Android

Ada Health GmbH	Ada Health App	• Age, sex, risk factors • Free text PC • Discrete answers to dialog questions for additional factors and symptoms including attribute details like intensity or time since onset.	• Pre-clinical triage / advice on next steps for the whole case • List of conditions most likely causing the presenting complaints • Per-condition advice level • (Roadblocks in case of immediate danger e.g., suicidal tendencies)	• English (US/UK), German, Spanish, Portuguese, French. Swahili, Romania • Top 1382 conditions • For smartphone users on Android/iOS (Ada Health App)
Ada Health GmbH	Ada Assess	• Age, sex, risk factors • Free text PC • Discrete answers to dialog questions for additional factors and symptoms including attribute details like intensity or time since onset.	• Pre-clinical triage / advice on next steps for the whole case • List of conditions most likely causing the presenting complaints • Per-condition advice level • (Roadblocks in case of immediate danger e.g., suicidal tendencies)	• Arabic, English (US, UK, CA), Dutch, French (FR, CA), German. Spanish, Portuguese (Brazilian), Italian • Top 1382 conditions • Embedded into partner websites (Ada Assess)
Babylon Health	Babylon App	• Age, sex, risk factors, country • Chatbot free text input and free text search (multiple inputs are allowed) • Answers to dialog questions for additional symptoms and risk factors including duration of symptoms, intensity	• Pre-clinical triage • Possible causes ("differentials") • Condition information • Recommendation of appropriate local services and products • Text information about treatments or next steps • Shortcuts in case of immediate danger	• Worldwide • English • 80% of medical conditions • For smartphone/web users • Android/iOS/Web
Baidu	Baidu's Clinical Decision Support System	• Age*, sex*, birthplace, occupation, residence, height, weight • Free text of PC*, CC*, Past Medical History, Family History, Allergic History, Menstrual History*, Marital and Reproductive History for female • Semi-structure text of medical exam report and test report * these details must be provided	• Pre-clinical triage • Diagnosis recommendation with explanation (structure or free text) • Next steps, such as medical exam, test • Treatment recommendation with explanation, such as drug, operations recommendation	• China • Chinese • General practice, 4000+ diagnoses • For Clinicians / Web users • CS SDK / BS SDK / API for HIT Companies integration • Web / mini program apps for Web users
Deepcare	Deepcare Symptom Checker			• Users: Doctor and Patient • Platforms: iOS android

				• Language: Vietnamese
Flo Health	Flo App	• Age, assumes female sex • Symptoms, risk factors, menstrual cycle history • Answers to discrete dialog questions	• Differential suggestions with explanation	• Worldwide - women only • English and 21 other languages • Women only • App (iOS and Android) and web based
Healthily	Healthily App	• Age, sex, medical risk factors • Chatbot free text input • User consultation output (report)	• Differentials for PC • Pre-clinical triage • Shortcuts in case of immediate danger • Condition information • Recommendation of appropriate local services and products • Medical factors	• Worldwide • English • >630 conditions • For smartphone users Android /iOS and web
Infermedica	Infermedica API, Symptomate	• Age, sex • Risk factors • Free text input of multiple symptoms • Region/Travel history • Answers to discrete dialog questions • Lab test results	• Differentials for PC • Pre-clinical triage • Shortcuts in case of immediate danger • Explanation of differentials • Recommended further lab testing	• Worldwide • Top 1000 conditions • 15 language versions • Web, mobile, chatbot, voice
Inspired Ideas	Dr. Elsa	• Age, gender • Risk factors • Region/ time of year • Multiple symptoms • Travel history • Answers to discrete dialog questions • Lab test results • Clinicians hypothesis	• List of possible differentials • Condition explanations • Referral & lab test recommendations • Recommended next steps • Clinical triage	• Tanzania, East Africa • Languages: English and Swahili • Android/iOS/Web/API • Users: healthcare workers/ clinicians
Isabel Healthcare	Isabel Symptom Checker	• Age • Gender • Pregnancy Status • Region/Travel History • Free text input of multiple symptoms all at once	• List of possible diagnoses • Diagnoses can be sorted by 'common' or 'Red flag' • Each diagnosis linked to multiple reference resources • If triage function selected, patient answers 7 questions to obtain advice on appropriate venue of care	• 6,000 medical conditions covered • Unlimited number of symptoms • Responsive design means website adjusts to all devices • APIs available allowing integration into other systems • Currently English only but professional site available in Spanish and Chinese and model developed to make available in most languages
Kahun	Kahun decision	• Age, sex • Risk factors • Pregnancy Status	• Pre-clinical triage • Differentials for PC	• Worldwide • Multilingual

	support, Patient1st	• Answers to discrete dialog questions • Lab test results	• Shortcuts in case of immediate danger	• Over 6000 medical conditions • Android/iOS/Web/API
Visiba Group AB	Red Robin	• Age • Gender • Free text input, several symptoms and diagnosis suspected by patient possible as initial complaints; • Identification of administrative cases based on initial free text • Region/ time of year • Discrete answers • Lab results, inputs from devices enabled; device on image recognition within dermatological domain integrated	• Automated patient history collection • List of differential diagnoses • Pre-clinical triage • Urgency • Type (message, video, call, physical) • Profession (physician, nurse, psychologist, etc.)	• Language: Swedish, English, Finnish, Norwegian • Android/iOS/Web • Users: patients (age 1+) and their proxies, medical and administrative personnel
XUND Solutions	XUND App	• Age • Gender • Risk factors • Guided dialogue • Standardised answers	• Pre-clinical triage • In-depth explanations • Recommendations • Navigation within healthcare system	• Europe (CEE & CIS) • Primary healthcare (350 conditions); up to 500 planned • German, English, Hungarian • Patient-centered • Mobile & API

3.4.2 Other systems for AI-based symptom assessment

Table 3 lists the providers of AI symptom assessment systems who have not joined the topic group yet. The list is most likely incomplete and suggestions for systems to add are appreciated. The list is limited to systems that actually have some kind of AI that could be benchmarked. Systems that e.g., show a static list of conditions for a given finding or pure tele-health services have not been included.

Table 3 – Symptom assessment systems outside the topic group

Provider	System
Aetna	Symptom checker
AHEAD Research	Symcat
Curai	Patient-facing DDSS / Chatbot
DocResponse	DocResponse
Doctor Diagnose	
Drugs.com	Symptom Checker
EarlyDoc	
FamilyDoctor.org	Symptom Checker

Healthline	Symptom Checker
Healthtap	Symptom Checker (for members)
K Health	K app chatbot
Mayo Clinic	Symptom Checker
MDLive	Symptom checker on MDLive app
MEDoctor	Symptom Checker
Mediktor	or Mediktor app
NetDoktor	Symptom Checker
PingAn	Good Doctor app
Sharecare, Inc.	AskMD
WebMD	Symptom checker

3.4.3 Input data

AI systems in general are often described as functions mapping an input space to an output space. To define a widely accepted benchmarking it is important to collect the different input and output types relevant for symptom assessment systems.

3.4.3.1 Input types

Table 4 gives an overview of the different input types used by the AI systems listed in Table 2.

Table 4 – Overview symptom assessment system inputs

Input Type	Short Description
General Profile Information	General information about the user/patient like age, sex, ethnics and general risk factors.
Presenting Complaints	The health problems the users seek advice for. Usually entered in search as free text.
Additional Symptoms	Additional symptoms answered by the user if asked.
Lab Results	Available results from lab tests which the user could enter if asked.
Imaging Data (MRI, etc.)	Available imaging data that the use could upload if available digitally.
Photos	Photos of e.g., skin lesions.
Sensor Data	Data from self tracking sensor devices like scales, fitness trackers, 1-channel ECG
Genomics	Genetic profiling information from sources like 23andMe.

3.4.3.2 Ontologies for encoding input data

For benchmarking, the different input types need to be encoded in a way that allows each AI to "understand" its meaning. Since natural language is intrinsically ambiguous, this is achieved by using a terminology or ontology defining concepts like symptoms, findings and risk factors and linking them with a unique identifier, the most commonly used names in selected languages and often a set of relations describing e.g., the hierarchical dependencies of "pain in the left hand" and "pain in the left arm".

There is a large number of ontologies available (e.g., at https://bioportal.bioontology.org/). However most ontologies are specific for a small domain, not well maintained or have grown to a size where they are not consistent enough for describing case data in a precise way. The most relevant input space ontologies for symptom assessment are described in the following sub sections.

3.4.3.2.1 SNOMED Clinical Terms

SNOMED CT (http://www.snomed.org/) describes itself with the following five statements:

- Is the most comprehensive, multilingual clinical healthcare terminology in the world.
- Is a resource with comprehensive, scientifically validated clinical content.
- Enables consistent representation of clinical content in electronic health records.
- Is mapped to other international standards.
- Is in use in more than eighty countries.

Maintenance and distribution are organized by SNOMED International (trading name for the International Health Terminology Standards Development Organisation). SNOMED CT is seen to date as the most complete and detailed classification for all medical terms. SNOMED CT is only free of charge in member countries. In non-member countries the fees are prohibitive. While being among the largest and best maintained ontologies, it is partially not precise enough for encoding symptoms, findings and their details in a unified unambiguous way. Especially for phenotyping rare disease cases it does not yet have high enough resolution (e.g., Achromatopsia and Monochromatism are not separated or "Increased VLDL cholesterol concentration" is not as explicit as e.g., "increased muscle tone"). SNOMED CT is also missing significant parts of the space of self-reportable symptoms of special importance for self-assessment AIs. SNOMED CT is also currently adapted to fit the needs of ICD-11 to link both classification systems (see below).

3.4.3.2.2 Human Phenotype Ontology (HPO)

The Human Phenotype Ontology[1] (HPO) is an ontology focused on phenotyping patients especially in context of hereditary diseases, containing more than 17,000 terms. In the context of rare disease it is the most commonly used ontology and was adopted by Orphanet for encoding the conditions in their rare disease database. Other examples are the 100K Genomes UK, NIH UDP, Genetic and Rare Diseases Information Center (GARD). The HPO is part of the Monarch Initiative, an NIH-supported international consortium dedicated to semantic integration of biomedical and model organism data with the ultimate goal of improving biomedical research [7].

3.4.3.2.3 Logical Observation Identifiers Names and Codes (LOINC)

LOINC is a standardized description of both, clinical and laboratory terms. It embodies a structure / ontology, linking related laboratory tests / clinical assessments with each other. It is maintained by the Regenstrief Institute. LOINC covers the domain of clinical observations, it can be used for symptoms, scales and specially results from clinical studies and procedures.

3.4.3.2.4 Unified Medical Language System (UMLS)

The UMLS, which is maintained by the US National Library of Medicine, brings together different classification systems / biomedical libraries including SNOMED CT, ICD, DSM and HPO and links these systems creating an ontology of medical terms. UMLS contains very useful lexical resources, useful to develop NLP tools. It is very rarely used for clinical coding.

[1] www.human-phenotype-ontology.org

3.4.4 Output data

Beside the inputs, the outputs need to be specified in a precise and unambiguous way too. For every test case the output needs to be clear so that the scores and metrics can reveal the distance between the expected results and the actual output of the different AI systems.

3.4.4.1 Output types

As for the input types, Table 5 lists the different output types that the systems listed in Table 1 generate.

Table 5 – Overview symptom assessment system outputs

Output Type	Short Description
Clinical Triage	Initial classification/prioritization of a patient on arrival in a hospital / emergency department.
Pre-Clinical Triage	A general advice of the severity of the problem and on how urgent actions need to be taken ranging from e.g., "self-care" through "see a doctor within 2 days" to "call an ambulance right now"
Differential Diagnosis	A list of diseases that might cause the presenting complaints, usually ranked by some score such as probability.
Next Step Advice	More concrete advice suggesting doctors or services that can help with the specific problem.
Treatment Advice	Concrete suggestions of how to treat the problem e.g., with exercises, self-medication etc.

The different output types will be explained in detail in the following section:

3.4.4.1.1 Clinical triage

The simplest output of symptom based symptom assessment systems is a pre-clinical triage. Triage is a term commonly used in clinical context to describe the classification and prioritization of patients based on their symptoms. Most hospitals use some kind of triage systems in their emergency department for deciding how long a patient can wait so that people with severe injuries are treated with higher priority than stable patients with minor symptoms. One triage system commonly used is the Manchester Triage System (MTS) which defines the levels shown in Table 6.

Table 6 – Manchester Triage System levels

Level	Status	Colour	Time to Assessment
1	Immediate	Red	0 min
2	Very urgent	Orange	10 min
3	Urgent	Yellow	60 min
4	Standard	Green	120 min
5	Non urgent	Blue	240 min

The triage is usually performed by a nurse for every incoming patient in a triage room equipped with devices of measuring the vital signs. While there are some guidelines, clinics report a high variance in the classification between different nurses and on different days.

3.4.4.1.2 Pre-Clinical triage

As triage helps with the prioritization of patients in an emergency setting, the pre-clinical triage helps users of self-assessment applications independent of a diagnosis to decide when and where to seek what kind of care. In contrast to the clinical triage where there are several methods known, pre-clinical triage is not standardized. Different companies use different in-house classifications. Inside the topic group the following classifications have been used:

1DOC3

- No need for any other medical attention
- Should have a medical appointment in a few weeks or months
- Should have a medical appointment in a few days
- Should have a medical appointment in a few hours
- Should have a medical attention immediately

Ada Health Pre-Clinical Triage Levels

- Self-care
- Self-care Pharma
- Primary care 2-3 weeks
- Primary care 2-3 days
- Primary care same day
- Primary care 4 hours
- Emergency care
- Call ambulance

Babylon Pre-Clinical Triage Levels

Generally:

- Self-care
- Pharmacy
- Primary care, 1-2 weeks
- Primary care, same day urgently
- Emergency care (usually transport arranged by patient, including taxi)
- Emergency care with ambulance

With additional information provided per condition.

Deepcare Triage Levels

- Self-care
- Medical appointment (as soon as possible)

- Medical appointment same day urgently
- Instant medical appointment (Teleconsultation)
- Emergency care
- Call ambulance

Healthily Pre-Clinical Triage Levels

- Self-limiting
- Self-care
- Primary care 2 weeks
- Primary care 2 days
- Primary care same day
- Emergency A&E
- Emergency ambulance

Infermedica Triage Levels

- Self-care
- Medical appointment
- Medical appointment within 24 hours
- Emergency care / Hospital urgency
- Emergency care with ambulance

On top of that, the system provides information on whether remote care is feasible (e.g., teleconsultation). Additional information provided per condition (e.g., doctor's specialty in case of medical appointments).

Inspired Ideas Triage Levels

- Self-care
- Admit patient / in-patient
- Refer patients to higher level care (District Hospital)
- Emergency Services

Triage is completed by a community health worker/ clinician, typically at a lower level health institution such as a village dispensary.

Isabel Pre-Clinical Triage Levels

- Level 1 (Green): Walk in Clinic/Telemedicine/Pharmacy
- Level 2 (Yellow): Family Physician/Urgent Care Clinic/Minor Injuries Unit
- Level 3 (Red): Emergency Services

Isabel does not advocate self-care and assumes the patient has decided they want to seek care now but just need help on deciding on which venue of care.

Visiba Care Pre-Clinical Triage Levels

Recommended contact within: 8 – Self-care

7 – 1 month

6 – 1 week

5 – 24 hours

4 – 4 hours

3 – 60 min

2 – 10 min

1 – 0 min

Depending on the condition additional triage dimension is recommended format of contact (physical or digital) and profession.

For the standardized benchmarking the topic group agree to start with the following set:

- Self-Care (SC)
- Primary care (PC)
- Emergency care (EC)

3.4.4.1.3 Differential diagnosis

Differential diagnosis is presented to the user as a list of conditions which may explain their symptoms. Depending on the intended use of the symptom checker, this may be intended to give the user an answer to the question “what is wrong with me”, to inform conversations with a medical professional or to provide the user with further information on conditions that may be relevant.

Depending on the symptom checker, the list of conditions may be given using a standard ontology such as SNOMED CT or as a list of condition names. The list may be an unranked list of possible conditions, a ranked list with or without an indication of category (e.g., high/medium/low probability) or a ranked list with explicit values of absolute or relative probability.

Differential diagnosis may also be combined with triage, with the triage level associated with each condition displayed along with an overall triage level.

3.4.4.1.4 Next steps advice

Next Steps Advice exist to guide the user towards a suggested action to take and is often based on pre-clinical triage. Next Steps vary in their granularity across different tools. They can guide the user towards specific services or institutions in order to further assess or treat their symptoms or conditions that may be compatible with their symptoms. There is sometimes a brief explanation of the type of action the recommended service might take. Examples of services or institutions that are recommended across the various tools include primary care doctors, genitourinary medicine clinics, pharmacists, dietitians and psychologists. These services and institutions tend to be localized to

country-specific guidelines or recommendations. Specific named services and institutions may also be recommended in line with commercial partnerships. Some Next Steps Advice sections also provide information about what action the user should take if their symptoms change, worsen or do not improve, as a form of ‘safety netting.’

3.4.4.1.5 Treatment advice

Treatment Advice provides the user with advice as to how manage the user’s symptoms or condition compatible with their symptoms. This can be in the form of possible treatment options that might be suggested by a recommended service or institution outlined in the Next Steps Advice (e.g., medicine that might be prescribed) or may be concrete suggestions of how to treat the problem if it is a condition or symptom that can be managed with self-care (e.g., simple painkillers, exercises). Treatment Advice is often generic and common across countries based on the evidence base for the condition or symptom in question, but may on occasion be informed by local guidelines or recommendations; further information can also be provided in the form of links to medically validated health information sources.

3.4.5 Scope dimensions

The table of existing solutions also lists the scope of the intended application of these systems. Analysing them suggests the following dimensions should be considered as part of the benchmarking:

Regional scope

Some systems focus on a regional condition distribution and symptom interpretation, whereas others don't use the regional information. As this is an important distinction between the systems, the benchmark should present the results by region as well as the overall results. Since the granularity varies, starting at continent-level but also going down to the neighbourhood-level, the reporting most likely needs to support a hierarchical or multi-hierarchical structure.

Condition set

With subtypes there are several thousand known conditions. The systems differ in the range as well in depth of condition they support. Most systems focus on the top 300 to top 1500 conditions while others also include the 6000-8000 rare diseases. Other systems have a narrower intended focus e.g., tropical diseases or single disease only. The benchmarking therefore needs to be categorized by different condition sets to account for the different system capabilities.

Age range

Most systems are created for the (younger) adult range and highly based on these conditions. Only few are explicitly created for paediatrics, especially very young children and some try to cover the whole lifespan of humans. The benchmarking therefore needs to be categorized into different age ranges.

Languages

Though there are some systems covering more than one language, common systems are created mostly in English. As it is essential for patient-facing applications to provide low-thresholds for everyone to access this medical information, this dimension may be considered as well - especially if at some point the quality of natural language understanding of entered symptoms is assessed. It is also noteworthy that some regulatory frameworks for software as medical devices prohibit using them with non-native-speakers.

3.4.6 Additional relevant dimensions

Besides scope, technology and structure, the analysis of the different applications revealed several additional aspects that need to be considered to define the benchmarking:

Dealing with "No-Answers" / missing information

Some systems are not able to deal with missing information as they require always a "yes" or "no" answer when asking patients. This may be a challenge for benchmarking with e.g., case vignettes as it won't be possible to describe the complete health state of an individual with every detail that is imaginable.

Dialog engines

More modern systems are designed as chatbots engaging in a dialog with the user. The number of questions asked is crucial for the system performance and might be relevant for benchmarking. Furthermore, dialog-based systems proactively asking for symptoms are challenging if case vignettes are used for benchmarking since the dialog might not ask for the symptoms in the vignettes. Later iterations of the benchmarking might explicitly conduct a dialog to include the performance of the dialog, while first iterations might provide the AIs with complete cases.

Number of presenting complaints

The systems differ in the number of presenting complaints users can enter. This might influence the cases used for benchmarking e.g., by starting with cases having only one presenting complaint.

Multimorbidity

Most systems don't support the possibility that a combination of multiple conditions is responsible for the users presenting complaints (multi-morbidity). The benchmarking therefore should mark multi-morbid and mono-morbid cases and differentiate the reported performance accordingly. The initial benchmarking might also be restricted to mono-morbid cases.

Symptom search

Most systems allow to search for the initial presenting complaints. The performance of the search and whether the application is able to provide the correct finding given the terms entered by users, is also crucial for the system performance and could be benchmarked.

Natural language processing

Some of the systems support full natural language for both the presenting complaints the dialog in general. While these systems are usually restricted to few languages, they provide a more natural

experience and possible more complete collection of the relevant evidence. Testing the natural language understanding might therefore be another dimension to consider in the benchmarking – a point that became even more important with the rise of GPT based applications in early 2023.

Seasonality

Some systems consider seasonal dynamics in certain conditions. For example, during springtime there can be a spike in allergies and hence, relevant conditions may be more probable than during other periods. Other examples include influenza spikes in winter or malaria in rainy seasons.

3.4.7 Robustness of systems for AI based symptom assessment

As meeting D underlined with the introduction of a corresponding ad-hoc group, robustness is an important aspect for AI systems in general. Especially in recent years it could be shown that systems performing well on a reasonable benchmarking test set completely fail if adding some noise or a slight valid but unexpected transformation/distortion to the input data. For instance, traffic signs might not be recognized any more if a slight modification like a sticker is added that a human driver would hardly notice. Based on the knowledge of such behaviours, the results of AI systems could be deliberately compromised e.g., to get more money from the health insurance for a more expensive disease or faster appointments.

A viable benchmarking should therefore also assess the robustness. While for e.g., machine learning based image processing technologies robustness is a more important issue, symptom assessment can also be compromised. The reminder of this section gives an overview of the most relevant robustness and stability issues that should be assessed as part of the benchmarking.

Memory stability & reproducibility

An aspect of robustness is also the stability of the results. For instance, a technology might use data structures like hash maps that depend on the current operating systems memory layout or apply some sampling approach. In this case running the AI on the same case after restart again might lead to slightly different, possibly worse results.

Empty case response

AI should respond correctly to empty cases e.g., with an agreed-upon error message or some "uncertain" expressing that the given evidence is insufficient for a viable assessment.

Negative evidence only response

Systems should have no problems with cases containing only negative additional evidence besides the presenting complaints.

All symptoms response

Systems should respond correctly to requests giving evidence to all i.e. several thousand symptoms rather than e.g., crashing.

Duplicate symptom response

The systems should be able to deal with requests containing duplicates e.g., multiple times with the same symptom - possibly even with contradicting evidence. This might include cases where a presenting complaint is mentioned in the additional evidence again. A proper error message pointing on the invalid case would be considered as correctly dealing with duplicate symptoms.

Wrong symptom response

Systems should respond properly to unknown symptoms.

Symptom with wrong attributes response

Systems should respond properly to symptoms with wrong/incorrect attributes.

Symptom without mandatory attribute response

Systems should respond properly to symptoms with missing but mandatory attributes.

4 Ethical considerations

Across the world, people increasingly use digital infrastructures, such as dedicated health websites, wearable technologies and also AI-based symptom assessment systems, to improve and maintain their health. A UK survey found that a third of the population uses internet search engines for health advice. This digitally mediated self-assessment also occurs in countries in the global South, where access to healthcare is often limited but where mobile and internet penetration over the last decade has rapidly increased.

Ethical, human rights, and cultural considerations should always be placed at the centre of design, development, and deployment of an AI technology in health care, including AI-based symptom assessment technologies. This section employs the guiding principles introduced in the World Health Organization's guidance on the Ethics and Governance of Artificial Intelligence for Health [8], and the specific ways in which symptom checkers raise ethical challenges that should be addressed. Furthermore, this section also considers the existing economic and social inequalities within societies and health systems, and worldwide, and how such inequities and realities can shape symptom assessment systems and their deployment.

4.1 Protect autonomy

4.1.1 Effect on decision-making in health

AISAs will modify how individuals seek care within a healthcare system. This can include persuading individuals to proactively seek out medical advice that is not required, or foregoing medical care that is required. Healthcare workers using AISAs may come to rely heavily upon them. This could either augment their capability to make an accurate judgement or could lead to automation bias, or the tendency of health workers to uncritically rely on the outputs of a machine and surrendering their own judgment and expertise.

How health workers respond to the use of AISA's by their peers or patients depends in part on the health care system, including the health system's human resource capacity, as well as existing supplies of medicines and diagnostic tests. For instance, if the AISA makes suggestions for next steps unavailable or inaccessible to users, the user may choose not to use the AISA, turning instead

to alternative forms of medical advice and treatment. Existing hierarchies can also shape individual health-seeking behaviour. For instance, a healthcare worker may feel undermined if a patient ignores their medical advice in favour of that given by the AISA, potentially hindering the patient's access to healthcare. Similarly, patients should be informed if healthcare workers use AI-based symptom checkers as the primary means to determine a diagnosis.

One suggestion in [8, Chapter 5.1] to maintain human autonomy is to offer ranked decisions instead of single decisions offering the user more choices, as it can already been found in most symptom assessment systems, usually offering ranked list their outputs (pre-clinical triage, next steps, underlying causes, etc.). Providing training to health workers on the appropriate use of AI-based symptom checkers is another mechanism to overcome or avoid automation bias, and to provide patients with appropriate notice that a symptom checker may assist in medical diagnosis or treatment.

4.1.2 Protection of privacy of personal health information

Symptom assessment AIs must meet legal standards with respect to data governance, privacy and data quality. Those standards also apply to the benchmarking process of AISAs. Evaluation of such systems requires test data in the form of clinical vignettes. Depending on the approach for creating the test data set, this might involve real anonymized patient cases in which privacy and protection are crucial. Given the importance of this issue, the focus group actively worked on ensuring that such data meet high standards for ethics and the protection of personal data, and applied the principles and requirements of several regulations such as the European Union's General Data Protection Regulation, other national data protection laws, and eventually new personal data regulations that may be introduced in artificial intelligence regulations currently under discussion (see chapter 8 for more details on regulatory considerations).

4.2 Promote human well-being, human safety and the public interest

AISAs must be technically robust and safe to use. They must continue working in the contexts they were designed for but also anticipate potential changes to those contexts. Symptom assessment AIs may be maliciously attacked or may break down. This can create problems if health systems and workers have become dependent on such technologies, and therefore, there should be contingency measures if such failures occur, as well as a focus on maintaining capacity in health systems so that health workers can continue to diagnose and treat medical needs without the use of an AISA. As discussed in Chapter 8, most AI-based symptom have to be classified as medical devices, which would therefore require appropriate testing and approval by a relevant regulatory authority prior to deployment in a health care setting (for LLM-based symptom assessment systems see for instance [44]).

4.2.1 The ethical implications of introducing benchmarking

Setting up benchmarking of symptom assessment systems will help assess their accuracy, a vital quality dimension. Performance assessment is critical in considering the ethical and cultural dimensions and implications of using AISAs compared to other options, including the aforementioned digital-based solutions like web-search or human experts – with variable levels of expertise, accessibility and supportive infrastructures, such as diagnostic tests and drugs.

The quality of technology should be assessed in multifaceted ways that go further than benchmarking via independently curated data sets. The metrics and results should be interpreted with a recognition of the limitations that these data sets might have. While there should be systematic efforts to ensure appropriate representation of all demographics in data sets used for AISAs, all data sets are inherently prone to bias of the majority data set. There is also a tendency of

those who curate the data to introduce underrepresentation of subpopulations and minorities (see section 4.5).

Test data used for benchmarking does not cover the whole possible input and output space, including various symptoms and diseases. Additionally, combinations of symptoms and comorbidities make it impossible to include all possible conditions in the evaluation. Therefore, benchmarking should not replace prospective studies, monitoring in real-world clinical settings and post market surveillance. Additionally, standardized benchmarks could give the illusion of safety that comes with standards.

The 'users' of the outputs of benchmarking (e.g., governments, health systems, regulators or clinicians) should ensure benchmarking alone is not seen as a guarantee of the safety or efficacy of AISAs and instead as one part of a more holistic evaluation.

4.3 Ensure transparency, explainability and intelligibility

Transparency for AI models should provide enough information about the model and its development so that it is possible for regulators to capture limitations and possible errors. In contrast to machine learning, symptom assessment systems are often knowledge-based models created by medical experts based on literature and studies, making them in principle easier to understand and explainable.

However, developers often claim that the specific knowledge extracted from resources and how such knowledge is represented is commercially confidential information. Therefore, the knowledge and model specifics are often not accessible to users and only occasionally is available to regulators (this is because during audits ordered by regulatory bodies, AI developers usually must talk through the modelling approach), making the models less transparent. Even if they would be more transparent, the time, knowledge and experience needed to comprehend the implications of implementation details would be challenging.

An increasing number of symptom assessment system also provides some explainability e.g., by indicating the contribution of inputs (e.g., symptoms) to the predicted outputs (triage, conditions). This ensures the output is more intelligible for the user and can be verified by a regulators and medical experts. Explainability is also increasingly required in regulations. For example, it is already required for automated decision-making under many countries' data protection laws.

General processes for quality assurance must be implemented by the AI manufacturers to establish transparency and trust in the AI development. Since symptom assessment systems have to be considered as software as a medical device, the implementation of comprehensive quality processes is mandatory. This includes adequate documentation and transparency depending on the assigned medical device class for clinical decision support systems.

4.4 Foster responsibility and accountability

The developer of a health application is responsible for the performance of that application, given the intended use and appropriate use conditions. Since most AISAs are knowledge-based models that use knowledge from different sources, they differ from ML models with respect to responsibility. With ML, it can be difficult to assign responsibility because there are 'many hands' that may be involved in the development and deployment of a model. For knowledge-based

systems, the selection of knowledge by medical experts, the assessment of the quality and how to incorporate the information into the model are the tasks that require high responsibility, and responsibility can be clearly assigned to one or more parties involved in the development of the model.

AISAs raise serious ethical questions around accountability. Standards for AISA accountability is not established, and guidelines are legally not binding. General challenges regarding accountability for clinical decision support systems are discussed in chapter 6.4 of the WHO Ethics Consideration guideline [8]. It is important to mention that currently AISAs are not used as autonomous decision-makers, which would change accountability considerations with regard to human judgment (see chapter 6.5 in [8]).

One major discussion with respect to accountability for AISAs is the differences between the intended use cases. In clinical use, accountability can be either transferred to the AI or the medical expert, depending on the error. Where accountability lies between AI and a medical expert may depend on the standard of care. If the standard of care requires a medical provider to use an AISA, and such AI-system generates an incorrect response that a medical provider could reasonably rely upon, accountability is likely to rest with the developer or expert who designed the system. If the medical provider is using an AISA outside of the standard of care, the provider may be held accountable for any harm.

For self-assessment apps, expectations of accountability and liability are set by the individual systems terms and conditions. For free self-assessment applications, the T&Cs usually state that the user should not take action based on the information provided without consulting a doctor or medical professional. In addition to this, in most countries the individual product liability laws apply. In any case it is important that the system transparently communicates to the user to what degree the user can rely on the result without further confirmation.

4.5 Ensure inclusiveness and equity

AISAs are based on medical resources such as literature, studies and expert knowledge. It is known that older studies have a bias towards male patients, so inferred knowledge has not and is not always applicable to female patients [9]. Biases apply not only to sex and gender but also to other socio-demographic attributes like ethnicity, race, age, residence or income. Those biases still exist in medical literature and can impact AISAs. It is not always possible to remove biases due to unavailable resources, language barriers or differences in data collection. However, it is important to account for those biases by stating which populations were included in the used resources and also by specifying the user group of the AISA.

For benchmarking, clinical vignettes are used to evaluate the AISA. Here, fairness between different populations could be assessed by using diverse patient vignettes and established fairness metrics.

AISAs usually conduct a dialogue to collect symptoms and other medical evidence. The educational background, disabilities and medical knowledge of the user could also influence the responses and therefore the outcome of the AISA. While the benchmarking focuses mostly on the predictive performance of the knowledge-based system rather than the AI used for the dialog, inclusiveness should be still taken into account and evaluated.

Another fairness issue with regard to data collection revolves around how authority has been established for the ownership, use and transfer of data. There may be inequalities at every level of a

health system and economy, whether at the level of the patient and provider, to the relationship between governments and companies or people and companies. Glossing over exchanges between these actors as mutually beneficial or egalitarian may obscure these inequalities. For instance, an actor may agree to provide health data in exchange for better-trained models or even for a subsidized or free service but, in the process, may lose control over how that data is subsequently used.

4.5.1 Wider potential societal effects of AISAs

There may also be long-term effects of AISAs on public healthcare systems. For instance, they may discourage policymakers from investing in human resources for health. This may adversely affect vulnerable, marginalized or remote populations who are unable to use AISAs due to factors including a lack of adequate digital data infrastructures and digital illiteracy. This could both reflect and exacerbate an existing 'digital divide'. Furthermore, in the case of clinician-facing AISAs, consideration would need to be put to re-training health workers, many of whom are increasingly required to utilize digital technologies and health information systems for which they had not previously received training or education to use.

Additionally, the role of private actors and their relationship to public actors in developing AISAs needs to be considered. Private ownership and control of data and algorithms for health may have several implications, especially if the development and use of an AISA also relies upon the public sector, including the data used to develop the AISA. For instance, how might the user data collected by the AISA's be used or sold? How might corporations shape the design of AISA's - for instance, for what populations or diseases? What role does the public sector maintain in governing such technology?

4.6 Promote artificial intelligence that is responsive and sustainable

AISAs will also rely upon existing digital infrastructures that consume resources in their design, production, deployment and utilization. Responsibility around this digital infrastructure is dispersed across many bodies, but the group should at least be aware of the harms that may exist to the environment along the supply chain, including the resources consumed by (and carbon footprint of) training models, the requirement for new devices and the disposal of outdated or non-functioning hardware.

5 Existing work on benchmarking

This section focuses on the existing benchmarking processes for assessing the quality of AI-based symptom-assessment systems. It addresses different aspects of the existing work on benchmarking of AI systems (e.g., relevant scientific publications, benchmarking frameworks, scores and metrics and clinical evaluation attempts). The goal was to collect all relevant learnings from previous benchmarking that could inform the benchmarking process in this topic group.

5.1 Self-Assessment

5.1.1 Publications on benchmarking systems

While a representative comparable benchmarking for AI-based symptom-assessment does not yet exist, some work has been done in the scientific community assessing the performance of such systems. This section summarizes insights from the most relevant publications on this topic. It covers parts of the deliverable DEL7 *"AI for health evaluation considerations,"* DEL7.1 *"AI4H evaluation process description,"* DEL7.2 *"AI technical test specification,"* DEL7.3 *"Data and*

artificial intelligence assessment methods (DAISAM)," and DEL7.4 *"Clinical Evaluation of AI for health"*.

To establish a standardized benchmarking for AI-based symptom assessment systems, it is valuable to analyse previous benchmarking work in this field. So far, little work has been performed, which is also a reason that the introduction of a standardized benchmarking framework is important.

5.1.1.1 "ISABEL: Accuracy of a Machine Learning Based Ddx Generator" [10]

563 cases of diagnostic error were collected over a period of 2 years from case reports, journals and detailed press articles. The cases covered 300 diagnoses and 27 specialties and, on average, contained 6 clinical features each. The free text case presentations were entered into Isabel DDx Generator and the position of the known final diagnosis within the tool's list of ranked possible diagnoses recorded. Results: In 74% of the cases the final diagnosis was in the top 3 suggestions. In 87% of cases the final diagnosis was in the top 5 suggestions and in 98% of cases the final diagnosis was in the top 10 suggestions.

5.1.1.2 „Asking ISABEL for diagnostic dilemmas in pediatrics: How does a web based diagnostic checklist perform?" [11]

Using a case-based textbook, 10 participants selected keywords from 25 cases; each read the same cases and were blinded to the diagnosis. Age, gender and keywords were entered into Isabel. The primary outcome measure was Isabel's inclusion of the diagnosis on 'Page 1' (top 10 diagnoses) and 'View all' (top 30 diagnoses). The secondary outcome measure was the impact of level of training on Isabel's success rate. Lower level of training (LLT) was defined as medical student and resident. Higher level of training (HLT) was defined as junior and senior faculty.

Isabel's performance with published cases:

- Isabel included the diagnosis in 60% (149/248) of cases on 'Page 1' and 81% (202/248) on 'View all' (p=0.001).
- With LLT users, Isabel included the diagnosis in 55% (55/100) of cases on 'Page 1' compared to 64% (94/148) with HLT (p=0.18).
- With LLT users, Isabel included the diagnosis in 78% (78/100) of cases on 'View All' compared to 84% (124/148) with HLT (p=0.25).

5.1.1.3 Performance of a Web-Based Clinical Diagnosis Support System for Internists [12]

ISABEL was tested using 50 consecutive Internal Medicine case records published in the New England Journal of Medicine. Between 3 and 6 key clinical findings from the case were entered (recommended approach) or the entire case history entered. The investigator entering key words was aware of the correct diagnosis. Graber and Mathew then determined how often the correct diagnosis was suggested in the list of 30 differential diagnoses generated by the clinical decision support system and also evaluated the speed of data entry and results recovery.

The clinical decision support system suggested the correct diagnosis in 48 of 50 cases (96%) with key findings entry and in 37 of the 50 cases (74%) if the entire case history was pasted in. Pasting took seconds, manual entry less than a minute and results were provided within 2–3 seconds with either approach.

5.1.1.4 "Evaluation of symptom checkers for self diagnosis and triage" [13]

Semigran et al. compiled 45 standardized patient vignettes evenly divided between the categories emergent care required, non-emergent care reasonable and self-care reasonable. These were used to test 23 English language symptom checkers. Symptom checkers providing diagnoses selected the correct condition in first place in 34% of evaluations and in the top 20 conditions in 58% of

evaluations. Symptom checkers providing triage returned the appropriate level in 80% of emergent care evaluations, 55% of non-emergent care evaluations and 33% of self-care evaluations, with triage performance of individual symptom checkers ranging from 33% to 78%. The authors concluded that the symptom checker had deficiencies in diagnosis and triage and that the triage provided was generally risk averse.

5.1.1.5 "Comparison of physician and computer diagnostic accuracy." [14]

Semigran et al. expounded on their 2015 systematic assessment of online symptom checkers by comparing checker performance—the previous 45 vignettes—to physician (n=234) diagnoses. Physicians reported the correct diagnosis 38.1% more often symptom checkers (72.1% vs. 34.0%), additionally outperforming in the top three diagnoses listed (84.3% vs. 51.2%). Physicians were also more likely to list the correct diagnoses for high-acuity and uncommon vignettes, while symptom checkers were more likely to list the correct diagnosis for low-acuity and common vignettes. While the study is limited by physician selection bias, the significance of the results lies in the vast outperformance of physician diagnoses.

5.1.1.6 "A novel insight into the challenges of diagnosing degenerative cervical myelopathy using web-based symptom checkers."[15]

Unique algorithms (n=4) from the top 20 web-based symptom checkers were evaluated for their ability to diagnose degenerative cervical myelopathy (DCM): WebMD, Healthline, Healthtools.AARP and NetDoctor. A single case vignette of up to 31 DCM symptoms derived from 4 review articles was entered into each symptom checker. Only 45% of the 31 DCM symptoms were associated with DCM as a differential by the symptom checkers and in these cases a majority 79% ranked DCM in the bottom two-thirds of differentials. Insofar as web-based symptom checkers are able to detect symptoms of degenerative disorder, the authors conclude that there is technological potential, but an overall lack of acuity.

5.1.1.7 "ISABEL: a web-based differential diagnostic aid for paediatrics: results from an initial performance evaluation" [16]

Ramnarayan et al. conducted a study in acute paediatric units in two teaching and two district general hospitals in the southeast of England. 99 hypothetical cases and 100 real life cases were provided by clinicians and these were used to check for the presence of the expected or final diagnosis in the ISABEL output list. Cases covered 14 paediatric specialities and 55 final diagnoses. ISABEL displayed the expected diagnosis in 91% of hypothetical and 95% of real-life cases.

5.1.1.8 "Safety of patient-facing digital symptom checkers." [17]

Fraser et al. examine concerns over the validity of the results of a comparative study of the Babylon symptom checker and human doctors for triage and diagnosis. They advocate the publication of study protocols and data sets used in the evaluation of symptom checkers and the creation of evaluation guidelines specific to symptom checkers have three benefits.

5.1.1.9 "A new artificial intelligence tool for assessing symptoms in patients seeking emergency department care: the Mediktor application" [18]

In a tertiary care university hospital emergency department, 1015 patients not requiring emergency treatment responded to questions from the Mediktor application on a tablet computer. Users and clinicians were blinded to the Mediktor output list of conditions and 622 cases selected as valid. Patients were excluded if they did not meet the inclusion criteria, did not have a discharge diagnosis, had a final diagnosis expressed as a symptom or their final diagnosis was not included in the Mediktor database. Compared to the gold standard of the phyisician’s diagnosis, the symptom assessment reached an F1 Score of 42.9% and F3 score of 75.4% and F10 score of 91.3%.

5.1.1.10 "Evaluation of a diagnostic decision support system for the triage of patients in a hospital emergency department" [19]

The results of a subsequent prospective study to the Moreno et al. (2017) evaluation of Mediktor were published in 2019. This study was also conducted in an emergency room setting in Spain and consisted of a sample of 219 patients. With this setting, the symptom assessment reached an F1 Score of 37.9% and F3 score of 65.4% and F10 score of 76.5%. It was further determined that Mediktor's triage levels do not significantly correlate with the Manchester Triage System for emergency care or with hospital admissions, hospital readmissions and emergency screenings at 30 days.

5.1.1.11 "Evaluation and accurate diagnoses of pediatric diseases using artificial intelligence." [20]

A study by Liang et. al. showed a proof of concept of a diagnostic decision support system for (common) paediatric conditions based on a Natural Language Processing approach to extract clinically relevant information from EHRs. The F1 Score overall was between those for groups of junior and senior physicians, with an average F1 score of 0.885 for the covered conditions.

5.1.1.12 "Evaluating the potential impact of Ada DX in a retrospective study." [21]

A retrospective study evaluated the diagnostic decision support system Ada DX in 93 cases of confirmed rare inflammatory systemic diseases. Information from patients' health records was entered in Ada DX in the cases' course over time. The system's disease suggestions were evaluated against the confirmed diagnosis. The system's potential to provide correct rare disease suggestions early in the course of cases was investigated. Correct suggestions were provided earlier than the time of clinical diagnosis in 53.8% of cases (F5) and 37.6% (F1) respectively. At the time of clinical diagnosis, the F1 score was 89.3%.

5.1.1.13 "Accuracy of a computer-based diagnostic program for ambulatory patients with knee pain." [22]

The results of a prospective observational study were published in 2016 in which researchers evaluated the accuracy of a web-based symptom checker for ambulatory patients with knee pain in the United States. The symptom checker had the ability to provide a differential diagnosis for 26 common knee-related conditions. In a sample size of 259 patients aged above 18 years, the symptom assessment reached an F10 score of 89%.

5.1.1.14 "How Accurate Are Patients at Diagnosing the Cause of Their Knee Pain With the Help of a Web-based Symptom Checker?" [23]

In a follow up to the Blisson et al. (2014) study investigating the accuracy of a web-based symptom checker for knee pain, a prospective study was conducted across 7 sports medicine clinics to evaluate patient's ability to self-diagnose their knee pain with the help of the same symptom checker within a cohort of 328 patients aged 18–76 years. Patients were allowed to use the symptom checker, which generated a list of potential diagnoses after patients had entered their symptoms. Each diagnosis was linked to informative content. Patients then self-diagnosed the cause of their knee pain based on the information from the symptom checker. In 58% of cases, one of the patients' self-diagnoses matched the physician diagnosis. Patients had up to 9 self-diagnoses.

5.1.1.15 "Are online symptoms checkers useful for patients with inflammatory arthritis?" [24]

A prospective study in secondary care in the United Kingdom evaluated the NHS Symptom Checker for triage accuracy and Boots WebMD for diagnostic accuracy against physician diagnosis of inflammatory arthritis: rheumatoid arthritis (n = 13), psoriatic arthritis (n = 4), unclassified

arthritis (n = 4)) and inflammatory arthralgia (n = 13). The study aimed to expand literature into the effectiveness of online symptom checkers in real patients in relation to how the internet is used to search for health information. 56% of patients were suggested the appropriate level of care by the NHS Symptom Checker, while 69% of rheumatoid arthritis patients and 75% of psoriatic arthritis patients had their diagnosis listed amongst the top five differential diagnoses by WebMD. Low triage accuracy led the authors to predict an inappropriate use of healthcare resources as a result of these web-based checkers.

5.1.1.16 "A comparative study of artificial intelligence and human doctors for the purpose of triage and diagnosis" [25]

In the study it was hypothesised that an artificial intelligence (AI) powered triage and diagnostic system would compare favourably with human doctors with respect to triage and diagnostic accuracy. A prospective validation study of the accuracy and safety of an AI powered triage and diagnostic system was performed. Identical cases were evaluated by both an AI system and human doctors. Differential diagnoses and triage outcomes were evaluated by an independent judge, who was blinded from knowing the source (AI system or human doctor) of the outcomes. Independently of these cases, vignettes from publicly available resources were also assessed to provide a benchmark to previous studies and the diagnostic component of the Membership of the Royal College of General Practitioners (MRCGP) exam. Overall, it was found that the Babylon AI powered Triage and Diagnostic System was able to identify the condition modelled by a clinical vignette with accuracy comparable to human doctors (in terms of precision and recall). In addition, it was found that the triage advice recommended by the AI System was, on average, safer than that of human doctors, when compared to the ranges of acceptable triage provided by independent expert judges, with only a minimal reduction in appropriateness.

5.1.1.17 "CONSORT-AI and SPIRIT-AI reporting guidelines" [26]

Adapted from traditional clinical trials guidelines, the CONSORT-AI and SPIRIT-AI reporting guidelines were published in September 2020 after a staged Delphi consensus process of academics, AI experts and clinicians. They comprise of a checklist for anyone reporting the results of a trial that includes an AI intervention and publishing its protocol. The checklist aims to standardise the sharing of information from the trial. The CONSORT-AI and SPIRIT-AI reporting guidelines are a significant and formative part of the evolution of clinical AI research, as they encourage research teams to conduct and report trials in a trusted way. If AI in healthcare is to be trusted by clinical stakeholders and decision makers, then transparent reporting, all the way from data sourcing to clinical outcomes, is key. These reporting guidelines are a first step in establishing a minimum standard of accountability in AI interventions.

5.1.1.18 “A study of automated self-assessment in a primary care student health centre setting” [27]

154 users of a student health centre used an automated self-assessment system prior to a face-to-face consultation with a general practitioner. The system’s triage rating was available to the GP, who also recorded their own triage rating following the consultation. Agreement occurred in 39% of consultations, with the self-assessment tool found to be risk averse and selecting the most urdent level of care in 56% of cases. In its prototype form, the self-assessment system was not a replacement for clinician assessment.

5.1.1.19 "Self-triage for acute primary care via a smartphone application: Practical, safe and efficient?" [28]

A cohort study of the Dutch "Should I see a doctor?" ("moet ik naar de dokter?") self-triage tool, using a questionnaire built in to the application and a follow up call from a nurse. The app achieved triage sensitivity of 84% and specificity of 74% for 126 of the telephoned participants.

5.1.1.20 "Online symptom checker diagnostic and triage accuracy for HIV and hepatitis C" [29]

A retrospective analysis was performed on 8,363 eligible adult patients, including 90 cases of HIV, 67 of hepatitis C and 11 of both HIV and hepatitis C. The outcome of five symptom checkers was compared with diagnosis data from physicians. All symptom checkers had poor diagnostic accuracy for HIV, hepatitis C and combined HIV and hepatitis C, <20% for top 1 and <40% for top 10 diagnoses. Significant variations existed between symptom checkers. Symptom checker diagnostic capabilities were found to be inferior to physician diagnostic capabilities.

5.1.1.21 "How accurate are digital symptom assessment apps for suggesting conditions and urgency advice? A clinical vignettes comparison to GPs" [30]

Eight popular symptom assessment applications were compared against general practitioners for 200 vignettes created and reviewed by experienced clinicians. Seven general practitioners were tested using telephone consultations. The apps were tested by primary cares physicians using the vignettes to play the role of patient. The best performing apps were found to have a high level of triage accuracy close to that of a general practitioner and in top-3 and top-5 condition suggestions, but not in top-1 condition suggestions. The findings of the study were supportive of the use of symptom assessment apps to supplement telephone triage.

5.1.1.22 "The quality of diagnosis and triage advice provided by free online symptom checkers and apps in Australia" [31]

An investigation of the quality of diagnostic and triage advice provided by 36 symptom checkers available in Australia via websites or mobile apps, using 1170 diagnosis vignettes and 688 triage vignettes. The 27 symptom checkers which provided diagnoses had a top-1 accuracy of 36%, top-3 accuracy of 52% and top-10 accuracy of 58%. Symptom checkers using artificial intelligence algorithms listed the correct diagnosis first in 46% of tests, compared with 32 for other symptom checkers. Appropriate triage advice was more frequent for emergency care (63%) and urgent care (56%) vignette tests than for non-urgent care (30%) and self-care (40%) vignettes.

5.1.1.23 "Triage accuracy of online symptom checkers for Accident and Emergency Department patients" [32]

An evaluation of the accuracy of the accuracy of triage provided by two symptom checkers, using 100 randomly sampled Accident and Emergency Department records from the Queen Mary Hospital in Hong Kong. All patients over the age of 18 attending the department in 2016 were included. The triage recommendations of each symptom checker was evaluated for overall sensitivity, sensitivity for emergency cases and specificity for non-emergency cases, when compared with the triage categories assigned by the triage nurses. The two symptom checkers had overall triage accuracy of 74% and 50% and emergency case sensitivity of 70% and 45%. The authors found that the symptom checkers were not suitable as alternatives to Accident and Emergency Department triage protocols due to their low overall sensitivities and negative predictive values.

5.1.1.24 "Accuracy of online symptom checkers and the potential impact on service utilisation" [33]

Twelve publicly available symptom checkers were tested using a standardised set of 50 clinical vignettes, run against each vignette by a non-clinical researcher. Wide variation in performance was found between the available symptom checkers, with top-5 diagnosis accuracy ranging from 22% to 84%. The authors recommended external validation and regulation to ensure that public facing symptom checkers are safe for use.

5.1.1.25 "Performance of a new symptom checker in patient triage: Canadian cohort study" [34]

A newly developed prototype symptom checker was assessed in a cohort study of 281 hospital emergency department patients and 300 family physician clinic patients aged 16 or above. Triage sensitivity was found to be 90% for hospital patients and 97% for primary care patients, outperforming patients' own triage judgements. The authors found that the symptom checker could reduce a significant number of unnecessary hospital visits, with accuracy and safety outcomes comparable to existing data on telephone triage.

5.1.1.26 "Artificial Intelligence-Based Application Provides Accurate Medical Triage Advice When Compared to Consensus Decisions of Healthcare Providers" [35]

An Artificial Intelligence-based symptom checker application was compared to seven emergency medicine providers and five internal medicine physicians, using 50 clinical vignettes. The AI-based application was found to operate at the same or higher level than individual clinicians when determining a triage decision for a vignette. The AI-based application's decisions were consistent with the consensus decisions of all human physicians at a similar rate to the individual human physicians' decisions consistency with the consensus decisions.

5.1.1.27 "Quality of condition suggestions and urgency advice provided by the Ada symptom assessment app evaluated with vignettes optimised for Australia" [36]

The Ada symptom assessment app was tested using 48 vignettes developed for a prior study, including 18 developed specifically for the Australian setting. Top-1 accuracy for diagnoses was measured to by 65% and top-3 accuracy as 83%. Triage advice matched the gold standard in 63% of vignettes.

5.1.1.28 "Agreement and validity of electronic patient self-triage (eTriage) with nurse triage in two UK emergency departments: a retrospective study" [37]

The study took place over eight months in two UK hospital emergency departments. Agreement between nurse-led triage using the Manchester Triage System (MTS) and triage from the eTriage electronic patient self-triage system was examined, for 25,333 patients who used eTriage and also had a recorded MTS triage. Agreement was found to be low, with a 10% under-triage rate and 59% over-triage rate by eTriage compared with nurse triage.

5.1.2 Benchmarking publications outside science

In addition to scientific benchmarking attempts, there are several newspaper articles reporting tests of primarily user-facing symptom assessment applications. Since these articles have not been peer

reviewed and are not always follow following scientific standards, they will not be discussed in this document.

5.1.3 Benchmarking by AI developers

All developers of AI solutions for TG Symptom implemented internal benchmarking systems for assessing the performance. This section will outline the insights and learnings from this work of relevance for benchmarking in this topic group.

Probably the most sophisticated systems for benchmarking symptom assessment systems are the ones created by the different companies developing such systems for internal testing and quality control. While most of the details are unlikely to be shared by the companies, this section points out insights relevant for creating a standardized benchmarking.

Data set weighting

In most test sets the distribution of conditions is not the same as the distribution found in the real world. There are usually a few cases for even the rarest conditions while at the same time the number of common cold cases is limited. This gives rare diseases a much higher weight in the aggregation of the total scores. While this is desirable to make sure that all disease models perform well, in some cases it is more important to measure the net performance of systems in real world scenarios. In this case the aggregation function needs to scale the individual cases results with its expected top match prior probability in order to get the mathematically correct expectation-value for the score. For example, errors on common-cold cases need to be punished harder than errors on cases of rare diseases that only a few people suffer from. The benchmarking should include results with and without correction of this effect.

Medical distance of the top matching diseases to the expected ones

In cases where the expected condition is not in the first position and the listed conditions are not in a set of "expected other conditions", the medical distance between the expected conditions and actual conditions should be included in the measure to take into account whether the suggestions are not only not correct, but (dangerously) wrong – something that is usually ignored in benchmarking publications.

Expected condition position

In case the expected condition is not in the first position, the actual position might be part of the scoring. This could include the probability integral of all higher-ranking conditions or the difference between the top scores and the score of the expected disease.

The role of secondary matches

Since AISA systems usually present multiple possible conditions, even if the top match is correct the quality of the other matches needs to be considered as well. For example, highly relevant differentials that should be ruled out are much better secondary diagnoses than random diseases.

5.1.4 Relevant existing benchmarking frameworks

Triggered by the hype around AI, recent years have seen the development of a variety of benchmarking platforms where AIs can compete for the best performance on a determined data set. Given the high complexity of implementing a new benchmarking platform that also meets

regulatory requirements (e.g., implementing a quality management system), the preferred solution is to use an established one. This section reflects on the different existing options that are relevant for this topic group and includes considerations of using the assessment platform that is currently developed by FG-AI4H and presented by deliverable DEL7.5 *"FG-AI4H assessment platform"* (the deliverable explores options for implementing an assessment platform that can be used to evaluate AI for health for the different topic groups).

Document C031 provides a list of the available platforms. While not specific for symptom assessment it provides important examples for many aspects of benchmarking ranging from operational details, over scores & metrics, leader-boards, reports to the overall architecture. Due to high numbers of participants and the prestige associated with a top rank, the platforms have also substantial experience in designing the benchmarking in a way that is hard or impossible to manipulate.

In response to the call for benchmarking platforms (FGAI4H-C-106), in meeting D in Shanghai FGAI4H-D-011 suggested the use of AICrowd. As discussed in meeting D, the topic group had a look at AICrowd to get a first overview if it could be an option for benchmarking the AI systems in this topic group.

Please note that the final solution chosen by the focus group was that EvalAI that was not part of the initial evaluation.

5.1.4.1 TG Symptom benchmarking platform requirements

While many AI benchmarks also involve tasks in health, the benchmarking for this topic group has some specific requirements that will be discussed in this section.

Custom scores & metrics

For the tasks benchmarked by the common benchmarking platforms the focus is on only a small number of scores. In many challenges it is possible to use common ready-made built-in scores and metrics. For benchmarking symptom assessment AI a multitude of new scores and metrics is needed to reflect the different aspects of the quality and performance of self-assessment systems. It is therefore important that the benchmarking platform allows to define and add new custom scores - ideally by configuration rather than changing the platform code, to compute them as part of the benchmarking and automatically add them to the generated reports.

Custom reports & additional reporting dimensions

Together with the custom scores, the platform also needs to support the generation of reports that include all the scores in a readable way.

Interactive custom reports & data export

Since the number of dimensions and score will grow fast, it will not always be possible to automatically provide the reports answering all the details for all possible use cases. For this case the platform needs to either provide interactive navigation and filtering of the benchmarking result data or at least an easy way to export the data for further processing e.g., in tools like Tableau.

Support for interactive testing

Whilst for the first benchmarking iterations providing cases with all the evidence at once might suffice, later iterations should test the quality of the dialog between the system and the user e.g., only answering questions the AI systems explicitly ask for. The platform should allow a way to implement this dialog simulation.

Stability & robustness & performance & errors

Beside benchmarking using the test data as-is, we also need to assess the stability of the results given a changed symptom order or in a second run. We also need to record the run time for every case or possible error codes, hanging AIs and crashes without itself being compromised. Recording these details in a reliable and transparent way requires the benchmarking platform to perform a case-by-case testing rather than e.g., letting the AI batch-process a directory of input files.

Online mode

In constrast to other topic groups, TG-symptom benchmarking participants are mostly companies with real products. Rather than submitting AIs in docker containers that then run in a sandbox, the platform needs to support running the benchmarking against AI API endpoints hosted by the participants in their infrastructure. Note that his also includes changes to the batch processing to minimize the risk of cheating.

5.1.4.2 AICrowd

The general preliminary assessment is that AICrowd has the potential to serve as a benchmarking platform software for the first iteration of the benchmarking in our topic group. However, benchmarking and reporting is designed for one primary and one secondary score. Adding the high-dimensional scoring systems with reporting organized by a multitude of additional dimensions is not yet supported and needs to be implemented. This also applies to the automatic stability and robustness testing. The interactive dialog simulation needed for future benchmarking implementations would need to be implemented from scratch. In general, we found that the documentation for installing the software, the development process and for extending it is not as detailed and up to date as needed and the necessary changes would probably require close cooperation with the developers of the platform.

5.1.5 Scores & metrics

At the core of every AI benchmarking there are scores and metrics that assess the output of the different systems. In the context of the topic group the scores have to be chosen in a way that can facilitate decision making when it comes to deciding on possible solutions for a given health task in a given context.

5.1.5.1 Outline of clinical considerations for scores and metrics

The aim of this section is to outline the perspectives that must be considered for the development of clinically relevant metrics for benchmarking. It is crucial for clinical stakeholders that the outputs of benchmarking are able to answer questions that are relevant to clinical outcomes for decision makers.

A benchmarking framework should aim to assess some of these outcomes and this section will aim to identify:

- which aspects are important to evaluate AI systems in a health setting
- which of these are feasible to capture in benchmarking

- which may not and would be required to be captured in some other way (e.g., clinical studies)

Examples of stakeholders that might require clinically relevant metrics:

- Health system and governmental decision makers (including procurement, tendering, commissioning)
- Healthcare payers and providers
- Clinicians interacting with or using symptom assessment tools (either via patient facing self-assessment tools or clinician facing decision support tools)
- Regulators and notified bodies

It must be noted that clinical evaluation of AI tools in healthcare is already performed through clinical trials. The special, nuanced considerations over and above clinical studies for medical hardware, medications or surgical interventions have been carefully addressed in the CONSORT-AI reporting guidelines released in September 2020. In addition, as part of Software as a Medical Device Regulations (e.g., the FDA, EU MDR), it is a requirement that companies building these tools carry out Post Market Clinical Follow Up (PMCF) in order to demonstrate claimed clinical benefits really do occur in the real world.

Bearing this in mind, it is pertinent to discuss where the global initiative succeeding the focus group fits within the overall terrain of clinical evaluation.

Oversight

Whilst the benchmarking framework and clinical metrics are created as a result of collaborative efforts between various companies, there was also crucial involvement from a diverse range of independent stakeholders across the world (including practising clinicians, academics, technology experts and ethics experts). This process had oversight and input from the WHO and ITU via an independent Clinical Evaluation Working Group and Regulatory Working Group.

5.1.5.2 What is meant by clinical metrics?

These refer to measurements relevant to the stakeholders outlined above and could be split into:

- Performance and Accuracy measures
- Safety measures
- Clinical outcome measures

This list is not exhaustive and later other possible measures specific to AI systems in this modality will be amended. All of these should be addressed in the clinical evaluation of an AI system deployed into a healthcare system. The detail of exactly which metrics to be considered will be discussed further in this section, but it is important to outline the following key determinants:

- Intended use of the device
- Intended users
- Risk classification
- Point of Information and comparison to Standard of Care
- Intended/stated benefits to the user, clinical workflow and health system

For the purposes of benchmarking AI powered symptom assessment tools, it is important to consider clinical metrics within the context of the above categories and subsequently discuss which of these are then possible to actually measure through an independently curated, representative and high-quality test data set.

In modalities such as image classification, the inputs and outputs are quite clear compared to symptom assessment. The metrics to be considered for these tasks will be very different to those for symptom assessment tools. The next section outlines some of the key differences and special considerations for this modality.

5.1.5.3 Differences compared to image classification

Image classification of, for instance, a histological slide for identifying potentially cancerous cells will have a clear set of inputs and outputs. Inputs will be image pixel data and the outputs (cancerous vs non-cancerous, for instance) can be benchmarked compared to gold standards.

In comparison, symptom assessment tools will have a large range of information to convert to inputs. Examples might be:

- Symptoms - even the way these are captured might vary. Some may use structured text, others free text, others may also additionally have other augmenting ways to capture symptoms.
- Attributes (such as onset, character, location, intensity)
- Risk factors
- Medication
- Demographic information
- Location, region
- Seasonality (e.g., hay fever in summer, winter for certain respiratory viruses, malaria in rainy season)
- Increasingly other data points can be included as part of AI powered assessment tools (examples include wearables data,

Additionally, for an image classification task, the AI tool is provided with the image data it needs to perform the task on. In other words, it is given all the relevant information it needs to get to its output. For symptom assessment tools the performance at the task will be greatly affected by how effectively it collects and comprehends the information. As an example, it may be important to collect the critical information that somebody is pregnant. If the tool has not elicited this information, it may not consider this when providing condition suggestions. Clearly a balance must be reached - there could also be situations where too many questions are asked, resulting in a user not completing their assessment.

5.1.5.4 Important considerations

Before discussing metrics, there are some key nuances to consider:

The Ground truth problem

In order to derive metrics around performance, ground truth or gold standards must be established. There are different approaches to this, but examples, as well as their problems are outlined in the table below:

Table 7 - Ground truth approaches with their problems

Gold Standard Case Vignettes	Creation of vignette cases by clinicians. In these, the gold standard conditions and triage level can be defined by the author Examples in literature include Semigran et al [13]	These might be based on clinician experience or real cases they have seen and as such are subject to the clinician's own heuristics and biases. Require quality and peer review. Consensus on gold standards and disagreement/variability in opinion between clinicians is common. Clinician opinion is of lower certainty compared to a definitive imaging, lab or histopathology report (we have learned from our colleagues in these topic groups that even for histopath/imaging reports, there can be issues with gold standards owing to intra and inter-observer variability)
EHR records (retrospective)	Converting anonymised/pseudonymised electronic health record episodes into cases, with the coded condition as Gold standard.	There are many issues with using EHRs as 'gold standard'. Examples: • Coding issues (usually optimised for billing) • May include information gathered after the initial encounter with a clincian • Depend on whether the clinician actually documented info (for example important negative information) • The final diagnosis may occur later down the line (after the evolution of symptoms and time, as well as added data points e.g., labs imaging) • Geographic variability • Lack of standard structure

Defining metrics in the context of point of information

Scope of data

To create a safe and effective symptom checker, the data sets used in each symptom checker need to have defined and standardised limits with respect to information that can be asked of the user and to conditions that are provided to the user as possible outcomes. Symptoms are the most logical inclusion in a symptom checker, as the name would suggest; as features that can be described by the user in a typical medical history-taking exercise, they reasonably easily translate to a symptom-checking environment (notwithstanding the nuances of symptom descriptions). But while there has historically been great emphasis on the power of a well-taken medical history in determining diagnoses [38], other elements of a medical assessment add critical data that enhance the diagnostic process. To this end, the criteria for inclusion becomes more complex.

Elements of the past medical history and social history that are sought depend on the scope of the symptom checker. It is once again logical to ask for information that can affect the probability of possible outcomes, such as a history of smoking or co-morbid hypertension in a middle-aged male user using the symptom checker to assess his chest pain. Beyond this, however, there is significant uncertainty in the boundaries of information that can or should be asked. Should all users be asked, for example, if they live at home alone or if they live with somebody who can drive? This can influence whether a user is advised to call an ambulance or to go to the emergency department if they are having sudden-onset visual impairment. Similarly, should users be asked about the number of people that live in their household or more information about their socioeconomic status? This can influence whether an outcome of scabies is given to a user with itchy hands. Furthermore, specific information may be necessary depending on the features the symptom checker offers. For example, if a symptom checker offers social care services alongside their outcomes and/or triages, a more thorough social history would be relevant.

The place of physical examination in a symptom checker is also ambiguous. Should signs or physical examination findings be asked and if so, which ones? Ultimately, these elements of the medical assessment need to be reasonable with respect to what the user can find or self-evaluate with an untrained eye. For example, asking the user if they have pain on their ribs only with pressing the area in order to elicit the sign of rib tenderness is reasonable; asking them to perform the manoeuvre required to elicit Murphy's sign, however, is not. With the increasing use of wearables, it is also important to consider whether or what elements of self-monitoring should be included, such as temperature, blood pressure measurements or blood glucose readings. Despite the limitations of the wearables themselves, the use of this data could make outcomes more accurate, if available.

The question of which conditions are to be included in a symptom checker is somewhat more straightforward. The suite of conditions needs to be limited to conditions that could reasonably be diagnosed or suspected with the information that can be provided by a user through a symptom-checking tool. This entails the exclusion of those conditions which require clinician face-to-face contact, laboratory testing or clinical procedure to be diagnosed or suspected. While no condition can be diagnosed with 100% probability without a full and comprehensive medical assessment (and sometimes, not even then), many self-care conditions (e.g., viral colds, sinusitis, constipation) can, with the appropriate inclusion and exclusion of purely symptoms, be comfortably 'diagnosed' through a symptom checker - that is to say, suspected with a very high probability. With more complex conditions, particularly conditions requiring emergency treatment, uncommon or rare conditions or conditions requiring histological diagnosis or specialist management, the specification of 'suspicion' in a symptom checker becomes an important one. The inclusion of such conditions, ones that can be suspected but not diagnosed, is necessary for engendering trust in a symptom checker. An application which considers only self-care conditions or conditions requiring routine review is not particularly helpful. A symptom checker needs to have the ability to consider or rule out emergency or urgent conditions. Appendicitis, for example, can only be diagnosed with certainty through surgical exploration. However, every general practitioner would be expected to consider or suspect this diagnosis in a user with acute right lower quadrant abdominal pain. Alongside trust, the value of including such conditions is in directing users to appropriate actions and focusing clinical contact time. If appendicitis is considered in a consultation, depending on the likelihood of this condition based on the user's symptoms, this suspicion may come with a recommendation to attend the emergency department in the first instance to assess, via face-to-face contact with a clinician, whether surgical exploration is necessary, regardless of whether dysmenorrhoea is also a likely diagnosis.

Defining the scope

When considering which information is considered appropriate for a symptom checker, it is also important to consider who defines which information is considered appropriate. As there is no established gold standard, the most appropriate method of defining the boundaries of these metrics is to use a panel of clinicians to reach consensus. This is a method also used in clinical medicine for identifying diagnostic reference standards in the absence of gold standard tests [39]. The use of 'expert panels' for diagnosis is not without issues, such as intra- and inter-observer variability. It is therefore important to define a clear methodology for how consensus will be reached to minimise this variability and increase reproducibility as much as possible. Furthermore, the make-up of members on the panel is dependent on the focus of the symptom checker, so it needs to be considered whether the panel is limited to general practitioners or a mix of general practitioners and specialists or whether there are multiple panels appropriate for different conditions or presentations. Numbers of panel members also need to be decided (an odd number is most practical), as does the criteria for inclusion on a panel (e.g., area of expertise, number of years of experience). There is also the consideration of whether a panel of patients is necessary, particularly for defining the validity of user symptoms and the finer details relating to language and presentation of information.

Getting the comparison right

A useful source of data or point of comparison for this process of defining appropriate clinical metrics may be the use of telehealth consultations. As medical assessments which are undertaken with the barrier of a phone or screen, they, like symptom checker consultations, lack access to physical examination and investigations. They may therefore offer the most like-to-like comparison for defining clinical metrics in a symptom checker, though they are still limited by the variability of individual clinician choices of what to ask and record in a consultation. Audio or telephone rather than video consultations are likely the most useful of telehealth consultations, as they do not have the added visual information that video consultations can provide. However, there are still issues with telehealth consults which affect the reliability of their data. The ability of the clinician to hear the voice of the patient in the consultation provides some degree of clinical information not available in a symptom checker. A telehealth consultation is also more dynamic than a machine-powered consultation, allowing the user to spontaneously change their history, clarify mutual understanding or the clinician to switch to video as required. The data sets gleaned from telehealth consultations are also affected by specific patient populations that utilise telehealth more frequently (e.g., rural/remote populations) or contexts in which telehealth is used (e.g., the 2020 coronavirus pandemic).

Another consideration for a source of data sets or a point of comparison is electronic health record (EHR) data. The use of these data sets, however, is rife with issues. Information recorded in EHRs is incomplete and variable. It is also limited by what the clinician or writer has chosen to gather and document or believes is relevant and may not include all questions asked of and all information given by, a patient in order to develop a complete picture of the possible diagnoses. The language used in EHRs is often heavily medical in nature, acting as 'translations' of patient histories and are also reflective of the writer's training, ethnicity, coding requirements and/or practices of the institution in which they work. In addition, the "true condition" taken as gold standard might have been arrived at through a combination of taking a history, clinical exam, bedside tests and potentially lots of other tests, so as discussed, it does not serve as a perfect like for like comparison to a patient facing symptom assessment tool.

Performance does not equal utility

Whilst it is important to consider the performance of AI based symptom checkers as part of evaluation, another key aspect in clinical evaluation is that of utility or impact. An AI tool may claim to have a 99% sensitivity, but clinical stakeholders are also considered with additional factors. In particular, what the impact is in a clinical setting. These clinical outcomes might be considered at patient level, clinician level (how does it impact clinical workflow) or health system level.

Different metrics for different use cases/contexts

Symptom assessment tools are not one homogenous group of tools. There are numerous variations on intended use, intended users, locations, populations, etc. Some might be specific for a certain region or population, others more general. They may also perform varying tasks (e.g., taking structured inputs vs free text) and be geared optimally towards their particular intended use case.

With this in mind, there need to be context-relevant metrics. Current discussions in the topic group centre around developing the ability to drill down into different contexts and use cases. For example, a stakeholder might be a health ministry in a certain region using the ITU/WHO benchmarking metrics to assess potential partners. They may be more interested in viewing specific metrics (or metrics from a specific sample of the independent data set) that are relevant to the setting in which they want to deploy an AI symptom assessment tool.

Some examples of contextual variations include:

- Geography/region/seasonality (important to also note that there are large demographic variances within countries and within cities)
- Population demographics – e.g., age groups, subpopulations, biological sex, language
- Health literacy
- Digital literacy
- Focus (via intended use) on specific medical specialties (e.g., Pediatrics or Muscoloskeletal medicine)

5.1.6 Metrics for symptom assessment

As indicated previously, these can be looked at with respect to:

- Performance and Safety measures (How accurate is this device? How safe is this device?)
- Clinical outcome and income measures (How does it impact clinical practice - for patients, clinician workflow or health systems)

Currently, the key outputs of patient-facing symptom assessment are:

- **Condition suggestions:** Some tools provide an ordered list of possible conditions (with varying nomenclature, the most common being differential diagnosis), others have an unordered list of possibilities. Another variable between tools[2] is the indication of probabilities for each suggestion. Sometimes informally any of this is called a "[differential] diagnosis" but it is only an informal name due to the fact that symptom assessment apps generally don't have enough accuracy (yet); also due to liability considerations and due to important safety & regulatory reasons.
- **Pre-clinical triage:** This gives advice about what level of care the user should seek. There is a large variance between tools in the exact levels, which can range from 'self-care' to "call ambulance". The triage advice might be given as direct advice or just as "information" (the

[2] Through this section, the terms tool and app are used interchangeably and refer to patient facing symptom assessment tools.

choice depends on the accuracy of a tool and the estimation of that accuracy by an app provider; by the amount of the liability that an app provider would like to take; on regulations requirements in a particular country; etc.).

There are other tasks that require their own metrics:

- Quality of information gathering (how well does the tool collect the required information)
- Safety of information gathering (does the tool consider relevant serious symptoms and ask them)
- Tools that assimilate free text also require measurement of the accuracy with which this use input is converted into a list of symptoms or other information that the tool can make use of.

5.1.7 Performance and accuracy

5.1.7.1 Condition suggestions

When measuring performance, traditional metrics in healthcare are focused around diagnostic accuracy. "How did the test predict the diagnosis of a condition compared to the Gold standard?"

At present, published clinical studies of AI based symptom checkers focus on assessing this by comparing the following to the Gold standard.

- Top matching condition (i.e. did the top condition suggested by the AI tool match to the gold standard?)
- Top 3 or 5 matching condition (i.e. was the condition defined in the gold standard present in the top 3 or top 5 of the list of suggestions of the AI tool?)

Whilst these measures are a good starting point, some important nuance must be considered.

- The top match condition metric assumes that it is always possible to get to the "final diagnosis" indicated in the gold standard from the inputs available to a symptom assessment tool. In other words, it assumes that taking the information that is available from a medical history will be enough to accurately predict what the patient has in future. This has been discussed further in the Point of Information section. Much depends on the stated intended use of the tool, but in general, at the time of writing, it is not a goal (and it simply can't be a goal) to give a diagnosis. Aside from the most clear-cut cases, the process of diagnosis requires so much more information: observing the evolution over time, clinical examination, bedside tests, lab and imaging tests.
- The top 3 or top 5 condition matching metric is useful as it starts looking at the list of conditions suggested by the AI tool. However, there is no measure of how good the other suggestions on the list are.

This leads us to discuss the merits of metrics that aim to measure the quality of the entire list of differentials provided by an AI tool, i.e. a differential diagnosis, since even the most astute clinicians can't always give a certain, precise diagnosis for a patient without further tests (e.g., X-rays, blood tests, etc.) in many cases. This means that when we measure accuracy of the "diagnostic" capabilities of symptom assessment apps, we always need to keep in mind that we are always evaluating their results for a particular patient case against a "ground truth" (i.e. the real, objective one) of differential diagnosis distribution of possible conditions. For example, for a relatively young patient who has a heart attack their differential diagnosis might look as follows: 15% heart attack, 75% panic attack and around 10% for other conditions. Note that when collecting this information from clinicians, we most likely need to approximate those probabilities or even just consider orders of likelihoods without assigning numerical estimates.

For any metric evaluation, it is also important to define what exactly is being evaluated. For example, what exactly are we trying to assess: Is it a new symptomatic condition(s)? A new symptomatic or asymptomatic condition(s)? Should it include flares of existing conditions? Should it include acute presentations of chronic conditions? Etc.

5.1.7.2 The presence of more than one condition (multi-morbidity)

When we mentioned above a distribution over the differential diagnosis for a particular patient, we often assume that a patient has generally only one of those conditions (i.e. a condition "of interest" or the "ground truth" condition). However, while less likely it is often possible that a patient has multiple conditions "of interest" (or "ground truth" conditions) at the same time (e.g., two conditions which are both new) and this affects the shape of the "ground truth" differential diagnosis distribution for the patient (e.g., for a patient who has a whiplash and a dislocation of shoulder after a traumatic car accident, the differential diagnosis distribution might look like this: 85% whiplash; 90% dislocation of shoulder; and some other conditions with other probabilities that are similar to whiplash or/and dislocation of shoulder).

5.1.7.3 Similar presentations, varying outcomes

It is possible that very similar constellations of symptoms have variable outcomes. As a very simplified illustrative example, if there is a cohort of 100 people (female) aged 25-30 who present to a GP in a very similar way, say, right lower abdominal pain, fever and mild dysuria for 3 days and followed them up 1 month later there would be a natural variation in what they ended up having. X% might have a urinary tract infection, Y% might have appendicitis, Z% might have pyelonephritis, an even smaller proportion might have an ectopic pregnancy. This happens even though the "inputs" captured at that first point of information were very similar. If a symptom assessment tool is providing condition outputs accompanied by probabilities for their differential diagnosis lists/suggestions, it may be an important consideration to include metrics that assess whether these distributions match real world/gold standards.

Some conditions have pathognomonic symptoms or signs - i.e. very strong indicators of a particular condition. But in reality, for most conditions it is not so simple - there is much more uncertainty. One of the main roles of those in primary or emergency care is to manage this uncertainty.

Also, note that a symptom assessment tool might predict a chance/probability for many different conditions (often for hundreds of them) and that is quite different from the settings of "diagnostic tests" which often just need to determine whether a patient has or does not have one or few particular conditions.

5.1.7.4 Basic metrics for differential diagnosis

There may be several ways to calculate these metrics for symptom assessment tools. Let us take a patient with a specific presentation (including: symptoms, if any; their medical history; etc.) and the "ground truth" that he/she has e.g., new presentations of conditions $\{X_1, \ldots, X_n\}$ (where n is likely to be equal to or less than 1) and does not have any other conditions. Let's say that a symptom assessment tool, after collecting all information it could collect from a patient, has identified that a patient might have some conditions $\{Z_1, \ldots, Z_m\}$ (for simplicity, let's assume that the app just says whether each condition is present or not; generally, it might return some likelihood/probability of it or some other degree of certainty). Any metric we calculate might be influenced by the outcome types of the tool. As mentioned, some assessment tools return ordered lists of conditions with probabilities; and others might just return ordered lists or even unordered lists. This means that only the top matching condition (top-N) metric might be calculated only for some of the tools. The

following metrics can be calculated per patient case (and then aggregated later) for a specific symptom assessment tool/app:

Table 8 – Overview patient case metrics

Recall (also called: true positive rate, sensitivity)	This is the ratio of conditions "of interest" that the patient has and which were identified by the app (i.e. presumed by the app to be happening to the patient) to the number of the conditions that the patient has. Note that in the case of a "simple" "one-condition diagnostic test" for a specific condition, the recall is much "simpler" and is usually calculated in an aggregated way across all patient cases: it is the ratio of sick people (i.e. people with that specific condition) correctly identified as sick (i.e. presumed to have that specific condition) to the number of sick people (i.e. having that specific condition).
Precision **(also called: positive predictive value)**	This is the ratio of conditions that the patient "of interest" has and which were identified by the app to the number of the conditions that the app has identified.
F1-score and Fn-score	This is the harmonic mean of precision and recall. In Fn-score, recall is considered n times as important as precision (n > 0).
Specificity **(also called: selectivity; true negative rate)**	This is the ratio of the conditions "of interest" that the patient does not have and which were not flagged (i.e. were not highlighted as present/"likely") by the app to the conditions which the patient does not have. Note: • False positive rate (also called: fall-out or false alarm ratio) can be calculated as follows: 1.0 - specificity. • Since there are quite a lot of conditions that a patient might have in general and since symptom assessment apps generally can rule out most of conditions that a patient does not have, specificity might often be close to 1.0. Because of that, a receiver operating characteristic curve (ROC curve) (calculated over many cases, not just for one case) that is created by plotting the recall (sensitivity) against the false positive rate (i.e. 1.0 - specificity) might look almost "trivial" in many cases since the false positive rate might often be close to 0.0.
Accuracy	Different things might be meant by "accuracy" and there are multiple ways to define "accuracy". Informally, different metrics can be called "accuracy". Because of this, it is recommended to always clarify what is meant by "accuracy" if this term is used.
Top-N **(Top matching condition)**	One of the simple ways to calculate it is this: for each case top-N is equal to 1.0 if an app's output (i.e. a "differential diagnosis") contains the condition of "interest" in its top N conditions in the output (assuming that the app returns ordered conditions). If there are M conditions of "interest" for the case and K of them are in the top-N conditions from an app's output, then top-N for the case for the app is equal to K/M. Note that an app might return fewer than N conditions in its output for a case: in this scenario, if an app returns J < N conditions, it might be assumed that top-N is equal to top-J for this particular case for this app (note that J might be equal to 0).

	Note that if an app provides an unordered list of conditions, then it is not possible/trivial to calculate Top-N.

There are at least 3 approaches to elicit conditions of "interest" for a case:

1) For each case condition(s) of "interest" is/are provided by a creator of the case or by an independent clinician (or a panel) or it comes from an EHR where we are certain about the diagnosis (i.e. exact condition(s)) of a patient.
2) Another option is to ask a clinician or a panel of clinics to provide a "differential diagnosis" (with multiple conditions) in some form, to which apps' outcomes will be compared.
3) Ultimately, we might want to naturally obtain a distribution over a differential diagnosis for very similar patients with very similar symptoms and risk factors. This is exactly the sought differential diagnosis distribution, to which we can compare apps' outcomes. However, this requires a lot of data and to the best of our knowledge many existing EHR systems might not be suitable for this due to many factors (due to their size/record format/"accuracy"/etc.). Also, it might be very hard to scale this approach internationally for different regions (e.g., due to the variability in how EHR systems are implemented and used).

(Note that in the 2nd and 3rd approach we receive a distribution over condition(s) of "interest".)

It should also be noted that there are always special cases: some patients might be healthy or they might have a condition that is not known by an app and so for the purpose of the metric calculation there might be zero conditions of "interest" that a patient has. Such situations might cause recall to be ill-defined. Also, an app might return an outcome with no conditions at all (i.e. it might assume that a patient does not have any conditions at all, at least of those that it is aware of) and in this case the precision might be ill-defined. Special rules must be applied for such cases as appropriate.

The metrics mentioned above might be calculated for each patient case. They can then be aggregated, e.g., by taking an average. A weighted average can be used (e.g., by weights associated with the severity of conditions, by epidemiological rates associated with the condition or by some other weights to balance patient cases and/or avoid bias; etc.). Also, note that another alternative is to treat all patients and all conditions as separate (but correlated) random variables such that e.g., each pair of a patient-condition (e.g., a patient X has a condition Y) is a separate variable (e.g., a Boolean one) and then the metrics might be calculated for all of them at the same time in one batch. One more alternative is to treat each patient-condition pair as a separate variable but instead of treating them as one batch, consider patients for each disease independently (in some sense, this is equivalent to treating a symptom assessment app as a set of independent one-disease "diagnostic tests"). There are other alternatives as well.

A curve of recall and precision could then be plotted over multiple test cases. (The area under such a curve might also be calculated.) There are multiple ways to calculate such a curve, similarly to how there are multiple ways (as discussed above) to calculate e.g., aggregate metrics for recall and precision.

Metrics such as precision and recall measure presence in the differential diagnosis list, but they cannot tell if differentials are returned with appropriate likelihood nor if they are returned in the right order (for instance with decreasing level of confidence).

To address this, Normalised Discounted Cumulative Gain (NDCG) is an example of a metric that gives a measure of ranking quality. Each item (that is a disease) has an assigned relevance. In the setting of a differential diagnosis list, relevance should be proportional to confidence in disease - more probable diseases should have higher confidence. If items with high relevance are in the top of ranking the score will be high. In the prior mentioned example of a young man with chest pain:

- 15% heart attack - relevance “medium”
- 75% panic attack - relevance “high”

(and around 10% for other conditions - we might give some conditions label “low”)

Note that this proposition measures only ability to return diseases from the most probable one to least probable. However, from a clinical perspective returning “heart attack” might be as relevant (and important) as “panic attack” because of its potential seriousness.

5.1.7.5 Basic metrics for triage

For triage, it might be of interest to measure whether the triage recommendation returned by an app is suitable for a particular patient case. There are two approaches to consider triage in this context. The first is to consider triage outcomes based on the seriousness of conditions that are present within the possible conditions. The other approach is the presence of serious symptoms within an assessment. In reality, clinicians will use a combination of both of these to settle on the most appropriate outcome. To benchmark this, it may be necessary to have an externally defined set of serious conditions and red flag symptoms.

One way to measure triage performance is to have one “perfect”, “ground truth” triage option for each patient case (provided by an expert clinician/panel of experts) and to match it with a tool’s triage, such that there is a match or no match. This can be averaged (and weighted if appropriate similarly to the differential diagnosis as described above) across multiple patient cases. This way we capture the “accuracy” of triage. However, this approach has limitations because:

a. there might be multiple appropriate triages for the same patient (especially if different experts believe some similar but different triage options are appropriate) and
b. some triages are safe but overcautious e.g., if a patient needs to see a primary care doctor but he/she is directed by an app to a hospital urgently, then the tool’s decision is safe but not accurate. This also has the potential to overburden health systems inappropriately.

Hence, the ground truth for a particular patient case might consist of a set (or a range if we assume (partial) ordering of triages) of appropriate triages rather than just one triage option. This set can also be separated into at least two subsets:

1. a subset of triage options that are safe and not overcautious and
2. a subset of triage options that are safe but overcautious.

If an assessment tool returns a triage outcome from any of two subsets, it could be deemed a safe decision. However, if the tool returns a triage outcome that it is in the second subset, then it could be deemed a safe but overcautious triage. For example, for a condition for which it is okay (safe) to see a primary care doctor in a few weeks, it is most probably safe but overcautious to go to a hospital or even go there by an ambulance.

With this separation, triage outcome can be measured with e.g., two metrics: (a) how safe it is; (b) how safe and non-overcautious it is? The latter might be called “accuracy” of triage. There are obviously other variations over these metrics.

Note that if there is some full or partial ordering of triage outcomes, then the two subsets mentioned above might be simplified: e.g., in the case of a full ordering, two threshold triage outcomes might be needed: e.g., one to define the “minimum safe triage outcome” and the “minimum safe not overcautious triage outcome”.

A special case scenario for an assessment tool might be when it does not return any triage outcomes or any explicit recommendation. This is a special type of a triage outcome and depending on the particular message that is returned by an app in such cases, it either should be treated as one of the default outcomes (e.g., if anyone is advised to see a doctor “quite urgently” in such cases by an app, such an outcome could be mapped to an urgent primary care appointment) or it should be treated as a separate category of triage outcomes for the benchmark purposes (and hence probably reported and analysed semi-independently which might involve additional complexity for report generation especially when they are aggregated for different apps).

It is a significant challenge to standardise and map triage outcomes (of different tools) to one particular set, especially internationally. This is because of local/contextual variation. Options for care in a remote village in one country are very different to those in an affluent neighbourhood in a large city in the same or other country, for instance.

It also needs to be considered that there are apps that return only information and do not provide any explicit triage outcomes. In addition overall triage metrics, e.g., triage safety, might be ultimately not that useful. For example, if a patient who needs to see a primary care doctor urgently is triaged by an app to see a primary care doctor non-urgently, this is not safe, but it is probably still less “unsafe” than advising that patient to self-care (e.g., “stay at home and keep hydrated”, even without going to a pharmacist who has medical knowledge).

Hence, some additional stratification/analysis of triage outcomes is important. One more metric can be relative/absolute confusion matrices where e.g., one dimension contains “expected” triages and another dimension contains triages provided by an app. In addition to this, different triage combinations (e.g., pairs of an expected and provided triage) can have different weight.

5.1.7.6 About statistical significance of metrics

In addition to providing statistics on the performance of symptoms checkers undergoing benchmarking, the designers of a benchmarking system should also consider how to assess the statistical significance of differences in results for different symptom checkers or for the same symptom checker evaluated for different domains. For example, is the combination of the magnitude of the difference in triage accuracies of two symptom checkers and the number of cases in the test set sufficient to conclude that one performs better than the other for triage or is it probable that the order could be reversed if the two symptom checkers were evaluated on a different test set? This consideration will be especially pertinent when evaluating symptom checkers on cases from a limited domain of conditions or a small population, as this is likely to lead to comparisons based on a small number of test cases.

5.1.7.7 “Ground truth” for differential diagnosis/triage

Evaluating the differential diagnosis and triage provided by a symptom checker will require a "ground truth" - the known correct response for each test case. If clinicians were always in agreement with 100% accuracy, this would be a straightforward process. However, one study [35] found during the process of creating vignettes that a panel of six experienced physicians all agreed on correct triage for only 35% of vignettes, with the individual agreement of each physician with the consensus triage of all six ranging from 69% to 92%. Hence the expected condition and triage attached to a case will represent an estimate of the true values, with an accuracy determined by the method used to combine different opinions into a consensus.

Increasing the accuracy of differential diagnosis and triage for each individual vignette is likely to require increased input from clinicians, either through increased discussion in cases of disagreements or through the use of larger panels of clinicians to review each vignette. With a finite amount of time available from clinicians, this will create a trade-off *where* reducing errors in individual vignettes by assigning more time to the creation of each case will reduce the number of cases that can be annotated for a given test set, increasing the statistical significance of any remaining errors. Two approaches to combining multiple approximations of ground truth from different experts should be considered.

The first is to average the individual contributions. For triage, this could simply be by taking the median triage level - so a panel of four clinicians in which three clinicians assign a triage level of urgent care and one a triage level of emergency care would result in a final level of urgent care, as would two clinicians assigning a triage level of emergent care and two assigning a level of possible self care. For conditions, this could be handled by including a set of weighted diagnoses to the output, e.g., if three clinicians in a panel selected condition A and the remaining member selected condition B, the weights for A and B would by 75% and 25%. A symptom checker suggesting only condition A in its top-N differential diagnoses would score 75% for that case, while a symptom checker suggesting A and B would score 100%. This will introduce a maximum achievable score of below 100% for top-N diagnoses where N is lower than the maximum number of conditions attached to any vignette.

The second approach to consider is a panel consensus. Methodologies such as the Delphi panel [40] can provide consensus in the face of uncertainty. This will give a single definite ground truth differential diagnosis and triage for each vignette, but at the cost of masking any uncertainty arising among the panel.

5.1.7.8 On balancing, biasing and weighting

In order to capture the full spectrum of conditions, collections of vignette test cases will contain multiple vignettes covering conditions with extremely low incidence, to allow for the testing of symptom checkers on differing presentations of conditions that in the real world clinicians will encounter very rarely. With the incidence of conditions differing by orders of magnitude, it will not be practical to create sufficient vignettes for more common conditions to create test sets with realistic condition distributions. This will lead to the creation of unbalanced test sets, where the a symptom checker performing well on common conditions but poorly on rare conditions will appear to be less accurate than would be expected on a real-world sample of users suffering from a realistic distribution of conditions.

Existing clinical vignettes disproportionately cover conditions and populations fonud in North America and Western Europe. This will lead to a bias in test cases and therefore results, with symptom checkers able to achieve high accuracy scores even with very low accuracy for vignettes covering other conditions and populations.

These two problems of bias and balance can both be alleviated by weighting cases based on the condition for which they were created. These weightings could be created by dividing the incidence or prevalence of a condition by the number of vignettes created for it, so that the total weightings for all vignettes for a condition will be proportional to the likelihood of encountering that condition.

5.1.7.9 Other performance measures

The parameters discussed so far relate to the performance of the outputs of symptom assessment tools. Clinical stakeholders are also concerned with the performance of how data is collected and interpreted by the tool. With this in mind, other aspects to include in overall performance metrics would be:

- Quality of question flow: How much of the relevant positive and negative evidence was actually elicited? Were there any key serious symptoms (red flags) not elicited by the tool?
- Do users actually understand/comprehend the questions? (this may be difficult to assess within benchmarking. It might be captured within usability testing)
- Measuring the task of converting the lived experience of the user into information that the tool can make use of. Different tools use different methods for this - some use Natural Language Processing, for example. It is important to have metrics on the performance/accuracy of this task.
- Metrics to consider that there are certain conditions that may be explicitly ruled out due to the age or the biological sex of the user. An example of this would be pregnancy related conditions (e.g.,pre-eclampsia) should usually not be included in the list of differentials for someone whose biological sex is male3)

Clinical Outcomes/Impact

Metrics measured here will relate to the stated clinical benefits of the AI tool. Mostly these are best measured with a study in a clinical setting. Examples of clinical outcomes and impact measurements could be:

Patient Journey

- Effect on patient journey - satisfaction with navigating health services after being given triage advice
- Increase/decrease in time spent waiting for appointments for conditions that can be helped with self-care or other healthcare pathway (e.g.,pharmacy)
- Effect on waiting times for appointments
- PROMs (Patient Reported Outcome Measures)

Clinical Workflow

- Effect on consultation times between patient and HCP
- Effect on clinician caseload management

[3] *It is also important to consider this within the context of people who have congenital sex differentiation disorders, where this notion may not be so clear cut for some users.*

Health System

- Effect on demand on emergency services
- Effect on demand for primary care appointments
- Effects and impact of under-triage (e.g.,advising people with a serious condition to self-care)
- Effects and impact of over-triage (e.g.,inappropriately advising someone with a non-urgent condition to attend emergency department)

In the future, it may be of interest to work with health economists to explore the utility of health economic metrics within benchmarking that might be a function of cost, QALYs, etc. These may be important for health system level stakeholders (e.g.,health ministries, providers, payers).

5.1.8 Putting it all together for clinicians

A big challenge is communicating this range of metrics (relevant to context) to clinicians and other healthcare stakeholders, in a way that aids understanding. A paper published in Nature in March 2020 by Sendak et al [41] explores this in great detail and provides a good example inspired by the nutrition information on a cereal box26. Benchmarking metrics summarised in a such a digestible way that takes the most relevant parts is worth seriously considering.

Model Facts **Model name:** Deep Sepsis **Locale:** Duke University Hospital

Approval Date: 09/22/2019 **Last Update:** 01/13/2020 **Version:** 1.0

Summary

This model uses EHR input data collected from a patient's current inpatient encounter to estimate the probability that the patient will meet sepsis criteria within the next 4 hours. It was developed in 2016-2019 by the Duke Institute for Health Innovation. The model was licensed to Cohere Med in July 2019.

Mechanism

- **Outcome** .....sepsis within the next 4 hours, see outcome definition in "Other Information"
- **Output** .....0% - 100% probability of sepsis occurring in the next 4 hours
- **Target population** .....all adult patients >18 y.o. presenting to DUH ED
- **Time of prediction** .....every hour of a patient's encounter
- **Input data source**.....electronic health record (EHR)
- **Input data type** .....demographics, analytes, vitals, medication administrations
- **Training data location and time-period** .....DUH, diagnostic cohort, 10/2014 – 12/2015
- **Model type**..... Recurrent Neural Network

Validation and performance

	Prevalence	AUC	PPV @ Sensitivity of 60%	Sensitivity @ PPV of 20%	Cohort Type	Cohort URL / DOI
Local Retrospective	18.9%	0.88	0.14	0.50	Diagnostic	arxiv.org/abs/1708.05894
Local Temporal	6.4%	0.94	0.20	0.66	Diagnostic	jmir.org/preprint/15182
Local Prospective	**TBD**	**TBD**	**TBD**	**TBD**	**TBD**	**TBD**
External	**TBD**	**TBD**	**TBD**	**TBD**	**TBD**	**TBD**
Target Population	6.4%	0.94	0.20	0.66	Diagnostic	jmir.org/preprint/15182

Uses and directions

- **Benefits:** Early identification and prompt treatment of sepsis can improve patient morbidity and mortality.
- **Target population and use case:** Every hour, data is pulled from the EHR to calculate risk of sepsis for every patient at the DUH ED. A rapid response team nurse reviews every high-risk patient with a physician in the ED to confirm whether or not to initiate treatment for sepsis.
- **General use:** This model is intended to be used to by clinicians to identify patients for further assessment for sepsis. The model is not a diagnostic for sepsis and is not meant to guide or drive clinical care. This model is intended to complement other pieces of patient information related to sepsis as well as a physical evaluation to determine the need for sepsis treatment.
- **Appropriate decision support:** The model identifies patient X as at a high risk of sepsis. A rapid response team nurse discusses the patient with the ED physician caring for the patient and they agree the patient does not require treatment for sepsis.
- **Before using this model:** Test the model retrospectively and prospectively on a diagnostic cohort that reflects the target population that the model will be used upon to confirm validity of the model within a local setting.
- **Safety and efficacy evaluation:** Analysis of data from clinical trial (NCT03655626) is underway. Preliminary data shows rapid response team, nurse-driven workflow was effective at improving sepsis treatment bundle compliance.

Warnings

- **Risks:** Even if used appropriately, clinicians using this model can misdiagnose sepsis. Delays in a sepsis diagnosis can lead to morbidity and mortality. Patients who are incorrectly treated for sepsis can be exposed to risks associated with unnecessary antibiotics and intravenous fluids.
- **Inappropriate Settings:** This model was not trained or evaluated on patients receiving care in the ICU. Do not use this model in the ICU setting without further evaluation. This model was trained to identify the first episode of sepsis during an inpatient encounter. Do not use this model after an initial sepsis episode without further evaluation.
- **Clinical Rationale:** The model is not interpretable and does not provide rationale for high risk scores. Clinical end users are expected to place model output in context with other clinical information to make final determination of diagnosis.
- **Inappropriate decision support:** This model may not be accurate outside of the target population, primarily adults in the non-ICU setting. This model is not a diagnostic and is not designed to guide clinical diagnosis and treatment for sepsis.
- **Generalizability:** This model was primarily evaluated within the local setting of Duke University Hospital. Do not use this model in an external setting without further evaluation.
- **Discontinue use if:** Clinical staff raise concerns about utility of the model for the indicated use case or large, systematic changes occur at the data level that necessitates re-training of the model.

Other information:

- **Outcome Definition:** https://doi.org/10.1101/648907
- **Related model:** http://doi.org/10.1001/jama.2016.0288
- **Model development & validation:** arxiv.org/abs/1708.05894
- **Model implementation**: jmir.org/preprint/15182
- **Clinical trial:** clinicaltrials.gov/ct2/show/NCT03655626
- **Clinical impact evaluation:** TBD
- **For inquiries and additional information:** please email mark.sendak@duke.edu

Figure 1 – Example "Model Facts" label for sepsis machine learning model from Sendak et al, 2020. (Nature)

5.1.9 Additional clinical considerations and limitations

Whilst some important considerations have already been outlined, there are some additional discussions that relate indirectly to clinical metrics.

Mapping Ontologies

Another problem to address is one of 'mapping'. This refers to variability in nomenclature and ontologies of symptoms and conditions. Each symptom assessment tool might have slightly different names for certain conditions. An example might be "heart attack". Common synonyms could be "myocardial infarction" or "acute coronary syndrome". A gold standard case may have defined the true condition to be either one of those. In addition, there is not one standard ontology. A robust, reliable and trusted approach is required to map these to a common, agreed ontology. In any case any ontology mapping is likey to produce some mapping friction that will influence the metrics and need to be considered when interpreting benchmarking results.

Explainability

Discussions about AI in healthcare naturally arrive at this aspect. It is important to clinicians and decision makers that, in many cases, that the reasons for outputs from AI tools are understood. There is concern that purely black box AI tools that give outputs that are 'blindly followed' in a clinical context could have detrimental effects. There is an example of an AI system being trained to detect Covid-19 related changes on chest x-rays in emergency departments. Whilst initial results appeared positive (even on external test images) it was discovered that the system was using other artefacts within the image to determine Covid-19 presence.

Coming back to symptom assessment tools, explainability might relate to the presence of communicated messages to users to justify certain outputs. For example - if a triage outcome is 'Seek emergency care', there may be an explanation to the user as to what led to that outcome. In terms of metrics, this could be measured as a) the presence of a justification as well as b) its quality and accuracy but a concrete, widely accepted, quantifiable measure of explainability does not currently exist.

Addressing clinician variability and the ground truth problem going forward

Within the topic group, there has been fervent debate about the issues with EHRs and clinician variability in defining gold standards or ground truth. Another approach that has been discussed as an enhancement to the current benchmarking framework is as follows.

Benchmarking could involve each AI tool being paired with clinical sites across the globe that are relevant to its intended use and users. Connecting anonymised data of the patients that come through over a period of time, the AI systems are given the symptom information of the patients using the service. The outputs are then compared to the outputs at the 'end of the episode" – i.e. after the diagnostic process is completed. The "final" (for that episode) triage and/or differential diagnosis in the real world is then compared to the AI tool's outcome. This has the advantages of being 'real world' test data and reduces the problems of clinician variability. However, examples of some challenges to overcome are: data security, reaching a critical mass of participating sites, standardisation of processes to achieve fair, comparable benchmarking tests across all sites.

5.1.10 Conclusion

This section has highlighted the myriad complexities, challenges and considerations that need to be addressed and applied for the successful adoption and implementation of symptom assessment tools. Benchmarking can capture and answer some of the questions relevant for clinical stakeholders, but it is important to acknowledge the limitations as discussed. What this should lead to is a pragmatic approach that brings alignment about what clinically questions can be answered for stakeholders within benchmarking and which questions may need to be answered in other ways (e.g., robust prospective clinical studies).

6 Benchmarking by the topic group

This section describes all technical and operational details regarding the benchmarking process for the AI-based symptom assessment including subsections for each version of the benchmarking that is iteratively improved over time. It reflects the considerations of various deliverables: DEL5 *"Data specification"* (introduction to deliverables 5.1-5.6), DEL5.1 *"Data requirements"* (which lists acceptance criteria for data submitted to FG-AI4H and states the governing principles and rules), DEL5.2 *"Data acquisition"*, DEL5.3 *"Data annotation specification"*, DEL5.4 *"Training and test data specification"* (which provides a systematic way of preparing technical requirement specifications for data sets used in training and testing of AI models), DEL5.5 *"Data handling"* (which outlines how data will be handled once they are accepted), DEL5.6 *"Data sharing practices"* (which provides an overview of the existing best practices for sharing health-related data based on distributed and federated environments, including the requirement to enable secure data sharing and addressing issues of data governance), DEL06 *"AI training best practices specification"* (which reviews best practices for proper AI model training and guidelines for model reporting), DEL7 *"AI for health evaluation considerations"* (which discusses the validation and evaluation of AI for health models and considers requirements for a benchmarking platform), DEL7.1 *"AI4H evaluation process description"* (which provides an overview of the state of the art of AI evaluation principles and methods and serves as an initiator for the evaluation process of AI for health), DEL7.2 *"AI technical test specification"* (which specifies how an AI can and should be tested *in silico*), DEL7.3 *"Data and artificial intelligence assessment methods (DAISAM)"* (which provides the reference collection of WG-DAISAM on assessment methods of data and AI quality evaluation), DEL7.4 *"Clinical Evaluation of AI for health"* (which outlines the current best practices and outstanding issues related to clinical evaluation of AI models for health), DEL7.5 *"FG-AI4H assessment platform"* (which explores assessment platform options that can be used to evaluate AI for health for the different topic groups), DEL9 *"AI for health applications and platforms"* (which introduces specific considerations of the benchmarking of mobile- and cloud-based AI applications in health), DEL9.1 *"Mobile based AI applications,"* and DEL9.2 *"Cloud-based AI applications"* (which describe specific requirements for the development, testing and benchmarking of mobile- and cloud-based AI applications).

6.1 Benchmarking self-assessment systems

The main goal of this section is to outline the automatic benchmarking of AI-based symptom assessment systems in way that allows any interested developer of such systems to participate in the benchmarking. Previously benchmarking was done manually by humans entering cases into systems in a one-time effort. The main reasons for that was that there was no standardized approach of expressing cases in a way that all symptom assessment AIs understand without human interaction. An important focus of the TG-symptom benchmarking initiative was therefore agreeing on a way for encoding cases across companies. Due to business constraints the topic group decided on an approach of stepwise increasing the complexity of the benchmarking until a "real" benchmarking

can be conducted. This first version that will enable a real benchmarking of production AI systems with viable results that will allow stakeholders to make decisions, is referred to as “minimal viable benchmarking” “MVB”. The different steps towards this first version are referred to as “MMVB x.y” where the first “M” stands for “minimal”.

Table 9 – Benchmarking iterations

Version	Focus/Goals
MMVB 1.0	● show a complete benchmarking pipeline including case generation, AI, metrics, reports ● with all parts visible to everyone so that we can all understand how to proceed with relevant details for MVB ● learn about the needed data structures and scores ● write/test some first case annotations guidelines ● learn about the cooperation on both software and annotation guidelines ● have a foundation for further discussions on if an own benchmarking software is needed or CrowdAI could be used
MMVB 2.0	● extend the MMVB model to attributes ● refine the MMVB factor model ● switch to cloud-based toy AI hosting ● test one-case-at-a-time testing
MMVB 2.1	● a new dedicated benchmarking frontend ● a new backend infrastructure ● a first simple case annotation tool
MMVB 2.2	● full implementation of the Berlin model in frontend, backend and annotation tool ● improve AI error handling / health check ● improved usability of the frontend
MMVB 3.0	● using the benchmarking platform of the OCI ● using the same AIs and cases as MMVB 2.2
MMVB 3.1	● an extension of MMVB 3.0 using FHIR encoded cases by automatically transforming the MMVB 2.2 case vignettes into FHIR and implementing a FHIR/SNOMED mapping wrapper arounds one of the AIs as prove of concept

Table 9 lists the different iterations. The final MVB has not yet been reached and requires further research as part of the Global Initiative. It is also expected that for the MVB we will change the approach from manual SNOMED encoded cases to FHIR encoded free text case vignettes and assume that the AI developers could easily use language models to map from free text to their own ontology (if needed). The mapping friction of humans mapping patient friendly cases to SNOMED plus the mapping friction from mapping SNOMED to their own ontologies by the companies is expected to be lower than using GPTs for directly mapping from free text to the AI developers ontology. It will also simplify the participation of purely text-based AI systems as they became widely available early 2023.

6.1.1 Benchmarking version MMVB 1.0

6.1.1.1 Overview

The first benchmarking iteration developed by the topic group was MMVB 1.0. The main goal of it was to implement a first working benchmarking pipeline for symptom assessment systems. The

technical requirements have been discussed by the topic group during the first topic group workshop held from 11.7.2019 to 12.7.2019 in London. The MMVB 1.0 benchmarking software was then accordingly implemented in the weeks following the workshop.

Since a central part of a standardized benchmarking is agreeing on inputs and outputs of the AI systems, the work was started by defining a simple medical domain model containing hand selected conditions, symptoms, factors and profile information. Based on this domain model then the structure of inputs, outputs and the encoding of the expected outputs was defined. We refer to this model as the "London-model".

As Figure 2 shows, the model consists of 11 conditions from the field of abdominal pain together with 10 symptoms and one factor. The model states also the expected triage level which can be primary care (PC), self-care (SC) or emergency care (EC). The topic group also decided to use symptoms with all attributes “baked” into them (sometimes call pre-coordinated symptoms) and to leave the more complex explicit modelling of attributes to the next MMVB iterations.

	IBD (first presentation non flare)	GERD	simple UTI	viral GE	bladder cancer (first presentation)	acute cholecystitis	appendicitis	ectopic pregnancy	IBS	acute pyelonephritis	Abdo Pain NOS (Idiopathic)
How Common Condition?	x	xx	xx	xx	x	x	x	x, only females	xxx	x	xx
Abdo Pain Cramping Central 2 days	xx			xx		x	x	x	xx	x	x
sharp lower quadrant pain	x		x				xx	xx		xx	
Diarrhoea	xx			xx			x		xx		
Vomiting	x	x	x	xx		xx	x	xx	x	xx	
Dysuria			xx							xx	
Increased Urination Freq.			xx		x		x			xx	
Haematuria			x		xx		x	x		xx	
Weight Loss	x				xx						
Fever	x		x	x		xx	xx	x		xx	
heartburn		xx									
Factor: missed period								x			
Expected triage level	PC	PC	PC	SC/PC	PC	EC	EC	EC	PC	EC	PC

Figure 2 – "London Model" used for sampling cases for MMVB 1.0

6.1.1.2 Benchmarking methods

The general benchmarking approach for the MMVB 1.0. was to use the London Model as a ground truth, sample synthetic cases from it which could then be send to the different AIs to compute triage and differentials to then show a table with the results. To allow fast development without the need to take care of business constraints like IP considerations, at this stage we use “toy-AI” systems designed to operate on the publicly known domain model rather than the actual production AI systems of the companies.

The benchmarking was also implemented to take another fundamental business constraint into account that is not common in academia: companies cannot submit their AI as software, hence all benchmarking needs to run online, which has strong implications like for instance that test cases can never be used twice and that benchmarking needs to run in parallel to minimize and cheating possibilities e.g.,by submitting multiple AIs.

6.1.1.2.1 Benchmarking system architecture

The MMVB 1.0 version the benchmarking software was implemented as a python backend application proving all the benchmarking functionality via REST APIs to an HTML5+JS frontend for performing the benchmarking and displaying the results. The components are:

Case Generator

The case generator reads the London Model and provides a service for sampling test cases from it that can be used for the benchmarking. The generated case-sets are stored in the Case Storage.

Evaluator

The evaluator is the core of the benchmarking pipeline feeding all benchmarking cases of a case-set in the Case Storage to all the toy-AIs. This includes both several trivial toy-AIs directly implemented in the benchmarking backend, as well the actual toy-AIs hosted by the benchmarking participants in their own datacenters. The remote toy-AIs all expose a REST API endpoint that is called by the evaluator. The results of each AI are persisted in the Results Storage.

Metrics Calculator

As the report displayed by the web interface is dynamically filtered and aggregated, the Metrics Calculator is called directly by the frontend application to compute the scores for all benchmarking metrics.

Domain Model

The domain model i.e., the London Model, is the medical model describing the 11 diseases with their 10 symptoms the doctors of the topic group created for the purpose of benchmarking. It is manually exported from the google spreadsheet as CSV file that is then pre-processed into a JSON file which is then used by the Case Generator.

Case Storage / Result Storage

Both the generated cases as well as the results collected from the different AIs are persistent as JSON files in the filesystem. At this early stage it was decided that a proper database was not needed yet.

An architecture overview can be seen in Figure 3. While every participant had their own instance of the benchmarking system running during development of their toy-AI, there was also a central system hosted by Babylon Health setup with all the API endpoints of the participants. Every participant hosted its toy-AI in their datacenter using a technology of their choice.

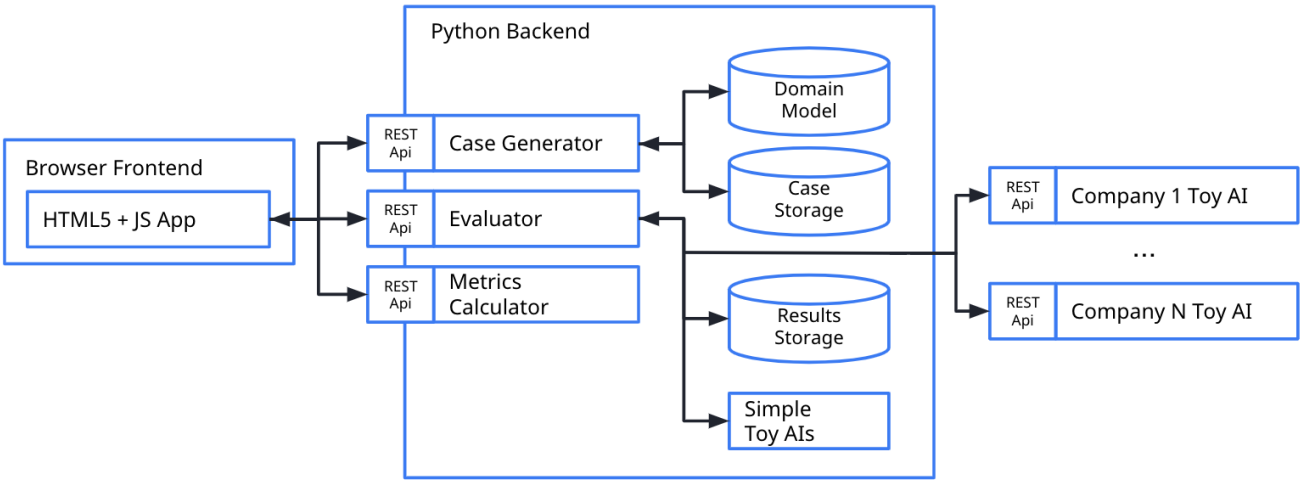


Figure 3 – MMVB 1.0 High-level architecture

6.1.1.2.2 Benchmarking system dataflow

In the MMVB 1.0 dataflow through benchmarking architecture had the following relevant stages:

Model Generation

- The medical domain model was defined by the doctors direct in a google spreadsheet.
- From there it was exported as CSV file.
- The CSV was then pre-processed and converted into a JSON file by python script.
- This JSON model is then used by the benchmarking as Domain Model.

Synthetic Case Generation

- The user triggered the creation of a new case-set in the web-interface.
- The generated case-set was then stored to the Case Storage.

Manual Case Generation

- The doctors also created a set of 12 manual cases directly in the same spreadsheet as the London Model based on a template structure.
- The cases have been exported as CSV file.
- The CSV was transformed into the same case format used by the Case Generator and stored as case-set in the Case Storage where it could be used as any other case-set by the benchmarking

Benchmarking

- The user selected a case set for a benchmarking.
- The Evaluator read a selected case set from the Case Storage and sent it to the AIs.
- The AIs responded with their result which have then been stored by the evaluator in the Result Storage.

- The web-application then used the Metrics Calculator to compute the metrics based on the results stored in the Results Storage as well as the corresponding cases stored in the Case Storage.
- The computed results have then been displayed by the web-application

6.1.1.2.3 Safe and secure system operation and hosting

In contrast to a later MVB, all MMVB iterations of the benchmarking are designed to facilitate the development of the benchmarking for AI-based symptom assessment systems. They use only toy-data and toy-AIs, hence safe and secure system operation have not been explicitly considered. The benchmarking system was hosted by Babylon Health in their infrastructure applying their standards. All the toy-AIs that participated in the MMVB 1.0 benchmarking have been hosted by the individual companies following their own standards for safe and secure operation. The only security consideration applied was that only Babylon, hosting the benchmarking system, had access to it and all the data stored including the REST API endpoints of all toy-AIs.

The data used for the benchmarking was generated with a case synthesizer running on the benchmarking system. All data sets and all results have been stored into the file system and a simple database with no further protection against data-loss or manipulation.

The benchmarking system persisted all results from all AIs, including any timeouts and errors. All results have been displayed by the benchmarking frontend application that was freely accessible in the web – including any issues with the AIs so that the AI developers could use this for debugging their toy-AIs. The benchmarking system was not part of any automated monitoring and needed to be restarted on demand.

6.1.1.2.4 Benchmarking process

This section describes what the benchmarking looks like, from the registration of participants, through the execution and resolution of conflicts, to the final publication of the results. The focus of MMVB 1.0 was to develop and test a very first symptom-assessment benchmarking pipeline. The process for this covered the step 1) case set generation, 2) running the benchmarking for a selected data set against all AI systems 3) computing & showing the results.

For performing the different steps, the benchmarking system offered a simple web-based user interface focused on the task rather than on user experience considerations. The user interface was public without password protection so that all developers and interested people from the focus group could explore it.

For creating the benchmarking case set the UI provided the screen shown in Figure 4. The only relevant parameter was here the number of cases to generate. It was also possible to select an existing case set by entering its identifier.

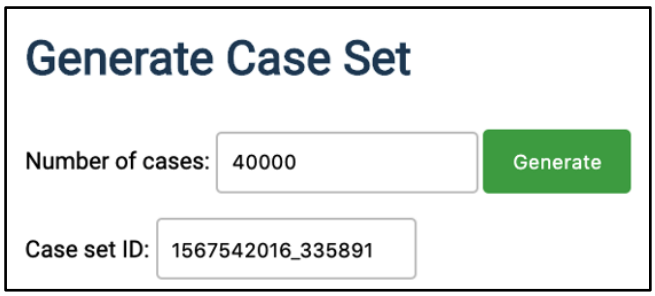


Figure 4 – MMVB 1.0 case generation UI

The user was then able to trigger the execution of the benchmarking in the screen shown Figure 5 . The first version did not support further selection of the AIs to run the benchmarking. Once the benchmarking was started a real-time log was displayed informing the user about the status.

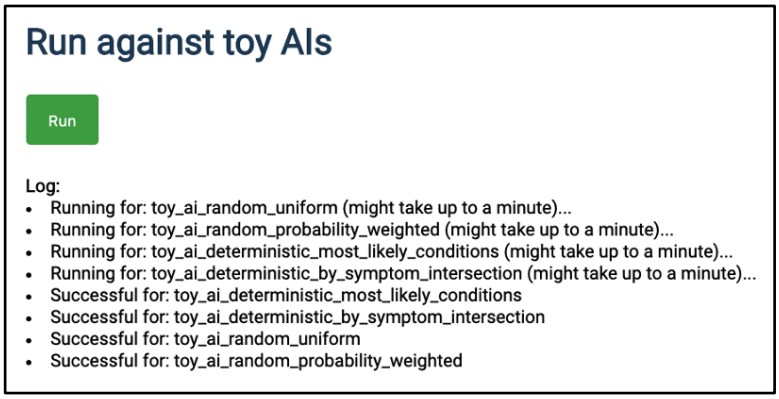
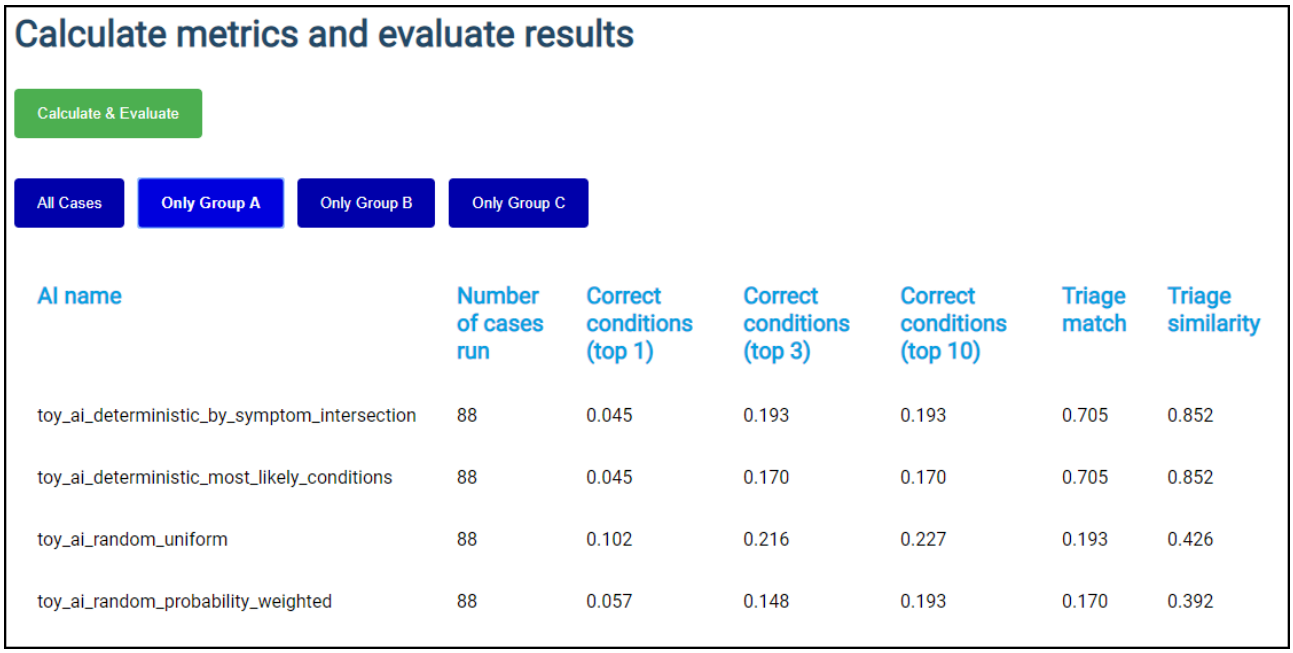


Figure 5 – MMVB 1.0 screen for running a benchmarking session

Running the benchmarking did not include the computation of the scores for the metrics. This was then triggered by clicking the "Calculate & Evaluate" button in Figure 6 . As result the report table show in this figure was generated.



AI name	Number of cases run	Correct conditions (top 1)	Correct conditions (top 3)	Correct conditions (top 10)	Triage match	Triage similarity
toy_ai_deterministic_by_symptom_intersection	88	0.045	0.193	0.193	0.705	0.852
toy_ai_deterministic_most_likely_conditions	88	0.045	0.170	0.170	0.705	0.852
toy_ai_random_uniform	88	0.102	0.216	0.227	0.193	0.426
toy_ai_random_probability_weighted	88	0.057	0.148	0.193	0.170	0.392

Figure 6 – MMVB 1.0 result screen

For this first MMVB 1.0 version there was no special benchmarking scheduled. Every developer could run a benchmarking at any time for building their own toy-AI which helped both the development of the pipeline and the toy-AIs.

For participating in the benchmarking with a toy-AI the process was to send a corresponding email with the API endpoint plus the name of the toy-AI to Yura Perov who was Babylon Health's scientist responsible for the benchmarking system instance. For building the toy-AI all participants had access to the git repository of the benchmarking system which contained the code for example toy-AIs. Inside the topic group there was also an invitation mail shared with the request and response objects for implementing the API endpoint. The participants then improved and tested their AI using the benchmarking system. If AIs were broken the developers of the benchmarking system and the AI sorted the issues out by email or the issue tracking offered by GitHub.

6.1.1.3 AI input data structure for the benchmarking

The input for the AIs the MMVB 1.0 consisted of a simplistic user profile, explicit presenting/chief complaints (PC/CC) and additional features. Beside symptoms the additional features could also contain risk factors. Table 10 shows the concrete fields with corresponding examples.

Table 10 – MMVB 1.0 input data format

Field name	Example	Description

profileInformation	`"profileInformation": {`
	`  "age": 38,`
	`  "biologicalSex": "male"`
	`}`
		● General information about the patient
		● Age is unrestricted, however for the case creation it was agreed to focus on 18-99 years.
		● As sex we started with the biological sex "male" and "female" only

presentingComplaints	`"presentingComplaints": [`
	`  {`
	`  "id": "c643bff833aaa9a47e3421a",`
	`  "name": "Vomiting",`
	`  "state": "present"`
	`  }`
	`]`
		● The complaints the user seeks and explanation/advice for
		● Always present
		● A list, but for the MMVB always with exactly one entry

otherFeatures	`"otherFeatures": [`
	`  {`
	`  "id": "e5bcdaa4cf15318b6f021da",`
	`  "name": "Increased Urination Freq.",`
	`  "state": "absent"`
	`  },`
		● Additional symptoms and factors available
		● Might include "absent", "present" and "unsure" symptoms/factors
		● Might be empty

	{
	"id": "c643bff833aaa9a47e3421a",
	"name": "Vomiting",
	"state": "unsure"
	}
	],	

As the London Model is not defining any identifiers the benchmarking system generated new ones using a hash function on the name. The different toy-AI systems used the same hash function to identify the given symptoms in the London Model again.

6.1.1.4 AI output data structure

For the MMVB 1.0 the AI systems had respond to benchmarking API calls with a JSON object encoding the conditions that might have caused the symptoms in the given input object as well as their triage result. While every case had only on correct condition the AI systems have been expected to generate a list of possible explanations that are sorted by descending likelihood. The group decided to not include an explicit score yet since the semantics of the scores of the group members is different and not comparable. The list of conditions might be empty and if so, it meant that with the given evidence no conclusive differential result was possible. For the benchmarking only the id of the condition was used. The name was added for improved readability by the developers.

In addition to the triage levels defined by the underlying London Model, for triage the AI might have responded with "UNCERTAIN" to declare that with the given evidence no conclusive triage result was possible.

Table 11 shows an example of the response data expected from an MMVB 1.0 toy-AI. Anything that could not be parsed into this structure was interpreted and logged as an error.

Table 11 – MMVB 1.0 API output encoding example

Field name	Example	Description

conditions	"conditions": [
	{
	"id": "ed9e333b5cf04cb91068bbcde643",
	"name": "GERD"
	}
	]
		• The conditions the AI considers best explaining the presenting complaints.
		• Ordered by relevance descending

triage	"triage": "EC"	• The triage level the AI considers adequate for the given evidence • Uses the same abbreviations defined by the London-model EC (emergency case), PC

		(primary care), SC (self-care), UNCERTAIN

6.1.1.5 Test data label/annotation structure

While the AI systems only received the input data described in the previous sections, the benchmarking system needed to know the expected correct answer (in ML often called 'labels') for each case of the input data so that it could compare the expected AI output with the actual one. Since this was only needed for benchmarking, it was encoded separately.

For the MMVB 1.0 benchmarking iteration the expected condition was the condition the case was synthesized or modelled for and its corresponding triage level. With the difference that the "correct" condition is only one condition rather than a list, the structure of the expected output is similar to the structure of the AI output described in the previous section. Again, it contains beside the necessary condition identifier also the human readable name. Table 12 shows an example of how the expected output is encoded in a case.

Table 12 – MMVB 1.0 AI output label encoding

Field name	Example	Description

condition	"condition": [
	{
	"id": "85473ef69bd60889a208bc1a6",
	"name": "simple UTI"
	}
	]
		• The conditions expected/accepted as top result for explaining the presenting complaints based on the given evidence.
		• A list, but only one entry for mono-morbid cases as it is the case for MMVB

expectedTriageLevel	"expectedTriageLevel": "PC"	• The expected triage level

For the MMVB 1.0 benchmarking system case data was organized in case sets. Each case set was encoded as JSON file with an array of cases. The cases contain the actual case data shared with the AI and separate section with the label/annotations to predict. Table 13 shows an example of a complete case set structure combining profile information, presenting complaints, other features and the expected values to predict.

Table 13 – An example of a MMVB 1.0 case set with a single case.

```
{
    "cases": [
        {
            "caseData": {
                "caseId": "case_mmvb_0_0_1_a_13588414",
                "metaData": {
                    "description": "a synthetic case for the MMVB"
                },
                "otherFeatures": [
                    {
                        "id": "6e16a75aff90a62324940175453741f1",
                        "name": "Diarrhoea",
                        "state": "absent"
```

```
                    },
                    {
                        "id": "7a3094ceceac3afdae15243c11031588",
                        "name": "sharp lower quadrant pain",
                        "state": "present"
                    }
                ],
                "presentingComplaints": [
                    {
                        "id": "bcdc01d83bfb31c85ec47efc0642304e",
                        "name": "Weight Loss",
                        "state": "present"
                    }
                ],
                "profileInformation": {
                    "age": 64,
                    "biologicalSex": "female"
                }
            },
            "valuesToPredict": {
                "condition": {
                    "id": "42e009a4e3d8c8a17a29b4c57311e9cf",
                    "name": "IBD (first presentation non flare)"
                },
                "expectedTriageLevel": "PC"
            }
        }
    ]
}
```


6.1.1.6 Scores and metrics

As the MMVB benchmarking iterations only used toy-AIs and toy-data the focus for the scores and metrics was to have any metrics for implementing the benchmarking at all. For this purpose, the topic group decided to use the classic top-n metrics top-1, top-3 and top-10 defining if the correct diseases was within the first n suggested conditions. The score is used internally by most topic group members and can also be found some papers on benchmarking systems for AI-based symptom assessment. For the triage the standard accuracy (# correct triage classifications / # all triage classifications) was used as well as a triage similarity score. The similarity score was defined as distance between the correct triage and the expected triage along the SC, PC, EC scale normalized by 2. For an UNCERTAIN triage level, the metric used 0.2 as soft triage level. The details for this soft triage match have not been discussed in the group as the goal was only to have some second more soft triage metric. Cases with no response or an error counted as “correct condition not contained in the result”, “no triage match” and “0 triage level similarity”. In this first iteration of the MMVB no robustness metrics have been implemented. As a first non-medical performance metric the number of successfully processed cases as computed.

6.1.1.7 Test data set acquisition

The primary data generation strategy for the MMVB 1.0 was to use the London-model to sample cases from it. Sampling was done in several steps. As first step the profile information was sampled. Here age was sampled from an equal distribution [18; 80] years. Sex was sampled between with 0.5 probability from (male, female). As next step the condition was sampled from prior probability distribution of the conditions for the sex sampled before. For this the number of “x” in the prior cells of the model ranging from “x” to “xxx” was interpreted as prior probability between 0.3 and 0.9. For each case then the symptoms are sampled according to their condition probability following the same scale from 0.3 to 0.9. However, for each symptom there is only a 0.8 probability to include the symptom in the case and of these the symptoms are marked as “unsure” with probability 0.1.

Even if synthetic data will play an important role especially for benchmarking robustness, the topic group agrees that the MVB benchmarking always must contain real cases as well as designed case vignettes. This case data needs to be of exceptionally high quality since it is used to potentially influence business relevant stakeholder decisions. At the same time, it must be systematically ruled out that any topic group member can access the case data before the benchmarking, effectively ruling out that the topic group can check the quality of the benchmarking data. This is an important point to maintain trust and credibility.

For creating the benchmarking data therefore, a process is needed that blindly creates with reliably reproducible high-quality benchmarking data that all the topic group members can trust to be fair for testing their AI systems. With the growing number of topic group members from the industry it also became clear that "submitting an AI" to a benchmarking platform for instance as a docker container containing all the companies IP is not feasible. With AIs not running in a sandbox, it must be assumed that the AIs will remember all cases for the next benchmarking so that cases cannot be reused and new case sets will be needed for new benchmarking iterations. The case creation process therefore does not only need to guarantee high quality but also high efficiency and scalability.

One way to approach to achieve consistently high quality is to define the methodology, processes and structures that allows clinicians all around the world in parallel to create new benchmarking cases. As part of this methodology annotation guidelines are a key element. The aim is that these could be given to any clinician tasked with creating synthetic cases or labelling real world cases and if the guidelines are correctly adhered to, will facilitate the creation of high quality, structured cases that are "ready to use" in the right format for benchmarking. The process would also include an n-fold peer reviewing processes. There will be two broad sections of the guideline:

1. **Test Case Corpus Annotation Guideline** - this is the wider, large document that contains the information on context, case requirements, case mix, numbers, funding, process, review. It is addressed to institutions like hospitals that would participate in the creation of benchmarking data.
2. **Case Creation Guideline** - the specific guidelines for clinicians creating individual cases.

As part of MMVB 1.0 the topic group decided to start the work on a first annotation guideline and to test them with real doctors. Due to the specific nature of the London Model the MMVB 1.0 is based on, a first, very specific annotation guideline was drafted to explore this topic and learn from the process. The aim was to:

- create clinically sound cases for MMVB 1.0 within a small "sandbox" of symptoms and conditions that were mapped by the clinicians in the group
- explore what issues/challenges will need to be considered for a broader context

A more detailed description of the approach and methodology will be outlined in the MMVB guideline itself, but broadly followed the following process:

- Symptoms and conditions mapped by TG clinicians within the GI/Urology/Gynaecology condition scope of the London Model

- Alignment on case structure and metrics being measured.

The bulk of this activity was carried out in a face-to-face meeting in London, telcos and through collaboration on shared online documents.

Table 14 – Case example for the London Model

Age 18-99	25
Gender Biological, only male or female	male
Presenting Complaint (from symptom template)	vomiting
Other positive features (from symptom template)	abdominal pain central crampy "present", sharp lower quadrant pain 1 day "absent" diarrhoea "present" fever 'absent"
Risk factors	n/a
Expected Triage/Advice Level What is the most appropriate advice level based on this symptom constellation	self-care
Expected Conditions (from condition template)	viral gastroenteritis
Other Relevant Differentials (from condition template) What other conditions is it relevant to have on a list based on the history.	irritable bowel syndrome
Impossible Conditions (from condition template) (are there any conditions, based on the above info, including demographics, where it is not possible* for a condition to be displayed) – e.g.,endometriosis in a male	ectopic pregnancy
Correct conditions (from condition template)	appendicitis

The instructions (with an example) were shared with the clinicians inside the topic groups for creating benchmarking cases. Feedback was collected on the quality of the guidelines and the process.

Both the synthetic data and the cases created by the doctors served as expected their purpose for allowing to build and test a first version of a benchmarking so that we will continue this approach for the next MMVB iterations.

6.1.1.8 Data sharing policies

For the MMVB 1.0 iteration only synthetic cases and 12 cases created by the doctors in the topic group have been used. The cases are highly specific of this minimalistic benchmarking iteration and are not based on real cases. Hence, the data was freely accessible as online spreadsheet[4].

6.1.1.9 Baseline acquisition

The main purpose of benchmarking is to provide stakeholders with the numbers they need to decide whether AI models provide a viable solution for a given health problem in a designated context (intended use). To achieve this, the performance of the AI models needs to be compared with available options achieving the same clinically meaningful endpoint. This, in turn, requires data on the performance of the alternatives, ideally using the same benchmarking data. As the current alternatives typically involve doctors, it might make sense to combine the test data acquisition and labelling with additional tasks that allow the performance of the different types of health workers to be assessed. For the MMVB 1.0 assessing any baseline was out of scope.

6.1.1.10 Reporting methodology

As the MMVB 1.0 uses toy-AIs and toy-data, the benchmarking results have only been relevant for the development of the benchmarking itself and for sharing with the focus group. The results have been documented in this TDD document and presented at the focus group meeting in Zanzibar, 2-5 September 2019. In this early development phase, all AI developers had always full transparent access to all results of all AI systems by using the screen shown in Figure 6.

The future reporting methodology was discussed during the topic group workshops planning the MMVB 1.0 benchmarking iteration. From the discussion it was clear that in contrast to other topic groups, a single leader-board is not sufficient for the benchmarking systems for AI-based symptom-assessment. There is the need for numerous dimensions to group and filter the results by in order to answer questions reflecting the full range of possible use cases (narrow and wide) e.g.,the questions which systems are viable choices in Swahili speaking, offline scenarios with a strong focus on pregnant women vs. a general use symptom-assessment tool.

As first step in this direction for MMVB 1.0, a simple interactive table was implemented to show that it is possible to filter results. For the illustrative purposes of the MMVB 1.0, three simple groups are introduced that filter the results by the age of case patients.

From the workshop it became also clear that for this topic group it is unlikely that all results for all AI can always be publicly shared. The thinking was going the direction of opting-in/out for result publication in combination with means for allowing participants to share access to their own results with their own stakeholders.

6.1.1.11 Result

As this benchmarking iterations was only an intermediate development step and no final result was recorded.

---

4 https://docs.google.com/spreadsheets/d/111D40yoJqvvHZEYI8RNSnemGf0abC9hQjQ7crFzNrdk/edit#gid=1175944267

6.1.1.12 Discussion of the benchmarking

As intended, the MMVB 1.0 reached a point where first results from a new build benchmarking pipeline for AI-based symptom-assessment could be seen. The first minimalistic user interfaces allowed to create case sets and run benchmarking against toy-AIs hosted in the cloud by the different participants. Building this iteration also successfully established the cooperation on the technical implementation inside the topic group.

While the MMVB 1.0 provides a good starting point, we collected the following learnings for the next MMVB iterations until the work on the MVB can start:

Adding symptom attributes

Using symptoms with pre-coordinated attributes like in “sharp, lower right quadrat abdominal pain” are too simplistic and need to be replaced in the next iteration with explicit attribute modelling

Adding more factors

The London Model contains the factor “females only” as comment and the factor “missed period” only as binary flag. For the next iteration modelling factors as probability distribution is needed.

Adding “dimensions” and using them in a more interactive reporting

For exploring the necessary drill-down reporting features needed to provide stakeholder with the answers from the benchmarking for their decision-making, we need to introduce the annotation for both data and AI with additional flexible metadata like their offline capabilities, supported languages, regulatory constraints etc.

Implementation of some robustness scores

In the next iterations we need to see how to integrate the medical performance metrics with non-medical ones and the ones for robustness as they technically work in a different way than only comparing AI results with expected results.

Better support for “unsure” / “unknown” AI answers

In the current iteration answering with a dangerously wrong or misleading answers is counted in the same way as stating that no reliable answer could be computed. As this is a feature some of the real AIs have, we need to reflect this in the MVB metrics.

Scores dealing with AI errors

While for the internal benchmarking of participants errors play no important role as final numbers are only taken after all bugs have been removed, a central benchmarking by the focus group or its successors we must expect some AIs to fail with an error on certain cases which needs to be reflected in some of the metrics.

Dynamic AI self-registration through the web-interface

The development of the MMVB 1.0 has shown that it would be more practicable if participants could register their different toy-AIs themselves without changing the codebase of the benchmarking system.

Running the benchmarking by case rather than by AI

In the current implementation performs the benchmarking AI by AI collecting all results from one AI before collecting it form the next AI. As part of increasing the resilience of the benchmarking against manipulation the next iteration should all AIs for the result of a case in parallel in combination with a short timeout so that side-channel communication between AIs would not provide any advantage.

Agreeing on how to encode test data is the core task of this topic group

The work on the first workshop also underlined that the most important and most complex unsolved task of the topic group on AI-based symptom assessment is agreeing on a joint input ontology for encoding factors, symptoms and their attributes in a way that can be interpreted by all AIs.

The need for a sub-topic taking care of NLP dialogs

The workshop for MMVB 1.0 has raised the point that we at some point will need a sub-group taking care of benchmarking the conversational NLP part of the self-assessment dialog some of the participants support with their systems. A point in 2023, when the focus group ended, became with the wide availability for LLM based chat systems even more relevant.

A case set statistics analysis is needed

The work on the pipeline has shown that future version would need tools for checking the statistics of the benchmarking data to make sure that there are for instance no issues with the case synthesizer.

The need to test the dialog question-flow

Most symptom assessment systems engage with the user in a dialog asking the user for their symptoms. While it was decided to start simple by providing the AIs with complete case vignettes, it is important to test the question-flow too since not asking the right questions can significantly reduce the accuracy of symptom assessment systems.

6.1.1.13 Retirement

As the MMVB 1.0 was an intermediate development step the benchmarking system and the corresponding toy-AI endpoints have already been retired. While the code of the benchmarking system is still in GitHub it was up to the participants how they handle the retirements of their toy-AIs and their source code. The synthetic test data was not archived. Both the London Model used for generating the synthetic test data as well as the 12 cases manually created by the doctors of the topic group are still available at:

https://docs.google.com/spreadsheets/d/111D40yoJqvvHZEYI8RNSnemGf0abC9hQjQ7crFzNrdk/edit#gid=575520860

6.1.2 Benchmarking version MMVB 2.0 - 2.2

This section described the benchmarking process for the benchmarking versions 2.0-2.2. As the versions 2.x are extensions of version 1.0 we focus on the differences.

6.1.2.1 Overview

After finishing the MMVB 1.0 version which was centred around the “London model” described in the previous section 6.1.1, the work focused on addressing the next steps pointed out in 6.1.1.12. How to approach the different challenges mentioned there was discussed by the topic group during the second workshop held from 10.10.2019 - 11.10.2019 in Berlin.

6.1.2.1.1 Adding symptom attributes

The most relevant limitation of the MMVB 1.0 model was the missing support of explicitly modelled *attributes* for describing details like intensity, time since onset or laterality of symptoms like headache. So far, the model contained only pre-coordinated symptoms combining a symptom with a specific attribute expression pattern like "abdominal pain cramping central 2 days" or "sharp lower quadrant pain". While this was not an issue for the small test domain model, it was not scalable for the real benchmarking with potentially millions of precoordinated symptom – attribute constellations. For MMVB 2.2 the attributes therefore have been explicitly added as shown in Figure 7.

	A	B	C	D	E	F	G	I	J	K	L
1	Type	Feature name	Feature ID	Attribute name	Multi-Select ?	Attribute ID	State name	State ID	IBD (first presentation non flare)	GERD	simple UTI
2		*Condition ID*							TMP_ID_D_1	ICD10K21	TMP_ID_D_2
3		*Prior probability*							x	xx	xx
4	symptom	abdominal pain	SNOMED21522001	PRESENCE		PRESENCE			xxx	xx	xx
5				pain intensity	N	SNOMED406127006	mild	SNOMED255604002		xx	xx
6							medium	TMP_ID_1	xx	xx	x
7							severe	TMP_ID_2	x		
8				time since onset	N?	TMP_ID_A_1	less than a day	TMP_ID_3			x
9							a couple days (1-2 days)	TMP_ID_4	x		xxx
10							3 days - to 1 week	TMP_ID_5	xx	x	xx
11							a few weeks (1 weeks - 1 month)	TMP_ID_6	xx	xx	
12							1 month to 1 year	TMP_ID_7	x	xx	
13							a year or more	TMP_ID_8		x	
14				quality of abdo pain	N?	TMP_ID_A_2	cramping	TMP_ID_9	xx		
15							dull	TMP_ID_10	x	xx	xx
16							sharp	TMP_ID_11	x	x	x
17				location	Y	TMP_ID_A_3	generalised	TMP_ID_12	x		
18							right upper	TMP_ID_13	x	x	
19							left upper	TMP_ID_14	x	x	
20							epigastric	TMP_ID_15		xxx	
21							right lower	TMP_ID_16	xx		xx
22							left lower	TMP_ID_17	xx		xxx
23							suprapubic	TMP_ID_18	x		xxx
24							right loin	TMP_ID_19			
25							left loin	TMP_ID_20			
26							central	TMP_ID_21	x	x	

Figure 7 – Abdominal Pain symptom with attributes inside the Berlin Model

Compared to MMVB 1.1 the pre-coordinated symptoms have been replaced with a single symptom like "abdominal pain". For expressing the details, the symptoms now contained sub structures for each attribute stating the probability distribution of the attribute for all the conditions causing this symptom. The corresponding extension of the London model agreed upon in Berlin was named the “Berlin model”.

6.1.2.1.2 Factor distributions

The second improved aspect was the more detailed modelling of risk factors. In the initial model it was only informally noted in a comment field that for instance "ectopic pregnancy" allows "only females". For later supporting more factors that also influence the AIs in a non-binary way, we

introduced explicit probability distributions modulating the prior distributions of the different conditions. Figure 8 and Figure 9 show the refined probability distributions for ectopic pregnancy.

A	B	C	D	E	F	G	I	J	K	L
Type	Feature name	Feature ID	Attribute name	Multi-Select ?	Attribute ID	State name	State ID	IBD (first presentation non flare)	GERD	simple UTI
						1 month to 1 year	TMP_ID_7		xx	
						a year or more	TMP_ID_8		x	
factor	period lateness	BLA1231232	presence		BLUB98762378	not present	YXC_NP	1	1	1
						present	ASD_P	1	1	1
factor	sex	SNOMED734000001	sex		TMP_ID_A_5	female	TMP_ID_25	1	1	1
						male	TMP_ID_26	1	1	1
	Expected triage level							PC	PC	PC

Figure 8 – Factors with state details inside the Berlin Model

A	B	K	L	M	N	O	P	Q	R	S
Type	Feature name	GERD	simple UTI	viral GE	bladder cancer (first presentation)	acute cholecystitis	appendicitis	ectopic pregnancy	IBS	acute pyelonephritis
		xx								
		x								
factor	period lateness	1	1	1	1	1	1	0.8	1	1
		1	1	1	1	1	1	1.2	1	1
factor	sex	1	1	1	1	2	1	1	1	1
		1	1	1	1	1	1	0	1	1
	Expected triage level	PC	PC	SC/PC	PC	EC	EC	EC	PC	EC

Figure 9 – Refined factor distributions for ectopic pregnancy inside the Berlin Model

For example, "male" chosen for factor "sex" now implies that the probability of "ectopic pregnancy" is zero.

6.1.2.1.3 New benchmarking backend application

With a view towards adding future functionality required for the MVB, the topic group agreed to reimplement the original backend implemented using Flask using the Django framework providing all benchmarking logic as REST API endpoints to the new frontend application. As part of implementing the new backend it was also decided to switch to a dedicated database to store cases and other data relevant to the benchmarking application.

6.1.2.1.4 New benchmarking frontend application

For MMVB 2.2 we also implemented a new web-based frontend application. Previously the frontend was a single file containing all necessary code to communicate with the backend and run basic benchmarks. Crucial data was stored in-memory and it was not supported to return to a running benchmark or viewing the results of a benchmark that was run in another browser. To address those issues, we implemented a new frontend meeting the following criteria: state-less, proper API communication, interactive, user-friendly, extensible (to in the future include features like interactive result drilldowns). The new user interface was also designed to support the new Berlin model attributes and factors. We also improved the usability and visual appearance of the frontend.

6.1.2.1.5 Case Annotation Tool

The benchmarking of the MMVB 2.2 version used mainly synthetic data sampled from the Berlin model defined by the doctors in the topic group. However, for learning how to create representative real-world cases for the later MVB version of the benchmarking, the doctors in the topic group also created cases. In MMVB 1.0, we used spreadsheets for describing the cases, however with the introduction of the Berlin model this reached its limit and it became necessary to develop a dedicated annotation tool. As part of MMVB 2.2 we therefore implemented a first simple

annotation tool using the same frontend technology stack as the new benchmarking frontend. The annotation tool was then used by the doctors in the topic group to create benchmarking cases on top of the synthetic ones to collect learnings for future annotation tool iterations.

6.1.2.2 Benchmarking methods

This section provides details about the methods of the MMVB 2.2 benchmarking. It contains detailed information about the benchmarking system architecture, the dataflow and the software for the benchmarking process (e.g., test scenarios, data sources and legalities).

6.1.2.2.1 Benchmarking system architecture

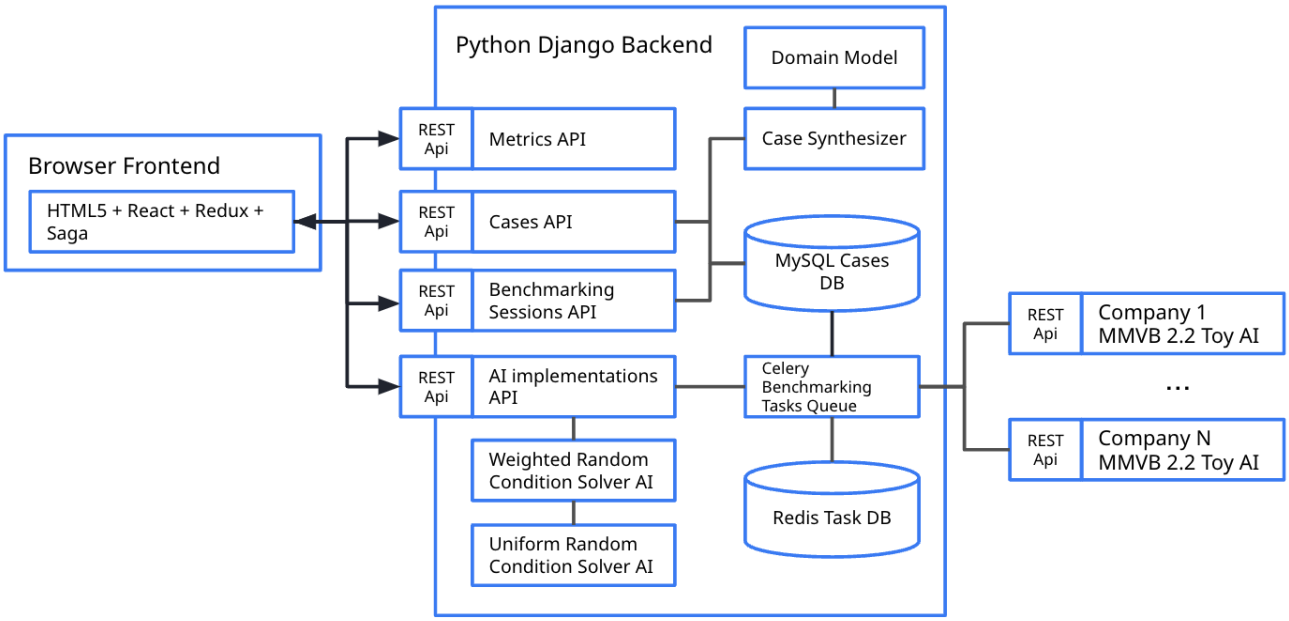


Figure 10 – MMVB 2.2 High-level architecture

6.1.2.2.1.1 Backend architecture changes

The new backend for the MMVB 2.2 benchmarking was based on Django. MySQL was chosen as the database for the new backend. To support executing the benchmarking on multiple AIs it was decided to use Celery and Redis to manage the task queue. The latest MMVB 2.2 backend was designed to support the new Berlin model data structures.

The most important part was implementation of a Berlin model case synthesizer. As with the previous London model, synthetic cases are generated based on the simple medical domain model for abdominal conditions, findings, factors and profile information. First a condition was sampled at random according to its prior probability, considering the weight of factors associated with the synthetic patient sampled from the patient factor distribution. Then clinical findings have then been sampled for that condition according to their strength of association with the condition. For the Berlin model, attributes have been sampled for each finding as needed.

Based on the new data structures we also changed the API interface to the toy-AIs to include the new attributes and factors in the case data. Accordingly, the topic group members update their toy-AIs to support the new model. For the build-in toy AIs “Weighted Random Conditions Solver” and “Uniform Random Condition Solver” this was implemented by the team working on the backend.

The previous backend separated each aspect of the benchmarking process (case generation, toy AIs, case evaluation and metrics calculation) into independent microservices. For the new backend we

implemented the different aspects as separate Django applications within the same project. The MMVB 2.2 design contained the following Django applications:

Common

This application acted as an aggregator of all the common/shared functionalities for the other applications.

Case Synthesizer

This application was responsible for implementing the structure for synthesizing cases and case sets from the Berlin model.

Toy AIs

This application was responsible for the structure needed for implementing toy AIs and registering them as available AI implementations.

AI Implementations (API)

This application was responsible for implementing the data model and API for AI Implementations. It allowed AI developers to register new AI implementation, list, delete and update the registered AI implementations or ask for their health status.

GET	/api/v1/ai-implementations
POST	/api/v1/ai-implementations
GET	/api/v1/ai-implementations/{id}
PUT	/api/v1/ai-implementations/{id}
PATCH	/api/v1/ai-implementations/{id}
DELETE	/api/v1/ai-implementations/{id}
GET	/api/v1/ai-implementations/{id}/health-check

Figure 11 – MMVB 2.2 ai-implementations API

Cases (API)

This application was responsible for implementing the data model and API for cases and case sets. It offered to list, retrieve, create and update cases and case sets.

GET	/api/v1/cases
POST	/api/v1/cases
GET	/api/v1/cases/{id}
PUT	/api/v1/cases/{id}
PATCH	/api/v1/cases/{id}
DELETE	/api/v1/cases/{id}
GET	/api/v1/case-sets
POST	/api/v1/case-sets
GET	/api/v1/case-sets/{id}
PUT	/api/v1/case-sets/{id}
PATCH	/api/v1/case-sets/{id}
DELETE	/api/v1/case-sets/{id}
POST	/api/v1/cases/synthesize
POST	/api/v1/case-sets/synthesize

Figure 12 – MMVB 2.2 cases and case sets API

Benchmarking Sessions (API)

This application was responsible for implementing the structure for creating, retrieving and handling benchmarking sessions.

GET	/api/v1/benchmarking-sessions
POST	/api/v1/benchmarking-sessions
GET	/api/v1/benchmarking-sessions/{id}
PUT	/api/v1/benchmarking-sessions/{id}
PATCH	/api/v1/benchmarking-sessions/{id}
DELETE	/api/v1/benchmarking-sessions/{id}
GET	/api/v1/benchmarking-sessions/{id}/results
GET	/api/v1/benchmarking-sessions/{id}/status
POST	/api/v1/benchmarking-sessions/{id}/run

Figure 13 – MMVB 2.2 benchmarking-sessions API

Metrics

This application was responsible for the structure needed for implementing metrics as well as calculating the metrics for a given benchmarking session result. In contrast to the benchmarking systems with a single leader-board, for symptom assessment we need an interactive drill-down to a given context which requires the dynamic recomputation of sub-set metrics via this API endpoint.

GET	/api/v1/metrics

Figure 14 – MMVB 2.2 metrics API

6.1.2.2.1.2 Frontend architecture changes

The new frontend was implemented using TypeScript. React was chosen for being a commonly used frontend technology. In the background Redux was used in conjunction with Saga to handle state-management and asynchronous communication with the backend. For basic design and user-friendly building blocks a React-specific community implementation of Google's Material Design guidelines called Material UI (material-ui.com) was chosen. Most of design-needs have been covered by the provided components. For charts the Baidu-backed ECharts library was chosen.

6.1.2.2.2 Benchmarking system dataflow

In the MMVB 2.2 version the overall date flow was similar to the one for MMVB 1.0 and has still the following components:

Model Generation

The more complex Berlin model was defined by the doctors directly in a google spreadsheet.

Derived from this was then a technically cleaner spreadsheet version.

From there it was exported into several CSV files for the findings, conditions, attribute value sets and the condition-finding relations.

These CSV files have then been read by the backend directly with no JSON intermediate format as in MMVB 1.0.

Synthetic Case Generation

The user triggered the creation of a new case set in the web-interface.

The cases have then been sampled from the Berlin model.

The case sets have then been stored in the MySQL database.

Manual Case Generation

The doctors created manually curated benchmarking cases using the newly developed case annotation tool.

The corresponding case sets have been stored in the same database as the synthetic case sets.

Benchmarking

The evaluator read a selected case set from the case storage and sent it to the AIs. In contrast to earlier versions the cases have been sent case by case.

The AIs responded with their results which have been stored by the evaluator in the database.

The web-app then used the metrics API to compute the metrics based on the results stored in the results storage and the corresponding cases stored in the case storage.

The computed results have then been displayed by the web-application. All case data and results from the benchmarking runs have been stored in the database of the backend. As this was still not the MVB yet there was no need for any backup strategy.

6.1.2.2.3 Safe and secure system operation and hosting

In contrast to the later MVB, all MMVB iterations of the benchmarking are designed to facilitate the development of the benchmarking of AI-based symptom assessment systems. They use only toy-data and toy-AIs, hence safe and secure system operation have not been explicitly considered. The benchmarking system for MMVB 2.2 was hosted by Ada Health in a GCP VM instance. All toy-AIs that participated in the benchmarking have been hosted by the individual companies following their own standards for safe and secure operation.

While for the first benchmarking iterations only Babylon, hosting the first benchmarking system, had access to it and all the data stored including the REST API endpoints of all toy-AIs, for the MMVB 2.2 we allowed self-registration of API endpoints with no special protection. The data used for the benchmarking was generated with a case synthesizer running on the benchmarking system. All data sets and all results have been stored in a MySQL database. The benchmarking system persisted all results from all AIs, including any timeouts and errors. All results have been displayed by the benchmarking frontend application that was freely accessible in the web – including any issues with the AIs so that the AI developers could use this for debugging their AIs. The

benchmarking system was not part of any automated monitoring and needed to be restarted on demand. As the new implementation was much more stable, this was rarely needed.

6.1.2.2.4 Benchmarking process

This section describes how the benchmarking looks from the registration of participants, through the execution and resolution of conflicts, to the final publication of the results. The MMVB 2.2 is still benchmarking toy-AIs with synthetic data. The process is therefore similarly simple as the one for MMVB 1.0. The most important steps include:

Landing Page

Starting point for benchmarking was the landing page shown in Figure 15. It featured cards as shortcuts to the AI-Implementations, data sets and benchmarking sessions, all with indications of the number of entities currently available there. We also added an additional navigation bar on the left side to have all relevant options permanently available.

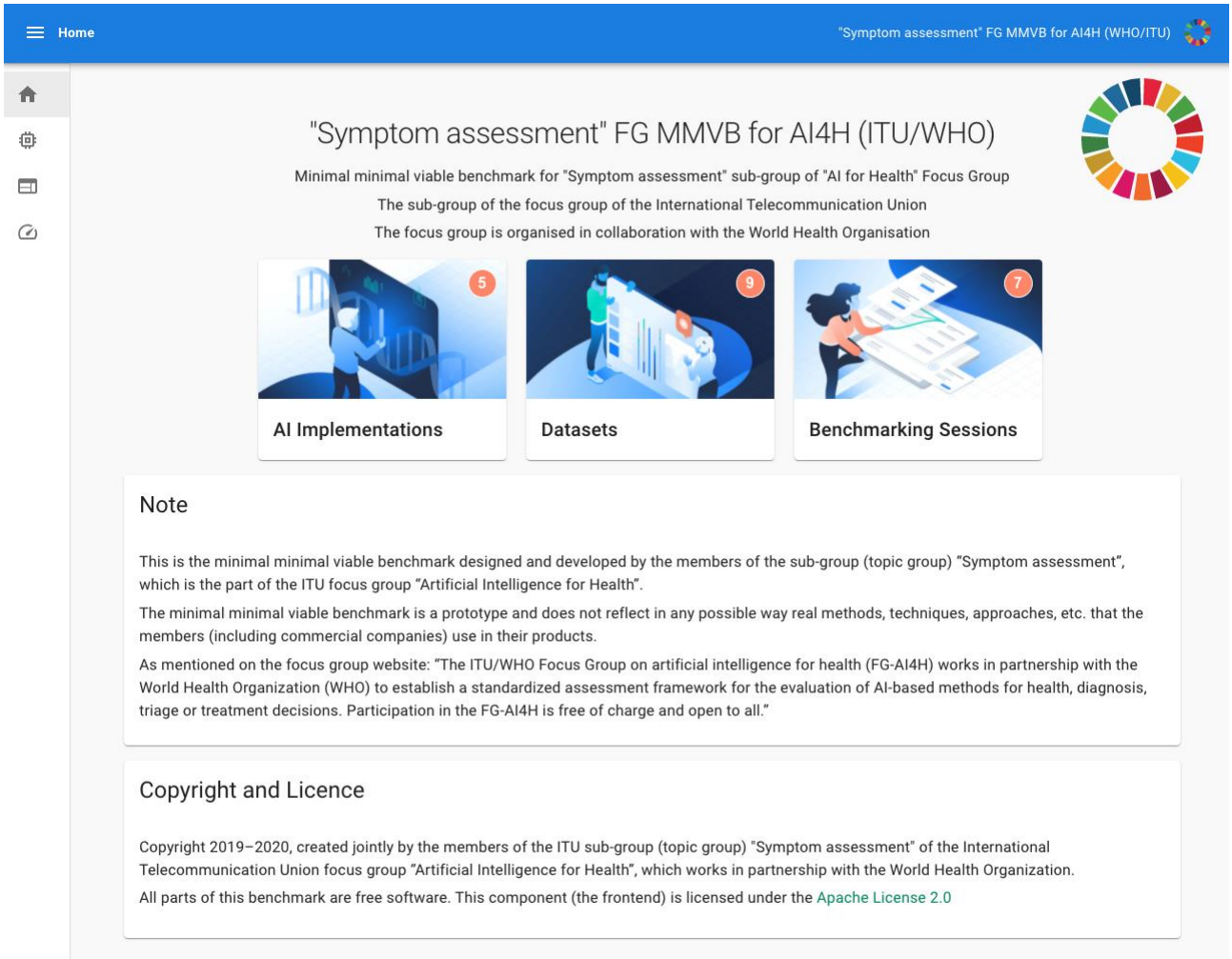


Figure 15 – 2.2 Version of the Benchmarking start page

AI Registration

As a second step AI developers could register their AIs with name and API endpoint. For had proven to be more practicable and was also closer to the MVB where AI developers would register and submit AIs for official benchmarking too. This version had no protection mechanisms

implemented so that for instance all AI developers could change the endpoints of all the other AIs. Adding authentication, rights and roles is therefore one of the important steps towards the MVB.

Figure 16 shows the list of AI implementations with the button for registering a new AI.

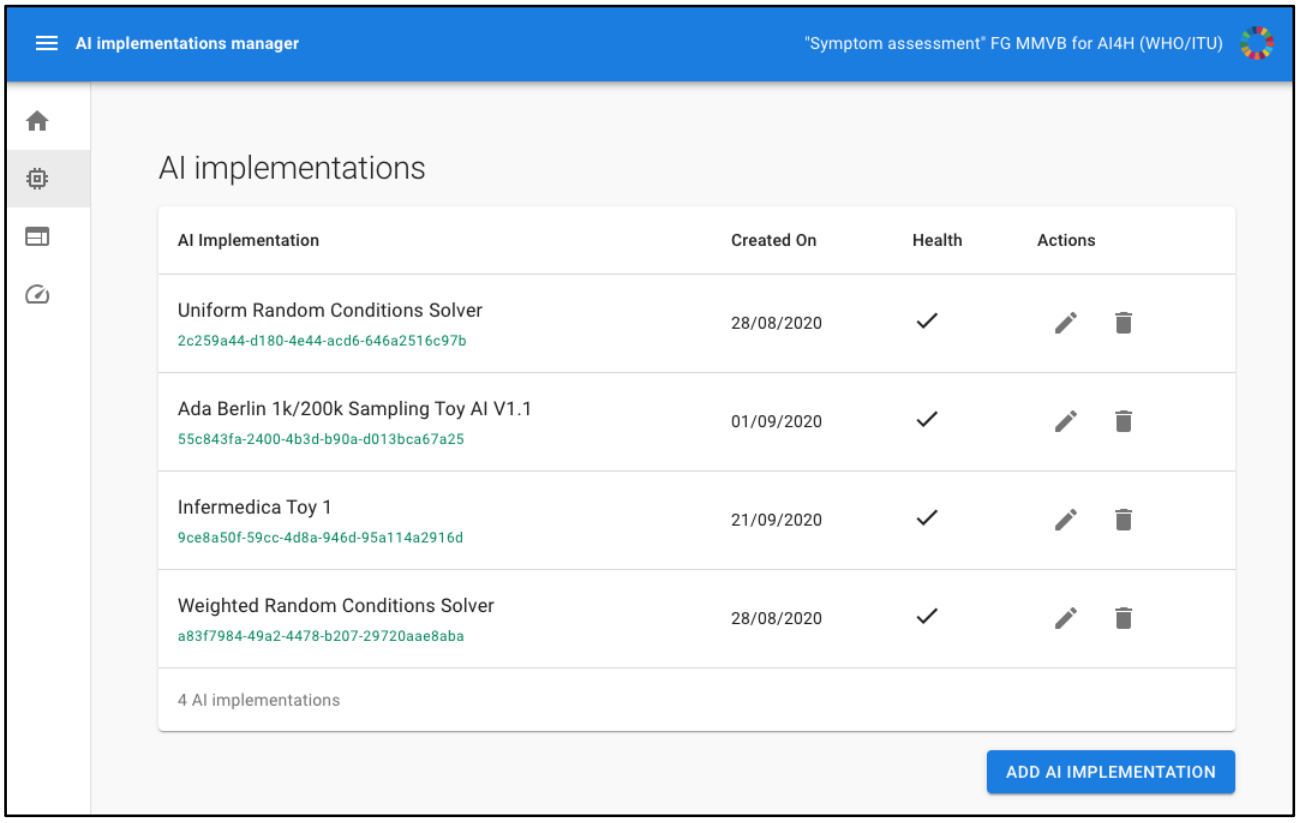


Figure 16 – The AI implementations list now featuring the ability of adding new AIs and editing existing ones.

Creating a benchmarking data set

The next step was creating a case set for the benchmarking. Figure 17 shows the page with all the case sets created so far. Users could create new case sets by clicking the “Add AI Implementation” button in the screen shown in Figure 16. Beside the name the user could define the number of synthetic cases to sample from the Berlin model. Manual cases had to be directly registered with the backend and could not be added through the UI.

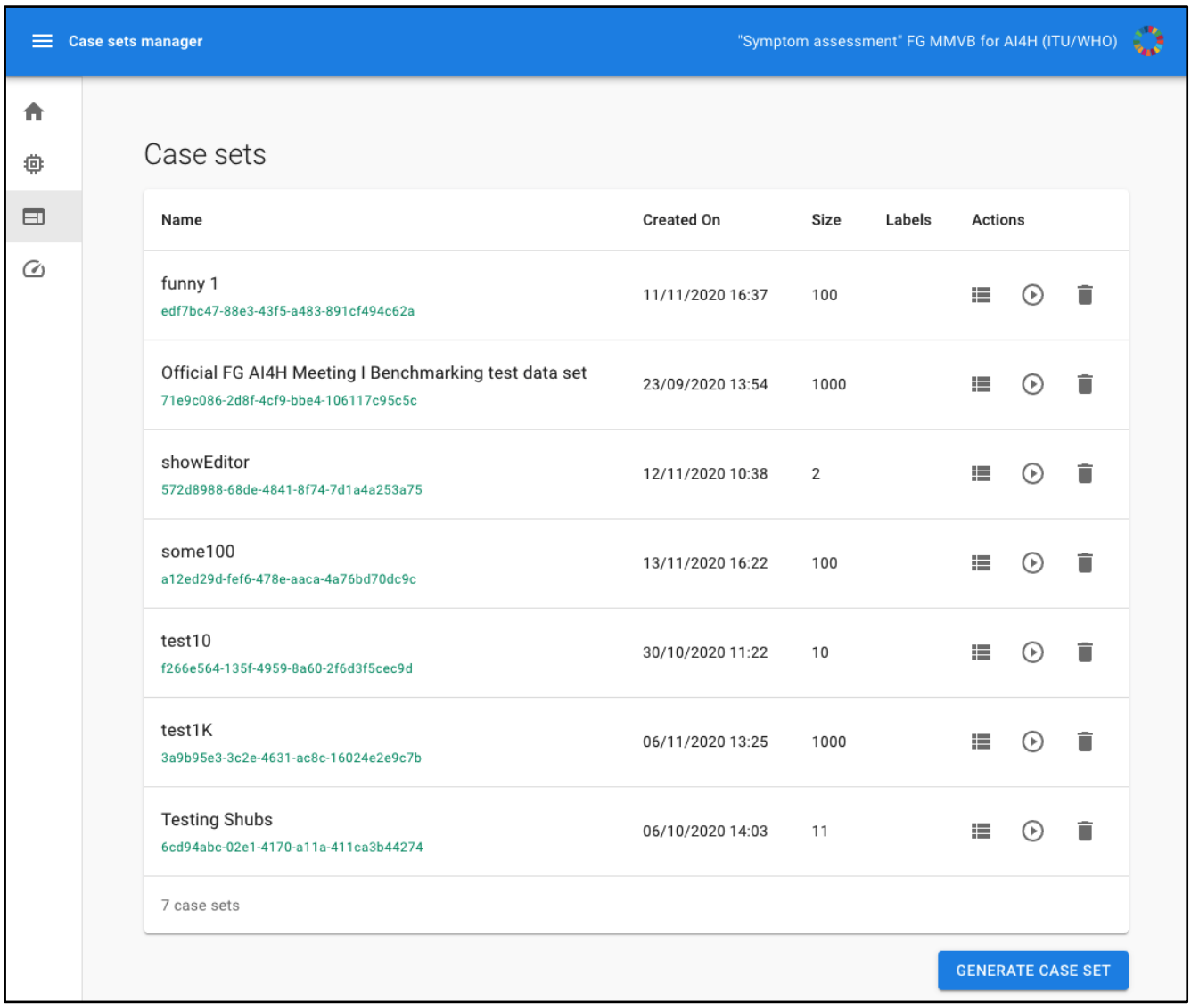


Figure 17 – MMVB 2.2 case sets overview page

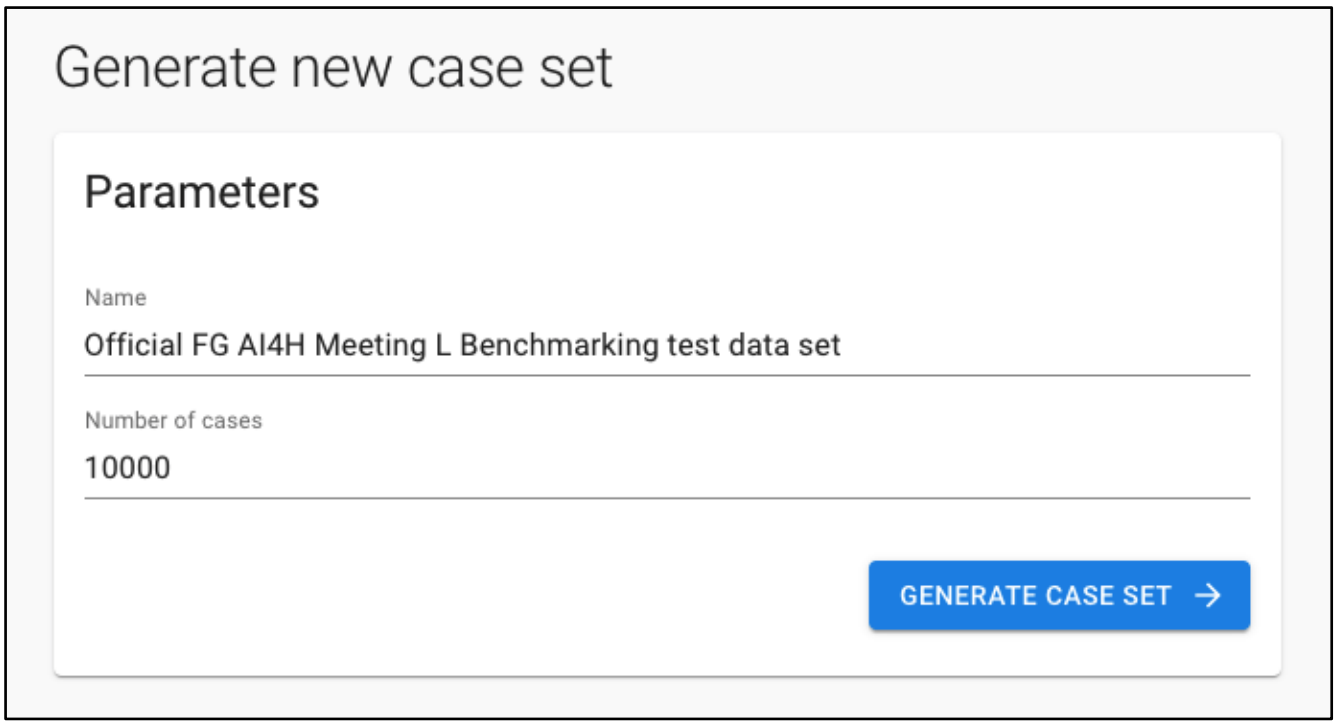


Figure 18 – MMVB 2.2 case set creation page

Creating a benchmarking session

Once AIs have been registered and a case set created, the user could create a new benchmarking session. Figure 19 shows the overview page with the list of the existing benchmarking sessions. By clicking on the corresponding button, the user could create a new session using the screen shown in Figure 20 by selecting the case set and the AIs that should participate in the benchmarking.

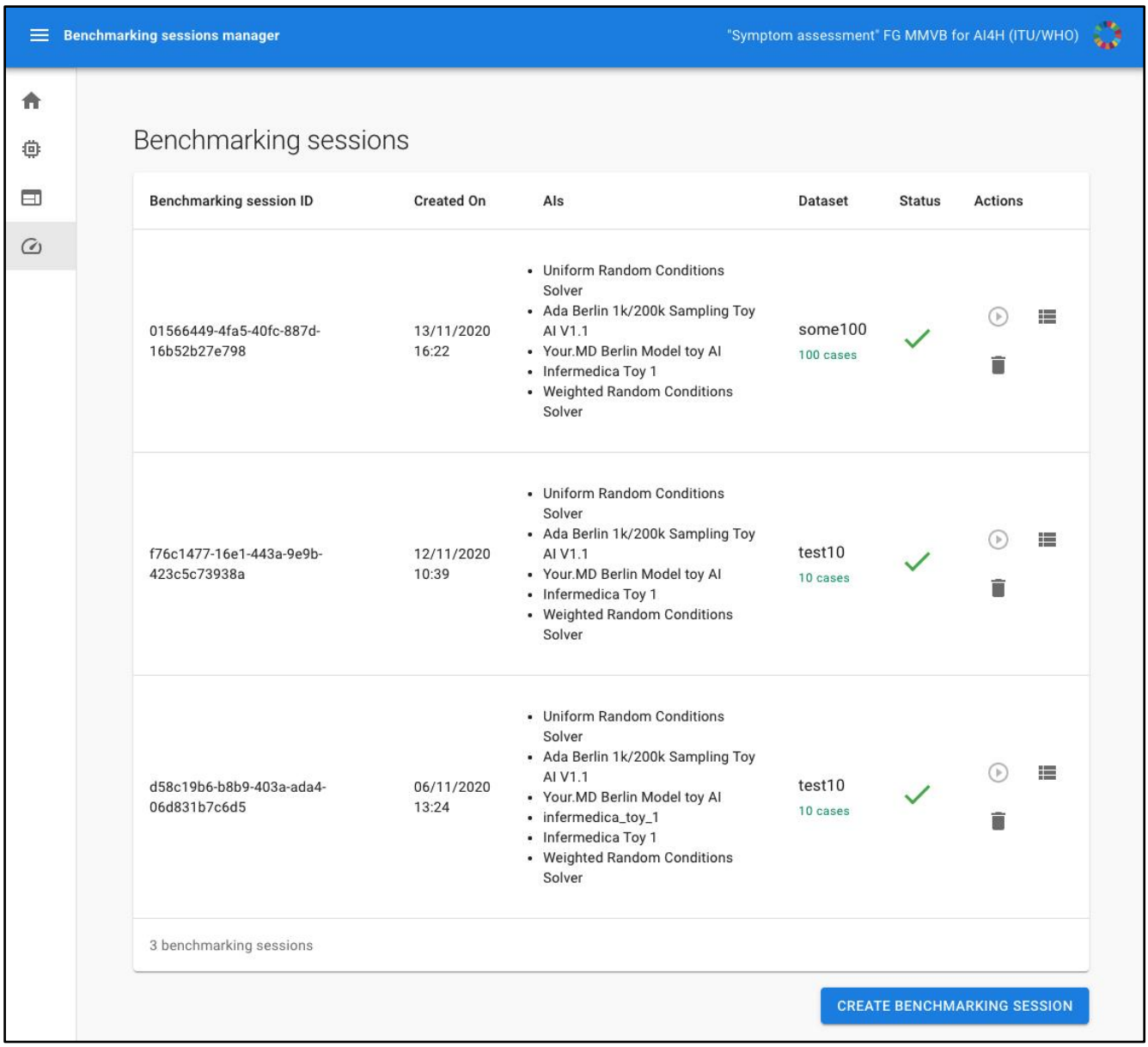


Figure 19 – MMVB 2.2 benchmarking sessions overview page

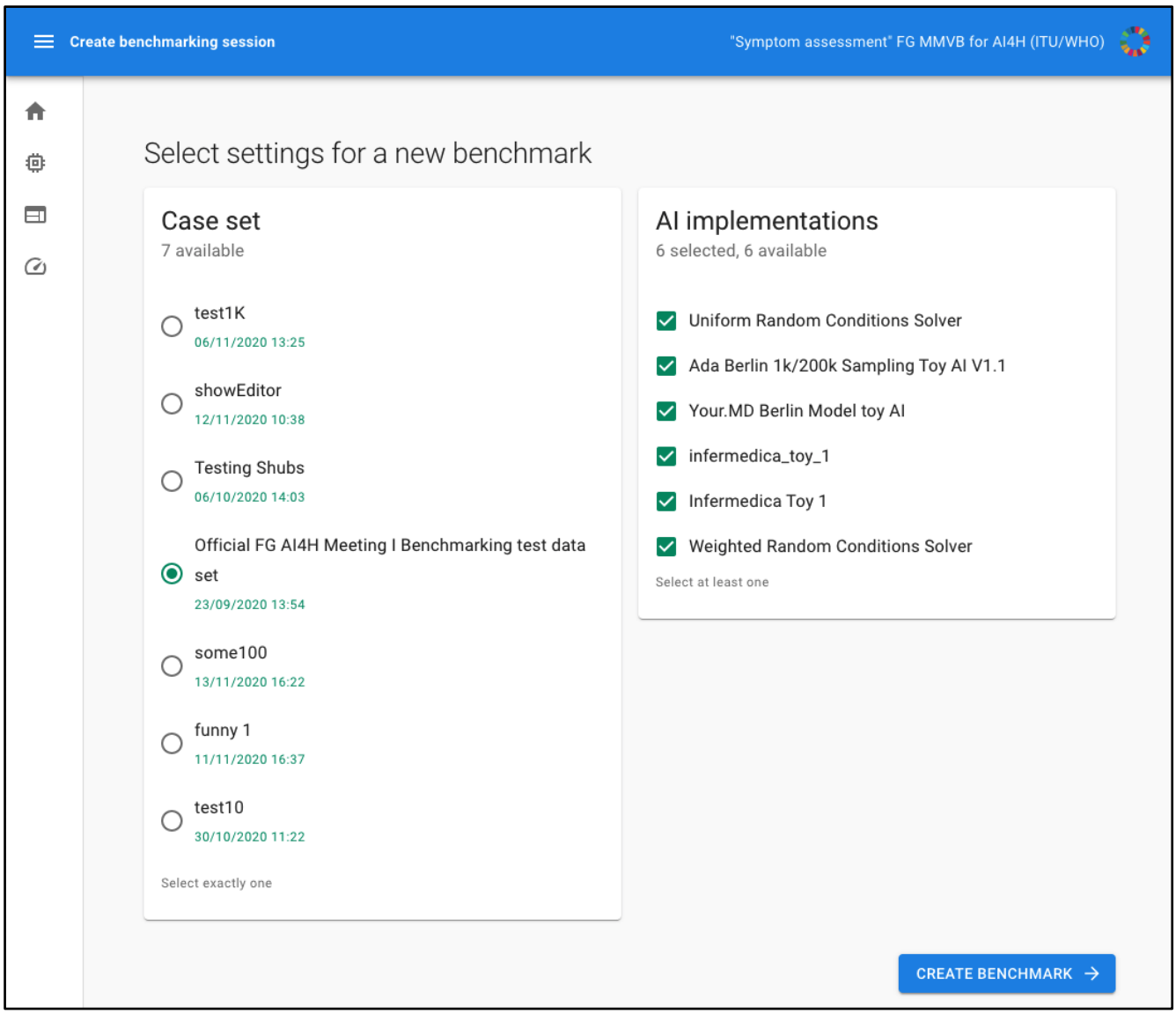


Figure 20 – MMVB 2.2 benchmarking session creation page

Running a benchmarking session

Once the benchmarking session was created, it could be run by clicking the play button. The benchmarking ran asynchronously on the server. While the benchmarking was running the benchmarking session showed the progress screen display in Figure 21 with a progress bar, number of completed cases, errors and timeouts.

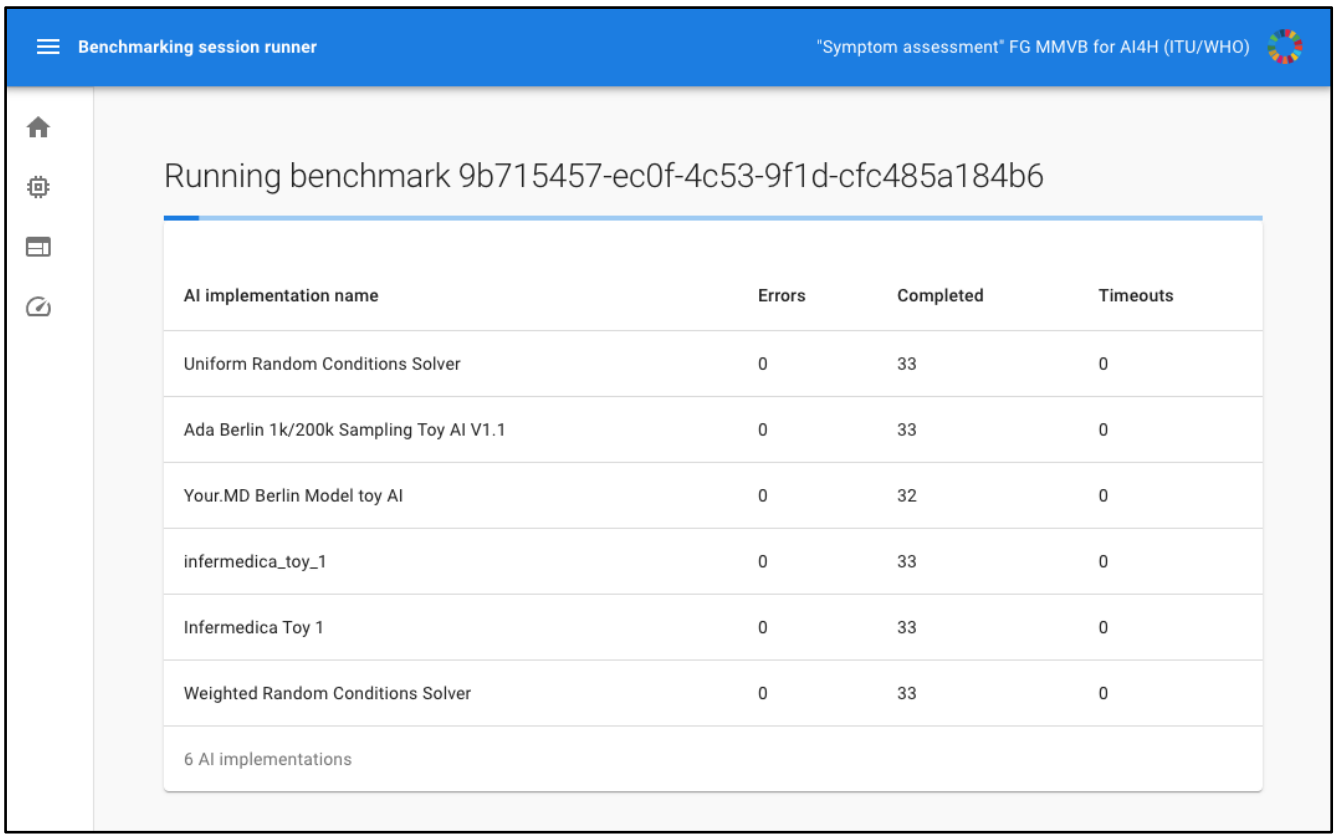


Figure 21 – MMVB 2.2 benchmarking session runner

Reviewing the benchmarking results

Once the benchmarking calculation was completed, the session showed the result screen displayed in Figure 22. The metrics have been calculated dynamically based on the stored responses of the AIs. While this might have been unusual for other AI competition platforms, for this topic group we needed the flexibility to drill down into a given context relevant to a stakeholder and recompute the metrics based on this filtered subset.

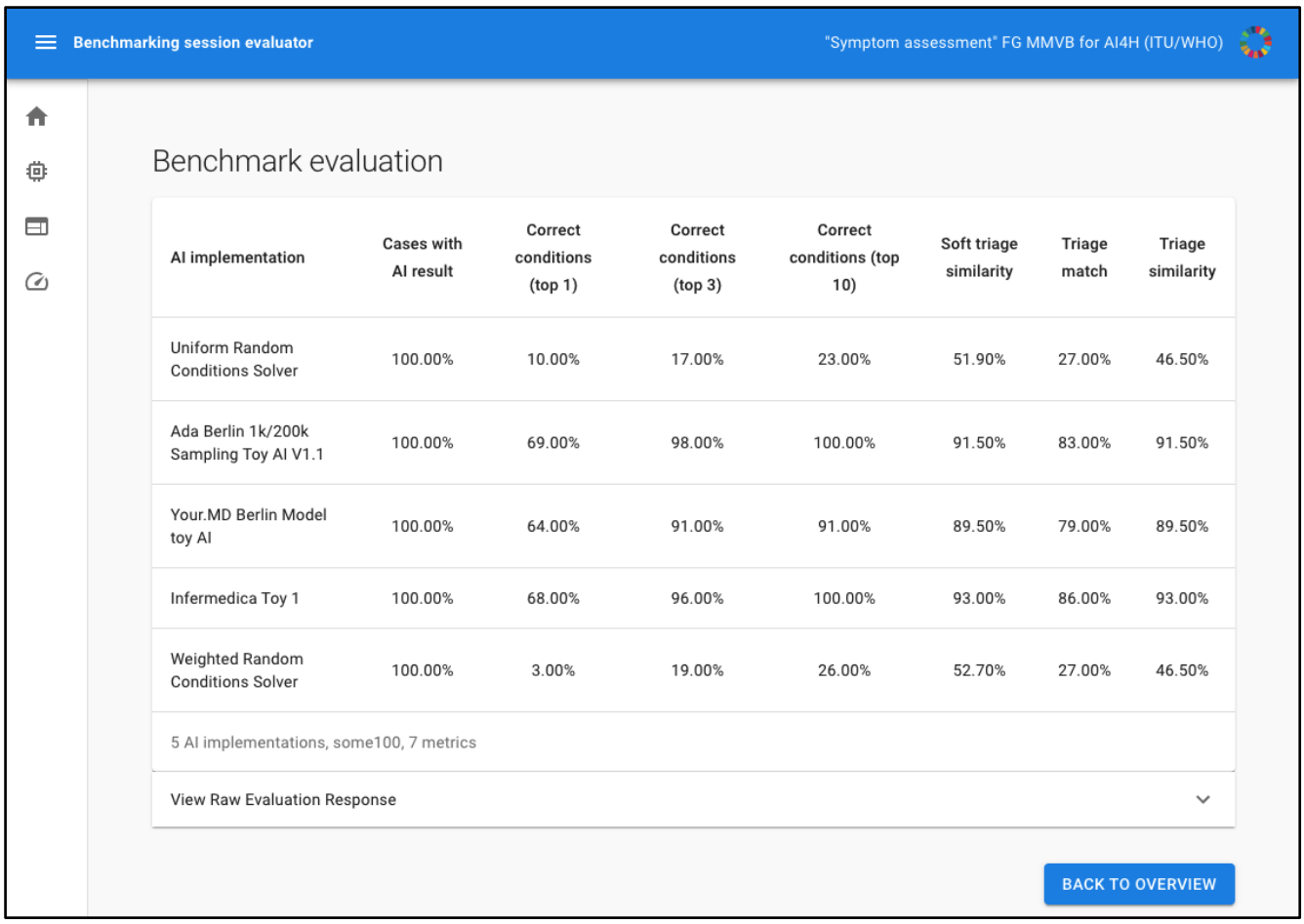


Figure 22 – MMVB 2.2 benchmarking result page

As for MMVB 1.0 for the MMVB 2.2 version there was also no scheduled benchmarking. Every developer could run a benchmarking session at any time for building their own toy-AI which helped both the development of the pipeline and the AIs. In contrast to MMVB 1.0 the step of submitting the toy AI to Yura Perov was also not necessary anymore as developers added theirs AIs using the self-registration feature.

6.1.2.3 AI input data structure for the benchmarking

The MMVB 2.2 used as input for the AIs a simplistic user profile, explicit presenting/chief complaints (PC/CC) and additional symptoms, findings and factors. The general case structure that was sent to a AIs can be seen in Figure 23. Since MMVB 1.0 we added an explicit field for the name of the AI implementation so that AI developers can host all AIs at the same systems and route the requests to the correct AIs by name. Table 15 shows the details of the different fields in the case structure.

```
{
  "caseData": {
    "otherFeatures": [
      ...
    ],
    "profileInformation": {
      ...
    },
```

```
    "presentingComplaints": [
      ...
    ]
  },
  "aiImplementation": "Company X Berlin Model toy AI"
}
```

Figure 23 – MMVB 2.2 General input case structure

Table 15 – MMVB 2.2 input data format

Field name	Example	Description

profileInformation	`"profileInformation": {`
	`  "age": 38,`
	`  "biologicalSex": "male"`
	`}`
		● General information about the patient
		● Age is unrestricted, however for the case creation it was agreed to focus on 18-99 years.
		● As sex we started with the biological sex "male" and "female" only

presentingComplaints	`"presentingComplaints": [{`
	`  "id": "0ef0...2d87",`
	`  "name": "Diarrhea (finding)",`
	`  "state": "present",`
	`  "attributes": [`
	`  {`
	`  "id": "7a0c...5d3c",`
	`  "name": "Time since onset",`
	`  "value": {`
	`  "id": "4395...90a23",`
	`  "name": "3 days - to 1 week",`
	`  "standardOntologyUris": [`
	`  "CUSTOM:102"`
	`  ]`
	`  },`
	`  "standardOntologyUris": [`
	`  "CUSTOM:1"`
	`  ]`
	`  }`
	`  ],`
	`  "standardOntologyUris": [`
	`  "62315008"`
	`  ]`
	`}]`
		● The complaints the user seeks and explanation/advice for
		● Always present
		● A list, but for the MMVB 2.2 always with exactly one entry
		● For MMVB 2.2 the PC now also contain attributes
		● In addition to MMVB 1.0 also standardOntologyUris (even if in this example it is only an ID and not an URI)

otherFeatures	"otherFeatures": [
	  {
	  "id": "356ae...a492",
	  "name": "Vomiting (disorder)",
	  "state": "unsure",
	  "attributes": [],
	  "standardOntologyUris": [
	  "422400008"
	  ]
	  },
	  {
	  "id": "b200...d5e8",
	  "name": "Dysuria (finding)",
	  "state": "present",
	  "attributes": [
	  {
	  "id": "7a0c...5d3c",
	  "name": "Time since onset",
	  "value": {
	  "id": "b505...ae52",
	  "name": "a year or more",
	  "standardOntologyUris": [
	  "CUSTOM:105"
	  ]
	  },
	  "standardOntologyUris": [
	  "CUSTOM:1"
	  ]
	  }
	  ],
	  "standardOntologyUris": [
	  "49650001"
	  ]
	  }
	],
		• Similar to the presenting complaints now with attributes and standard ontology identifiers

The MMVB 2.2 case data explicitly encoded the presence of each symptom. All symptoms had an explicit "state" attribute, which was responsible for information on whether a symptom is "present", "absent" or a patient was "unsure" about it.

6.1.2.4 AI output data structure

Similar to the input data structure for the benchmarking, this section describes the output data the AI systems are expected to generate in response to the input data. It covers the details of the data format, coding and error handling at the level of detail needed for an AI to participate in the benchmarking.

The case object described in the previous section was sent to the "/solve-case" context of the API endpoints specified by the AIs as a JSON post request payload. The expected response was a JSON object with the fields described in Table 16. It had not changed compared to MMVB 1.0.

Table 16 – MMVB 2.2 AI output structure

Field name	Example	Description

`conditions`	`"conditions": [`
	`{`
	`"id": "ed9e333b5cf04cb91068bbcde643",`
	`"name": "GERD"`
	`}`
	`]`
		• The conditions the AI considers best explaining the presenting complaints.
		• Ordered by relevance descending

`triage`	`"triage": "EC"`	• The triage level the AI considers adequate for the given evidence • Uses the same abbreviations defined by the London-model EC, PC, SC, UNCERTAIN

In addition to the "solve-case" endpoints the AIs have also been supposed to listen to the "health-check" endpoint. It was used during the benchmarking to make sure that the AIs have been ready to process cases. In the later MVB the benchmarking would pause if an AI is not responding anymore. The expected response for the health check was an JSON object like `{"data":"OK"}`. Every answer other than "OK" was considered an error.

6.1.2.5 Test data label/annotation structure

While the AI systems could only receive the input data described in the previous sections, the benchmarking system needs to know the expected correct answer (sometimes called 'labels') for each element of the input data so that it can compare the expected AI output with the actual one. Since this was only needed for benchmarking, it was encoded separately.

MMVB 2.2 relied on synthetic data sampled from the Berlin model. All cases have been stored as case sets in the MySQL database. Internally the case sets have been encoded similar to the cases sent to the AI but with the additional fields listed in Table 17.

Table 17 – MMVB 2.2 case with labels included.

Field name	Example	Description
`correctCondition`	`"correctCondition": {` *`"id"`*`: "2333...13c8",` *`"name"`*`: "GERD",` *`"standardOntologyUris"`*`: [ "http://snomed.info/id/235595009"`	• The correct condition i.e. in MMVB 2.2 the condition this case was sampled from. Note: the correct condition might not be the correct expected one given the evidence! For instance, in the

	] }	case of only headache a common cold is expected even if the case was a brain cancer case.
expectedCondition	Same as for correctCondition	• The conditions expected/accepted as top result for **explaining the presenting complaints** based on the given evidence. • A list, but only one entry for mono-morbid cases as it is the case for MMVB 2.2
impossibleConditions	Same as for correctCondition	• An optional list of diseases that must not be contained in the results e.g.,because have contradict sex e.g.,male diseases for females in vice versa
otherRelevantDifferentials	Same as for correctCondition	• Other diseases that should be present in the result as relevant differentials to rule out.
expectedTriageLevel	"expectedTriageLevel": "PC"	• The expected triage level (EC, PC, SC, UNCERTAIN)
name	*"name"*: "BPPV test case #1",	• The name of the case. This is especially helpful for cases created by doctors.
caseSets	*"caseSets"*: ["4354...3a75"]	• The list of case sets containing this case. • Only used for case set management.

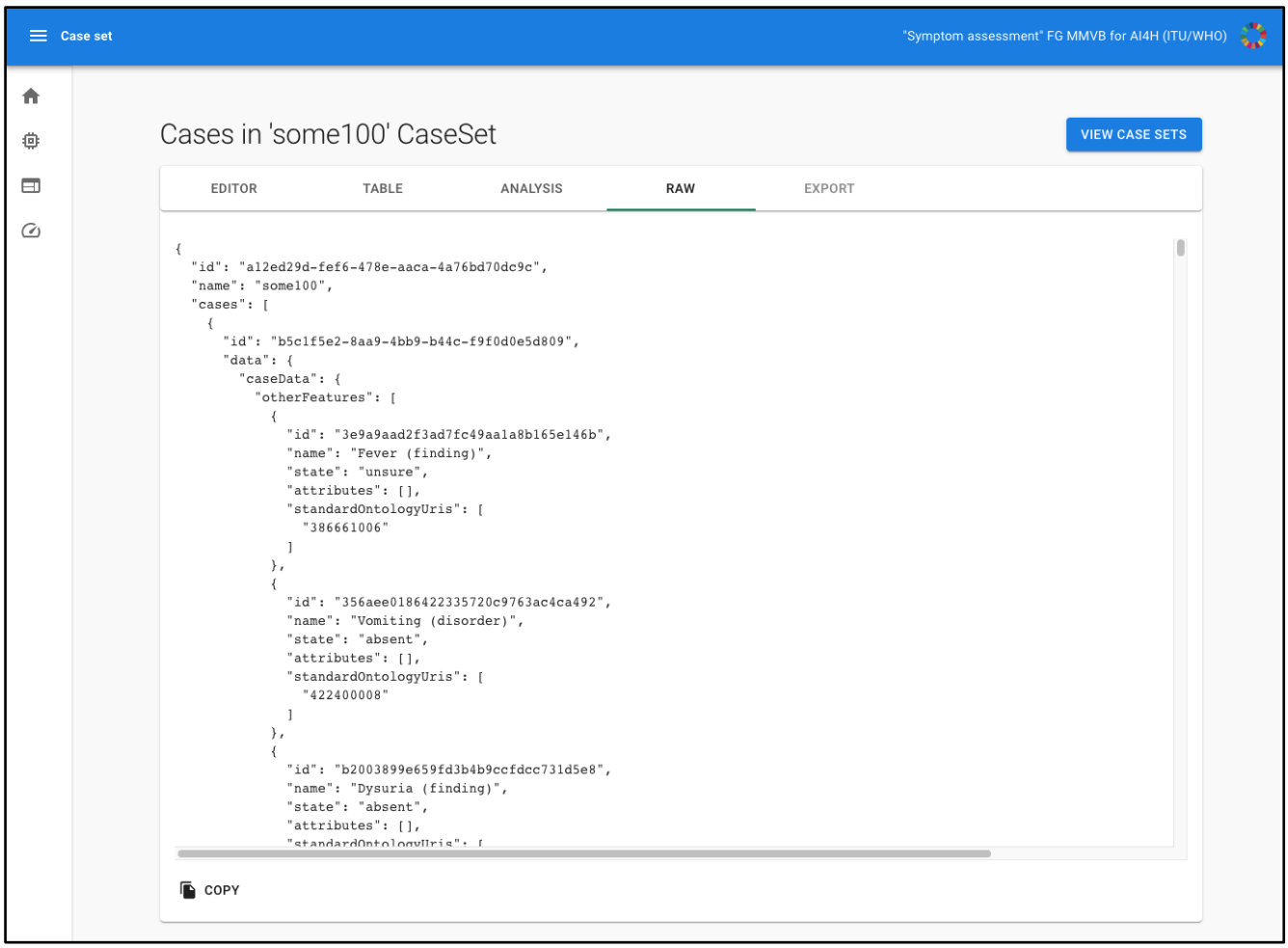
The overall case set structure can be seen in Table 18.

Table 18 – MMVB 2.2 overall case set structure

```
{
  "id": "f266e564-135f-4959-8a60-2f6d3f5cec9d",
  "name": "test10",
  "cases": [
    {
      "id": "29959f5c-c4e6-4341-9a1a-4802ce451629",
      "data": {
        "caseData": { ... },
        "metaData": {
          "name": "Synthesized case a5e425a2",
          "caseCreator": "MMVB Berlin model case synthesizer"
        },
```

```
        "valuesToPredict": {
          "correctCondition": {
            "id": "9d27455f69d907a7dad7bb471f3717a4",
            "name": "Appendicitis (disorder)",
            "standardOntologyUris": [
              "74400008"
            ]
          },
          "expectedCondition": {
            "id": "9d27455f69d907a7dad7bb471f3717a4",
            "name": "Appendicitis (disorder)",
            "standardOntologyUris": [
              "74400008"
            ]
          },
          "expectedTriageLevel": "EC",
          "impossibleConditions": [],
          "otherRelevantDifferentials": []
        }
      },
      "caseSets": [
        "f266e564-135f-4959-8a60-2f6d3f5cec9d"
      ]
    },
    ...
  ]
}
```


The frontend application offered a feature to preview and download all the case set data (see Figure 24).




```
{
  "id": "a12ed29d-fef6-478e-aaca-4a76bd70dc9c",
  "name": "some100",
  "cases": [
    {
      "id": "b5c1f5e2-8aa9-4bb9-b44c-f9f0d0e5d809",
      "data": {
        "caseData": {
          "otherFeatures": [
            {
              "id": "3e9a9aad2f3ad7fc49aa1a8b165e146b",
              "name": "Fever (finding)",
              "state": "unsure",
              "attributes": [],
              "standardOntologyUris": [
                "386661006"
              ]
            },
            {
              "id": "356aee0186422335720c9763ac4ca492",
              "name": "Vomiting (disorder)",
              "state": "absent",
              "attributes": [],
              "standardOntologyUris": [
                "422400008"
              ]
            },
            {
              "id": "b2003899e659fd3b4b9ccfdcc731d5e8",
              "name": "Dysuria (finding)",
              "state": "absent",
              "attributes": [],
```


Figure 24 – MMVB 2.2 case set raw viewer

6.1.2.6 Scores and metrics

As the MMVB 1.0 benchmarking, MMVB 2.2 also only used toy AIs and toy data and so the focus was still to have metrics for implementing the benchmarking in the first place. For this purpose, we implemented the same metrics:

- Cases with AI result (success rate)
- Correct conditions (top 1)
- Correct conditions (top 3)
- Correct conditions (top 10)
- Triage match
- Triage similarity
- Soft triage similarity

We decided to implement a new "Triage similarity (soft)" score such that if an AI respond to the triage with "unsure", the AI is given a triage similarity score higher than 0. The reason to introduce this score is to learn how to integrate "unsure" into the scoring calculations.

6.1.2.7 Test data set acquisition

Test data set acquisition included a detailed description of the test data set for the AI model and, in particular, its benchmarking procedure including quality control of the data set, control mechanisms, data sources and storage. For the MMVB 2.2 benchmarking iteration we used both synthetic data and case vignettes created by doctors.

6.1.2.7.1 Synthetic test data acquisition

The sampling was performed by the case synthesizer in the backend based on the CSV exported spreadsheet of the Berlin model[5].

The cell on the intersection between symptom's first row and a disease was a rough estimate of a link strength (captured by "x", "xx" or "xxx" labels where "xxx" stands for the strongest link) between a disease and a symptom. Each attribute state could also have a link to the disease. However, it was already conditioned on the presence of the symptom. Some symptom attribute states have been exclusive, meaning that only one attribute state can be "present". Other symptom attribute states have not been exclusive, meaning several states might have been present at the same time. The encoding for this can could be found in the “attributes-value sets” sheet. If a symptom was "absent" or "unsure", then no attributes or attribute states have been sampled.

In general, the case synthesizer first sampled a patient from a uniform sex distribution and uniform age distribution between 18 and 80 years. It also sampled the remaining factors obeying their sex dependencies. Based on this a condition that was not contradicting the factors was sampled based on their prior probability which was assumed linear to the number of “x” in the model. Based on the condition then the symptoms have been sampled based on their conditional probability which was defined as 30% times the number of “x” in the model. It was also sampled if the symptoms should have been part of the case, marked as “unsure” or omitted. In the last step the attributes have been sampled based on their conditional probabilities but also considering their multi-selectable / single-selectable type. For checking the sampling, we also implemented a first statistical tool showing the sex and age distributions (see Figure 25).

5 https://docs.google.com/spreadsheets/d/1dxzHFA8Rz2erN16dKKf8Sq9MffHiEWsHEsyDEtYW7dg/edit#gid=2083361879

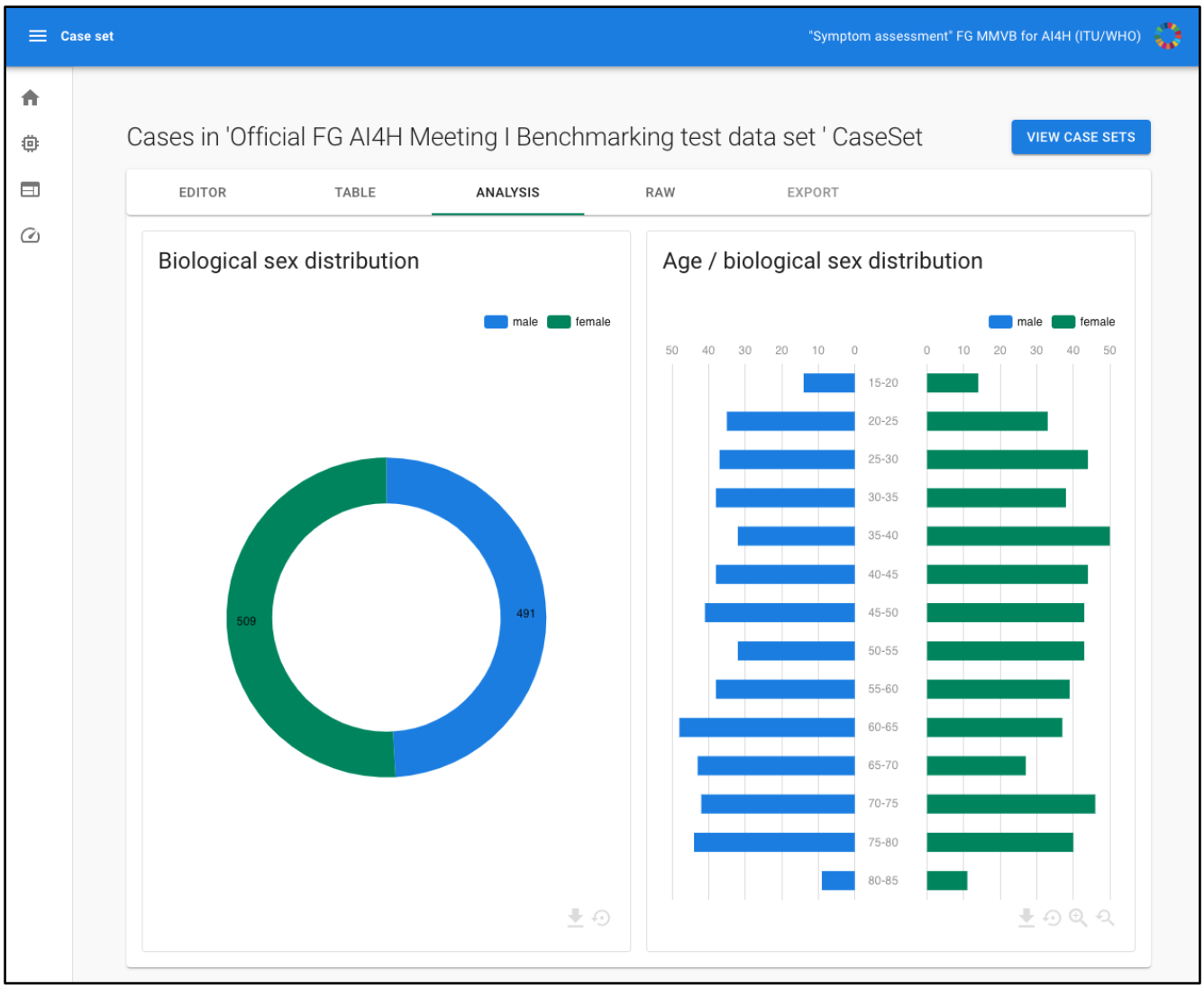


Figure 25 – MMVB 2.2 case set statistics view

6.1.2.7.2 Manual test data acquisition

The manual case creation was based on the annotation tool created for MMVB 2.2. It was a generic tool largely automatically generated from the Berlin model. Using this approach, it was made sure that only valid cases could be created by offering for instance only the available attributes if adding a symptom. This generic approach implied a trade-off between reusability (such as for other focus groups, as discussed in FG-AI4H-H-038-R01) and user experience. Figure 26 shows examples of the case annotation tool.

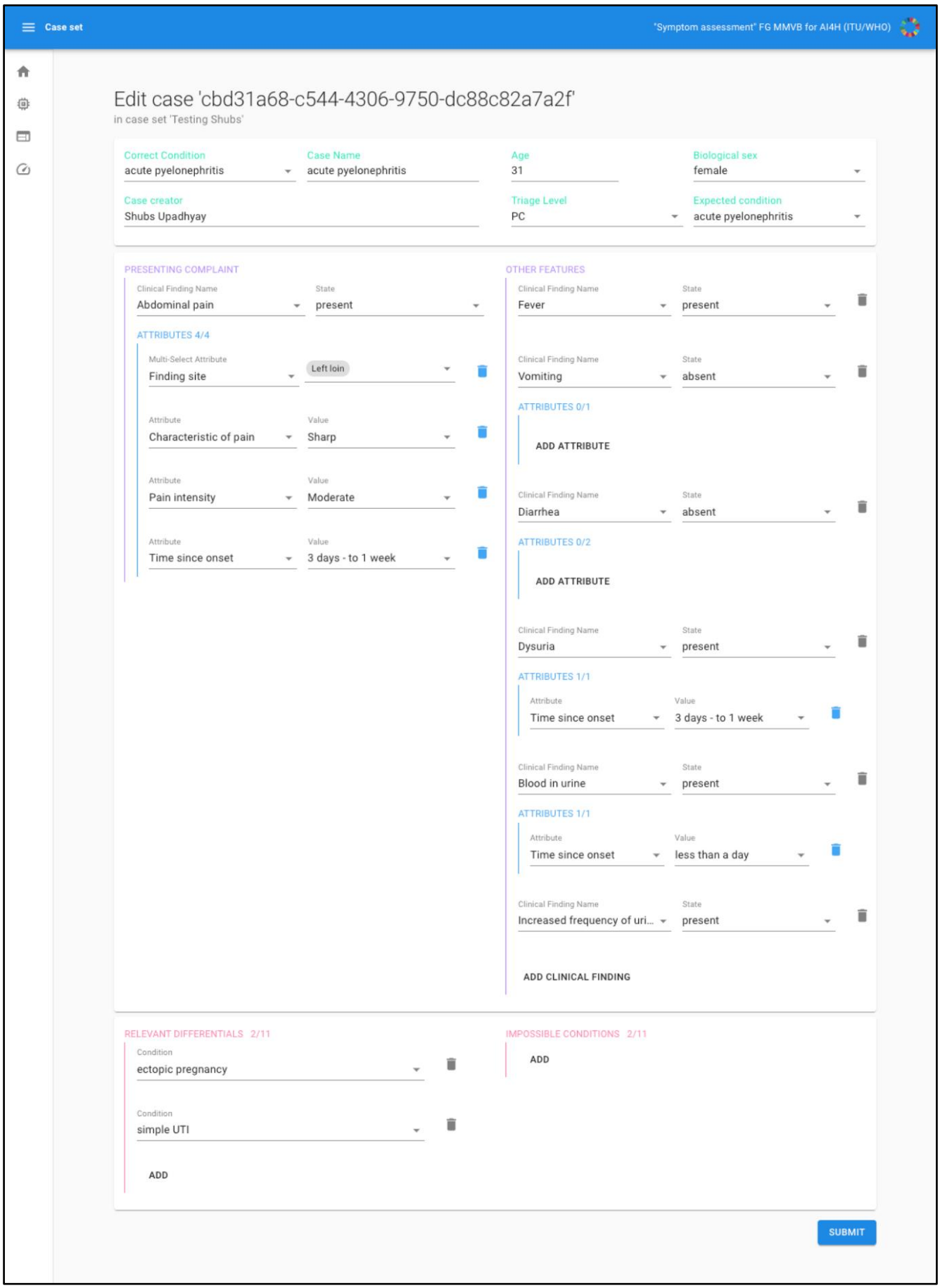


Figure 26 – MMVB 2.2 example of a case defined by a doctor using the case annotation tool

For the manual case creation, we also created cases annotation guidelines helping the doctors with what to consider when creating cases. The link to these can be found here - MMVB 2.2 Guidelines. Clinicians in the participating organizations then used these new guidelines to create a new set of cases for the MMVB 2.2 benchmarking.

The latest set of 11 cases has then been manually imported into the benchmarking system where they have been available as a case set for then benchmarking.

6.1.2.8 Data sharing policies

For the MMVB 2.2 iteration only synthetic cases and some cases manually created by the doctors in the topic group have been used. The cases have been highly specific to the MMVB benchmarking iteration and have not been based on real cases. The data was freely accessible through the freely online available benchmarking system instance.

6.1.2.9 Baseline acquisition

For the MMVB 2.2 assessing any baseline was out of scope.

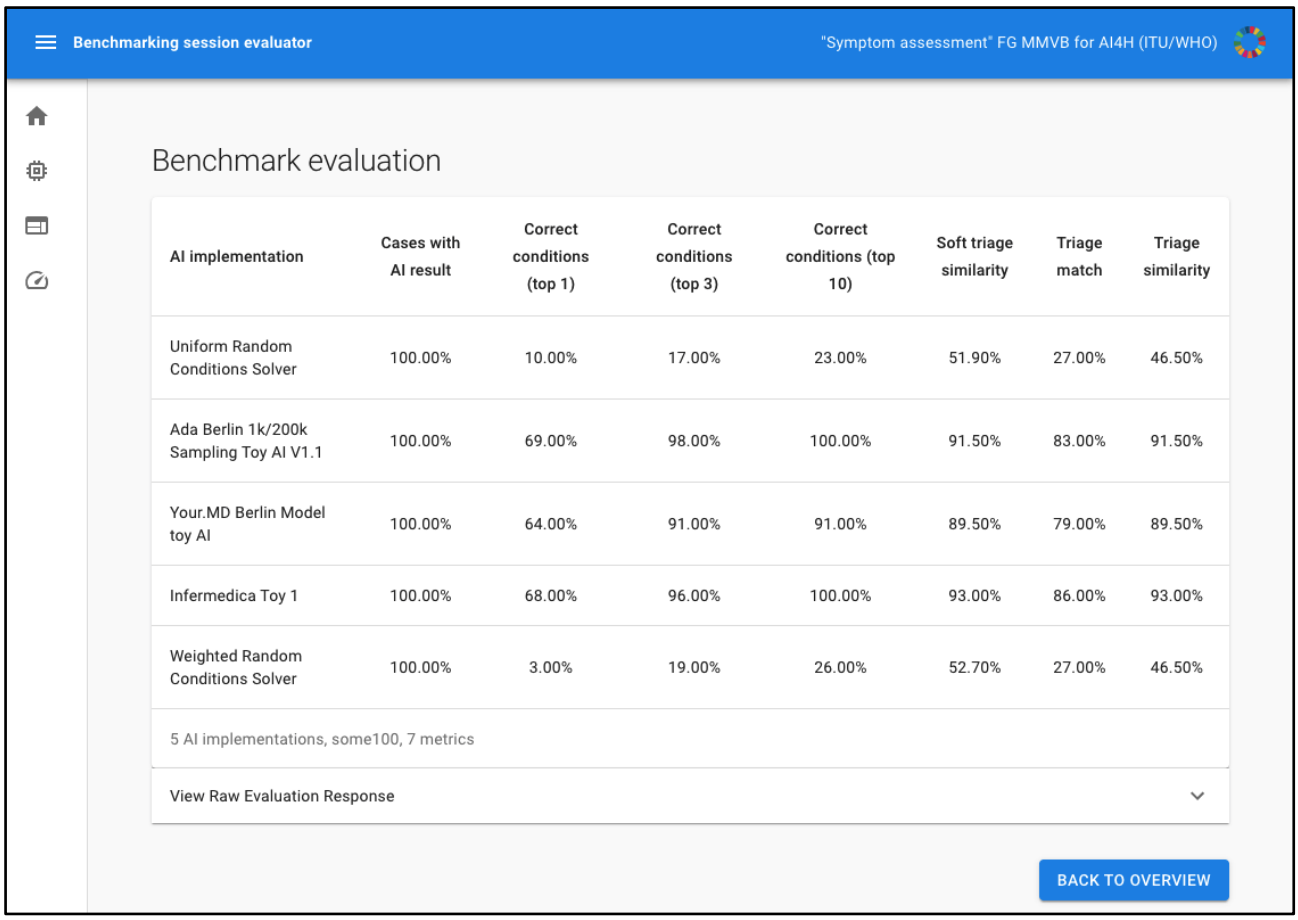
6.1.2.10 Reporting methodology

As the MMVB 2.2 used toy-AIs and toy-data, the only stakeholder interested in results was the focus group itself. The results have been documented in this TDD document and presented at the focus group meeting.

In contrast to the MMVB 1.0 version, the MMVB 2.x frontends did not contain any interactive drill-down feature yet.

6.1.2.11 Result

As this benchmarking iteration was only an intermediate development step, no final result was recorded, nor would it be of any practical relevance. As expected, the best performing toy-AI used a sampling approach based on the perfect knowledge of the domain model which would be asymptotically optimal and could not be outperformed. Figure 27 shows the results for a 100 cases test set.



AI implementation	Cases with AI result	Correct conditions (top 1)	Correct conditions (top 3)	Correct conditions (top 10)	Soft triage similarity	Triage match	Triage similarity
Uniform Random Conditions Solver	100.00%	10.00%	17.00%	23.00%	51.90%	27.00%	46.50%
Ada Berlin 1k/200k Sampling Toy AI V1.1	100.00%	69.00%	98.00%	100.00%	91.50%	83.00%	91.50%
Your.MD Berlin Model toy AI	100.00%	64.00%	91.00%	91.00%	89.50%	79.00%	89.50%
Infermedica Toy 1	100.00%	68.00%	96.00%	100.00%	93.00%	86.00%	93.00%
Weighted Random Conditions Solver	100.00%	3.00%	19.00%	26.00%	52.70%	27.00%	46.50%

Figure 27 – MMVB 2.2 benchmarking results for test set with 100 cases sampled from the Berlin model

6.1.2.12 Discussion of the benchmarking

The release of the MMVB 2.2 version concluded the implementation of the Berlin model with the more complex domain model also considering attributes and more detailed factors. The work included a complete rewrite of the frontend and backend application and also the implementation of a dedicated case annotation tool. The work on the benchmarking pipeline also underlined that in general the benchmarking with AI systems hosted by the participants is technically feasible. During the work on the MMVB the focus group started to pursue the idea of a focus group wide open code initiative (OCI) for a benchmarking platform. If feasible TG-Symptoms would adopt this platform for the benchmarking of AI-based symptom assessment systems. This would however shift the focus of our work on the parts that are specific for our topic group, namely:

- A joint ontology for encoding case vignettes for benchmarking.
- The case annotation and creation tool that integrated in the OCI's annotation package.
- The scores and metrics specific for AI-based symptom assessment.
- The interactive result drill-down and filtering system provides the benchmarking results for a specific context relevant to a stakeholder.
- The approach of distributed AIs hosted by the participants.

The first point is the largest and most important open task for implementing the first real minimal viable benchmarking so that we will focus all our work on that and will not implement another benchmarking iteration until this work is completed.

6.1.2.13 Retirement

The MMVB 2.2 was still an intermediate development system. Frontend, backend and the annotation tool will be hosted as a demo and topic group internal reference system until a new benchmarking iteration is available. The code of the benchmarking system is available in GitHub at:

https://github.com/FG-AI4H-TG-Symptom/fgai4h-tg-symptom-benchmarking-frontend

and

https://github.com/FG-AI4H-TG-Symptom/fgai4h-tg-symptom-assessment-mmvb-backend

The Berlin Model used for generating the synthetic test cases are available at:

https://docs.google.com/spreadsheets/d/111D40yoJqvvHZEYI8RNSnemGf0abC9hQjQ7crFzNrdk/edit#gid=575520860

6.1.3 MMVB 3.0 - 3.1

6.1.3.1 Overview

After finishing the MMVB 2.x iterations the most important step to explore was how to encode benchmarking cases in a way that could be processed by all participating health AIs. To optimally support symptoms a patient could self-report or answer, most AI developers created their own ontology tailored to meet this specific requirement, rather than using main-stream ontologies which tend to have a strong bias towards the findings medical professionals collect. This in turn lead to a situation where the case structures of the different AIs are not interoperable and hence the same encoded case could not be processed by different AIs. For the case studies found in literature this problem was circumvented by real users, nurses or doctors entering cases manually into the different systems – an approach that would have not scaled for automated benchmarking the topic group aims for.

For the MMVB 3.x versions we therefore started the work towards a joint case encoding approach. The discussions inside the topic group showed a few points:

- There was good reason why companies did not use existing ontologies for encoding their cases, including: incomplete coverage of patient-friendly self-reportable symptoms, heavy bias towards the doctor's view, ambiguity, redundancy, imprecision, inconsistent level of granularity, inconsistent/incoherent modelling of symptom-attributes.
- The cost (time and money) of building and maintaining a new ontology for encoding benchmarking cases would be prohibitive for the topic group or the focus group.
- The cost for mapping to any benchmarking ontology to the company ontology of benchmarking would be high.
- Mid-term all AIs could be directly benchmarked on textual case vignettes.
- Independent of the approach used for encoding factors, findings and attributes we would also need a standardized container for representing the benchmarking cases.
- The open code initiative (OCI) of the focus group has reached a majority where we should explore if switching to their benchmarking platform would be an option.
- The audit group reached a point where we could work towards integrating TG-symptom into it.

In the light of these points the topic group agreed for the MMVB 3.x iteration

- to explore the possibility to use SNOMED for encoding cases
- to explore FHIR as container format for benchmarking cases

- to draft a new case annotation tool supporting SNOMED
- to use the annotation tool for collecting evidence on the feasibility of encoding cases using SNOMED
- to work with the OCI on integrating the annotation tool with the annotation package
- to work with the audit group on the audit process and benchmarking platform
- to migrate our benchmarking from our own platform to the OCI platform
- perform an MMVB test benchmarking using the existing synthetic data in combination with a FHIR/SNOMED mapping and selected test AIs that could process SNOMED encoded FHIR cases.

There was also the expectation that a first MVB benchmarking could be possible with these changes, but this point was not reach before the end of the focus group.

6.1.3.2 Benchmarking methods

6.1.3.2.1 Benchmarking system architecture

The MMVB 3.x benchmarking consists of four major components, the annotation tool's frontend, the annotation tools backend, the AIs to benchmark and the benchmarking platform. The latter was developed by the OCI and is outside the scope of this document. They will be described in the following sections.

Annotation tool frontend

The most prominent component was case annotation tool frontend application that allowed doctors to encode benchmarking cases using the SNOMED ontology. As starting point it offered a case list view (Figure 28) showing all the cases entered so far with action buttons for editing, deleting and adding new cases. The page was only needed since the annotation tool was not integrated with the annotation package yet, which then would have taken care of the case storage and annotation task handling.

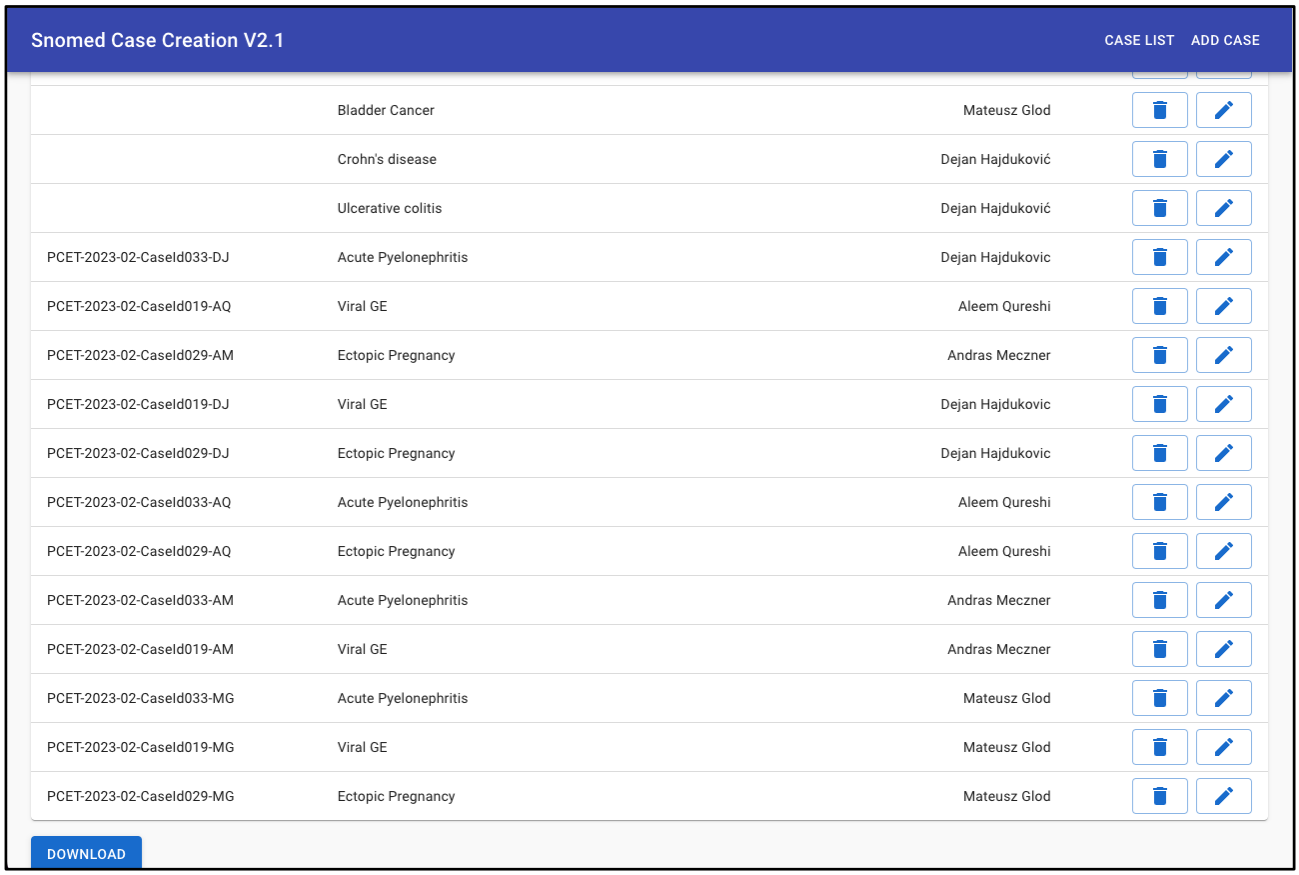


Figure 28 - MMVB 3.x annotation tool case list

For creating a new case or editing an existing one the app navigates to the case editor shown in Figure 29. It consisted of a panel for case metadata details like the name of the case, the author, the expected AI outputs and patient details like age and sex. In the right side-panel the annotators could leave notes and comments which during the development have been used for collecting feedback about the tool itself and the annotation process. The panel also contained the case-vignette text that the annotator was supposed to transcribe into a SNOMED case or a task description for instance for creating a new case for a given condition. The central part of the editor contained the symptom lists for presenting complaints, present symptoms and absent symptoms. There was no fourth list for unsure symptoms.

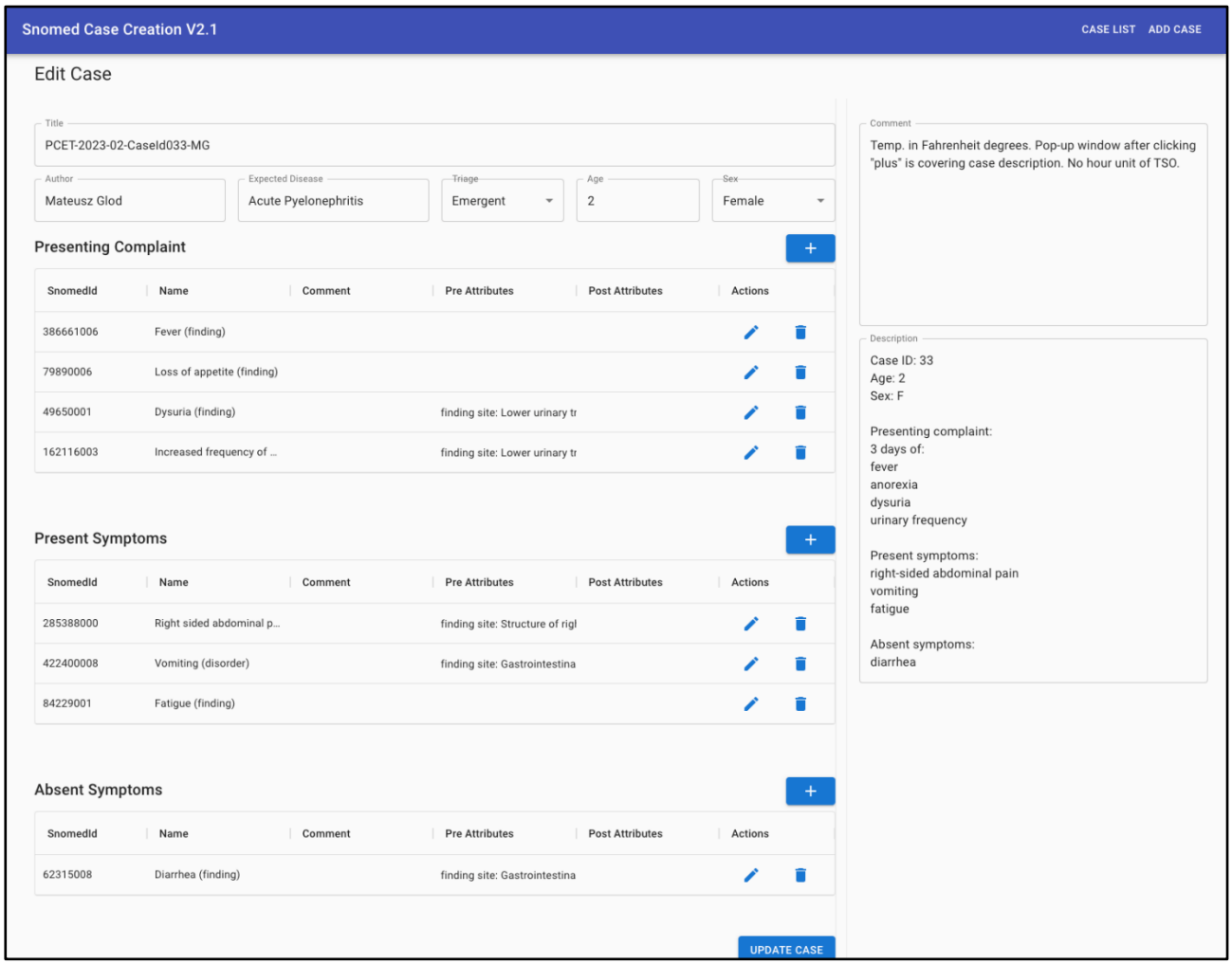


Figure 29 - MMVB 3.x annotation tool case editor showing case #33 from the parallel case encoding test

If the button for adding a symptom was clicked the annotator could search the SNOMED concept space using the panel shown in Figure 30. Beside the search results it also listed the ancestors and children of the selected finding so that the annotator could fast navigate to the right symptom.

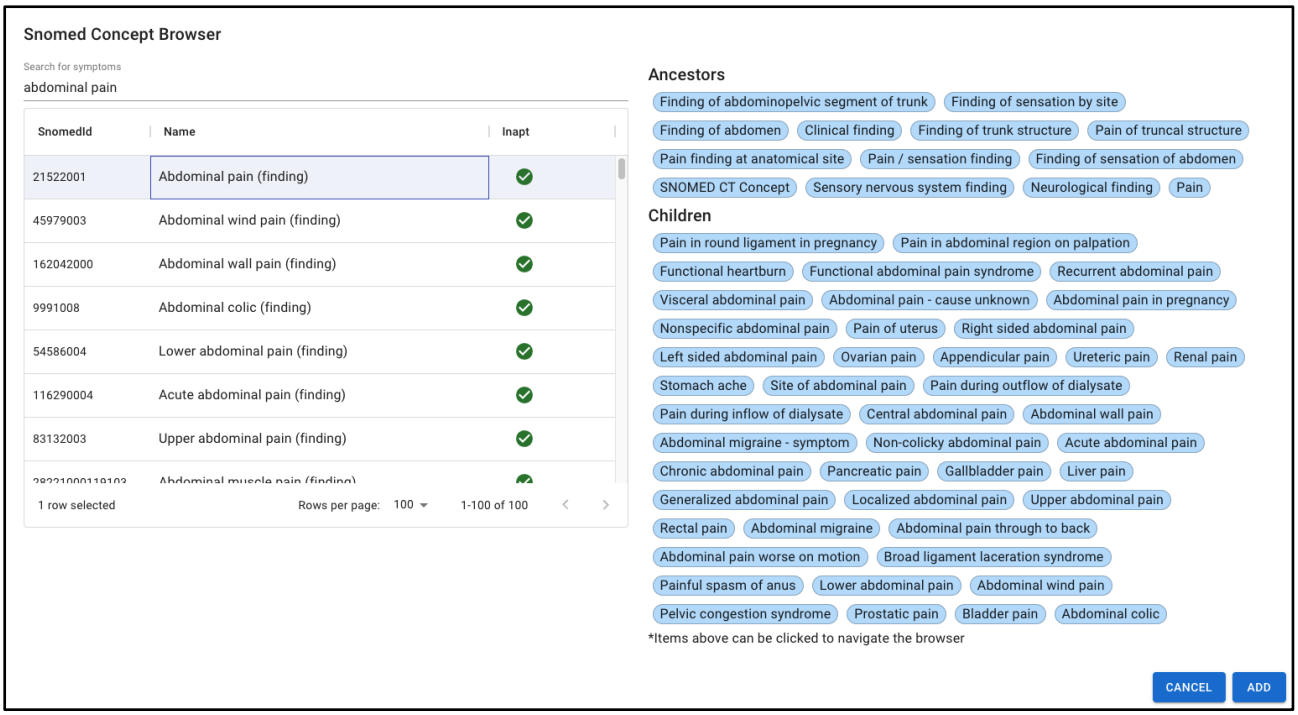


Figure 30 - MMVB 3.x dialog for adding a finding to a case.

Once a symptom was added, its attributes could be specified by opening the screen shown in Figure 31. If offered explicit controls for defining the time since onset which was considered important enough for self-assessment systems for being implemented as an explicit number + unit control. The panel also offered an editor for the attributes a finding supported. For this prototype we explicitly restricted the attributes to severity, clinical course and finding site. Finding site was special since as only attribute it had thousands of possible states which have been pre-coordinated in most cases. For instance, does “abdominal pain” already imply that there was a “pain” with the location narrowed down to the “abdomen”. The annotator could then use the remaining degree of freedom to further specify where inside the abdomen the pain was located e.g.,the “right lower quadrant” which would have been characteristic for an appendicitis.

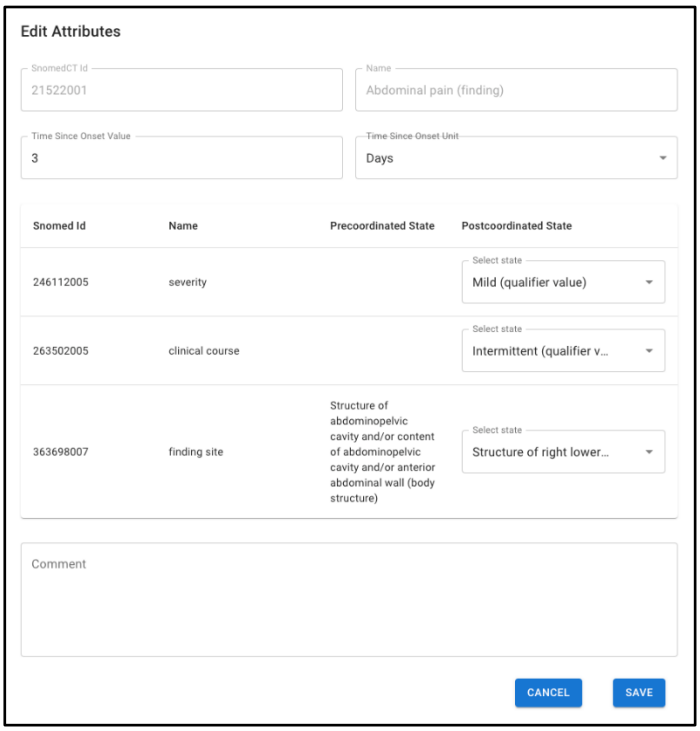


Figure 31 - MMVB 3.x annotation tool panel for editing finding attributes.

The annotation tool was implemented in JavaScript using react, material-ui and axios. It was considered a prototype for evaluating the feasibility of a SNOMED encoding tool and production grade quality in terms of architecture, stability and documentation was out of scope.

A more detailed description of how to use the annotation tool can be found in Annex D.

Annotation tool backend

The frontend application was relying on the REST API served by a corresponding backend application. It took care of the communication with an instance of Snowstorm[6], an open-source terminology server with special support for SNOMED CT. We used several different servers, but the latest version used directly the server[7] hosted by IHTSDO. The backend also provided create/update/remove functionality for benchmarking cases. Those cases have been stored in a mongoDB. We also implemented mechanism for marking findings as inappropriate for case modelling and computed usage statistics so that doctors could see which symptom have been used in cases to reduce variation in symptom usage. This feature however was not implemented in the latest frontend version. Any statistics would need to be reimplemented with the migration to the OCI's annotation package since the annotation tool would not use any own storage anymore. The API endpoints that have been used by the latest frontend implementation are listed in Table 19.

Table 19 – MMVB 3.x annotation tool backend API interface.

API call	Description
`GET /allCases`	Returns the list of all case stored in the database. The returned JSON object contains entire case objects, which would not scale beyond a proof of concept.

[6] https://github.com/IHTSDO/snowstorm
[7] https://browser.ihtsdotools.org/snowstorm/snomed-ct

`POST /addCase`	Adds the case given in the body as new case. Returns the newly created case (now containing also an id).
`GET /case/:id`	Return the case with the given case id.
`DELETE /case/:id`	Deletes the case with the given case id.
`PUT /case/:id`	Updates the case with the given case id with the case provided in the body.
`GET /search-symptoms/:query`	Returns symptom search results for the given search query. Search results are restricted to sub-classed of "<404684003 | Clinical finding (finding)".
`GET /ancestors/:conceptId`	Returns the ancestors of the SNOMED concept with the given concept id.
`GET /children/:conceptId`	Returns the children of the SNOMED concept with the given concept id.

The backend was implemented in the same project as the frontend using JavaScript/nodeJS. As database for storing cases, we used a mongoDB via mongoose. The backend was hosted by one of the topic group members inside their infrastructure.

Benchmarking system

For MMVB 3.x we migrated from our own benchmarking platform to the platform developed and hosted by the "FG-AI4H Open Source Project" / „FG-AI4H Assessment Platform" initiatives of the focus group, aiming to take care of the entire benchmarking pipeline from data acquisition, data storage, data annotation, evaluation to reporting. The details of the project can be found at their website[8].

For the MMVB 3.x benchmarking iteration we worked on an integration with the annotation package and on implementing a symptom assessment challenge for the benchmarking system the OCI platform. The corresponding project can be found in the aiaudit-org github repository[9]. It was derived on a template that was then adjusted to benchmarking symptom assessment systems. We used the file-based benchmarking approach since at the time of development there was not support for docker images that could wrap cloud hosted AI implementations. It is important to notice that in contrast to most other topic groups symptom assessment AIs are commercial products where the business constraints not allow to submit the AIs as containers containing all the company IP – a trend that this also underlined by the emergence of LLM models where the developers, even if this would be technically possible for some models, cannot submit their models for IP reasons.

Symptom assessment AIs

The symptom assessment AIs have been hosted in by the AI developers in their individual cloud infrastructure. For the benchmarking they exposed a dedicated REST API endpoint. For MMVB 3.x the AIs had to implement the API endpoints listed in Table 20. Behind the interface the system architecture and hosting details have been up to the AI developers.

[8] https://dev.azure.com/mllabai/fg-ai4h/_wiki/wikis/FG-AI4H-Assessment-Platform.wiki/13/Reporting-Package-(RP)
[9] https://github.com/aiaudit-org/trial-audits-team-a-tg-symptoms

Table 20 – MMVB 3.x AI API interface.

API call	Description
`GET /solve-case`	Taking a case according to JSON format already used for MMVB 2.x described in Table 21 and returning the result in the JSON response described in Table 24.
`GET /solve-fhir`	Taking a case according to the new FHIR JSON format described in Table 22 / Table 23 and returning the result in the JSON response described in Table 24 which is the same for both endpoints.
`GET /health-check`	Endpoint signalling the health state and the connection to an AI during the benchmarking.

6.1.3.2.2 Safe and secure system operation and hosting

Safe and secure operation and hosting have been out of scope of the MMVB 3.x benchmarking iteration. The annotation tool and the individual AIs have been hosted by topic group members in their own cloud infrastructure which provided basic safety and security according to their company guidelines. The secure operation and hosting of the benchmarking system were the responsibility of the OCI.

6.1.3.2.3 Benchmarking process

The benchmarking runs conducted with MMVB 3.x focused on testing the integration with the audit benchmarking of the focus group's open code initiative and to see if the benchmarking works for AIs enhanced with the /solve-fhir API endpoint accepting FHIR encoded benchmarking cases. The process was therefore not representative for a realistic MVB benchmarking run. The concrete process involved the following steps:

AI update and FHIR AI implementation

For the benchmarking we asked the AI developers to update their toy-AIs and double-check that they are still online. In addition to this we implemented the FHIR transcription script converting MMVB 2.x benchmarking cases into the agreed upon FHIR format. Based on this the "Ada Sampling Toy AI" was testwise extended by the /solve-fhir API endpoint.

Benchmarking data set generation

For the benchmarking we used the existing "Official FG AI4H Meeting I Benchmarking test data set (71e9c086-2d8f-4cf9-bbe4-106117c95c5c)" sampled from the Berlin Model for the MMVB 2.x benchmarking to have a baseline for the numbers we expected from the MMVB 3.x audit benchmarking. The data set was downloaded and added to the audit trial script project.

Audit script

For setting up a challenge in the benchmarking platform the topic group implemented an audit script based on a template provided by the OCI together with online documentation and online support-sessions. Beside going though file by file and adjusting the template to the topic group's needs, the main task was to implement the computation of the topic group specific metrics based on the labeled benchmarking data and the AI submission.

Challenge setup

Once the audit script was locally running it could be packaged and used to setup a new challenge in the audit system[10] by uploading the packaged audit-script. The challenge then needs to be approved by an OCI admin.

Generating AI input files

For the benchmarking the case set generated by the MMVB 2.x system needed to be converted into devsplit files and testsplit files - with annotations for the later comparison a metric computation and without for the corresponding AI input files. For the file based benchmarking this was done locally using a corresponding "mmvbAuditDataGenerator" helper script developed implemented by the topic group.

Collecting AI response files

The same script also iterated all AIs configured in this script, sent them the cases and recorded the response in AI submission file format.

Submitting AI responses to the audit system

The AI response have then been uploaded as AI submissions to the audit-platform. Since this was not a formal benchmarking the AI submissions have all been uploaded by the topic driver as dev-submissions. The audit system then in the background scheduled tasks for evaluating the submissions.

Reviewing results

Once the results have been computed, the topic driver could look them up at the corresponding results page.

6.1.3.2.3.1 Audit group cooperation

The outlined process primarily served the assessment of the technical feasibility of integrating with the OCI benchmarking platform. For defining the real benchmarking/audit process we started the cooperation with the focus group's audit group. The cooperation was coordinated by a dedicated TG-Symptom audit group consisting of experts for the relevant fields.

Right from the beginning it was clear that the audit of symptom assessment systems significantly deviated in its needs and requirements from most of the more machine learning centric topic groups. Accordingly, the focus was more on learning about these difference and less on conducting an audit according to the schedule.

This following sections address the challenges and learnings of the trial audit for TG-Symptoms. More information on the trial audit can be found in the playbook for the trial audit in DEL7.3.

[10] https://health.aiaudit.org/web/challenge-create

Audit Verification Checklist

At the beginning of the trial audit, the audit team was provided with a template for the Audit Verification Checklist (derived from document FGAI4H-J-038). The checklist was focusing strongly on machine learning methods for health applications. Symptom checkers, however, are primarily knowledge-based models that follow different development and evaluation processes than classic machine learning (ML). Therefore, the audit checklist for symptom checkers was modified completely. The process included the initial development and formulation of AI audit questions covering requirements for knowledge-based models and receiving feedback from the TG drivers/two symptom checkers (Ada Health and Healthily). This feedback loop was supportive of refining and improving the auditing process for symptom checkers. The formulation of precise questions for the auditees is crucial in order to receive the required information.

Regulatory Aspects

The task of designing an auditing process for symptom checkers can be challenging due to the wide range of possible use cases. Depending on the user and the user environment, different regulatory considerations and requirements can apply. After discussions, the audit team focused on the self-assessment use case. Other use cases include medical professionals as users and clinical environments. Additionally, symptom checkers can have different tasks like pre-clinical triage, possible causes and treatment advice. Those tasks require different types of data making them more complex (see Chapter 5.1.5.3), so other regulatory dimensions compared to ML must be included.

Evaluation

A significant part of the checklist comprises the model development. In ML, this process usually includes data collection, cleaning and pre-processing, as well as mostly data-driven model training. AI-based symptom assessment systems are not solely data-driven and often rely on content created by medical experts through reviews of literature, studies and data. Therefore, the model development process had to be adjusted for symptom assessment. Ethical considerations regarding the resources remained mostly the same (e.g.,addressing privacy and bias).

The evaluations and metrics are not comparable between regular ML and symptom checkers. While there are common standards for ML evaluation (e.g.,cross-validation, train-validation-test data splits) and metrics (e.g.,accuracy, ROC-AUC), AISAs can be evaluated on multiple aspects (see all details in chapter 5.1.5). The checklist mainly considered the performance of AISAs regarding their output, as described in chapter 5.1.7 . Other performance measures (see chapter 5.1.7.9) were not considered. Another difference to ML is the use of test data. In ML, the data used for evaluation has to follow the same structure as the training data (same type, similar features). Knowledge-based AISAs are based on information from literature, data, studies and experts, whereas performance is measured with clinical vignettes and other considerations that apply.

Scoring

Ada Health GmbH and Healthily completed the audit verification questionnaire for the trial audit and the answers were scored by the audit team in order to test the applicability of the audit questions. For some questions, the answers should be assessed by comparison with existing standards. It was not possible to find established standards for some of the items. Topic Group-Symptoms provides some suggestions regarding performance and accuracy metrics and special considerations in their topic description document (see chapter 5.1.7). Current benchmarking performances from the topic group Symptoms could not be assessed at this point in time.

In summary, the developed audit questionnaire should be considered as a starting point for conducting any audit. It should be followed by an analysis of relevant documentation and interviews with the responsible personnel in the auditee institution. Depending on the desired assurance level of the audit and the auditor's access to information, the TG-symptom audit group would consider a review of the code and data appropriate.

6.1.3.3 AI input data structure for the benchmarking

For the MMVB 3.x we used two different models for the AI input data. To enable joint benchmarking of AIs already supporting FHIR encoded cases and the unchanged MMVB 2.2 ones.

Table 21 – Example of the case without FHIR encoding – similar to the MMVB 2.2 case format.

Field name	Example	Description
caseId	`"id": "e728...660",`	• Random UUID generated for each case

profileInformation	`"profileInformation": {`
	`  "age": 38,`
	`  "biologicalSex": "male"`
	`}`
		• General information about the patient
		• Age is unrestricted, however for the case creation it was agreed to focus on 18-99 years.
		• As sex we started with the biological sex "male" and "female" only

presentingComplaints	`"presentingComplaints": [`
	`  {`
	`  "id": "1402…2d82",`
	`  "name": "Weight loss (finding)",`
	`  "state": "present",`
	`  "attributes": [],`
	`  "standardOntologyUris": [`
	`  "89362005"`
	`  ]`
	`  }`
	`],`
		• The complaints the user seeks and explanation/advice for
		• Always present
		• A list, but for the MMVB 3.x always with exactly one entry
		• Attributes are supported but are empty in this example
		• standardOntologyUris for MMVB 3.x – in the example it corresponds this SNOMED concept

otherFeatures	`[`
	`  {`
	`  "id": "f531...6fed",`
	`  "name": "Abdominal pain (finding)",`
	`  "state": "present",`
	`  "attributes": [`
	`  {`
	`  "id": "9124...e844",`
	`  "name": "Finding site (attribute)",`
	`  "value": {`
		• Similar to the presenting complaints
		• State can also be „unsure" and "absent"

	"id": "dd20...c8af", "name": "Epigastric region structure (body structure)", "standardOntologyUris": ["27947004"] }, "standardOntologyUris": ["363698007"] }, ...] }, ...]	

Since we were testing if FHIR could be used as container format for SNOMED encoded cases, we extended the TG-symptom audit trial OCI project by an FHIR converter. That could convert the cases on the MMVB 2.2 format into FHIR.

Since FHIR covers a broad range of use-cases and clinical setups there have been several options for the encoding of benchmarking cases. In several workshops we analysed the different structures provided by the FHIR specification and decided for a fully self-contained FHIR case using a top-level “bundle”[11] resource containing “observation"[12] resources for all the symptoms as well as the expected conditions and expected triage result. The format specified means to represent present, absent and skipped symptoms as well as attribute expressions. We also included a “patient” resource but explicitly decided against using it for the representation of age and sex.

Table 22 – Top-level structure of a FHIR encoded benchmarking case.

Field name	Example	Description
resourceType	“resourceType”: “Bundle”,	● The FHIR resource type „Bundle” we have chosen as container for cases
id	“id”: “case-1”,	● Random UUID generated for each case
entry	“entry”: […]	● The array with the actual bundle entries for e.g., symptoms (see next able)

[11] https://build.fhir.org/bundle.html
[12] https://build.fhir.org/observation.html

Table 23 – FHIR encoded input case

Entry type	Example	Description

patient	{
	"fullUrl": "Patient-",
	"resource": {
	"resourceType": "Patient",
	"identifier": [
	{
	"value": "Patient-...14ecade47"
	}
	]
	}
	}
		● Container for assessment-relevant patient details
		● Currently only an empty placeholder
		● Contains a random patient-id

age	{
	"fullUrl": "Age",
	"resource": {
	"resourceType": "Observation",
	"status": "final",
	"code": {
	"coding": [
	{
	"system": "http://snomed.info/sct",
	"code": "424144002",
	"display": "Age"
	}
	]
	},
	"valueInteger": 78
	}
	}
		● Describes the age for patient as integer value in years.

gender / sex	{
	"fullUrl": "Gender",
	"resource": {
	"resourceType": "Observation",
	"status": "final",
	"code": {
	"coding": [
	{
	"system": "http://snomed.info/sct",
	"code": "248153007",
	"display": "Male"
	}
	]
	}
	}
	}
		● Instead of a key-value assignment it directly uses e.g.,“female” as it implies that it is a “Finding related to biological sex (finding)” via its superclass
		● (female would be 248152002)

presenting complaint	{
	"fullUrl": "PresentingComplaint-0",
	"resource": {
	"resourceType": "Observation",
	"status": "final",
	"category": [
	{
	"coding": [
	{
		● Presenting complain represented as “Observation”
		● Marked as “Chief complaint“ using a corresponding SNOMED code (33962009)

	"system": "http://snomed.info/sct", "code": "33962009", "display": "Chief complaint" }] }], "code": { "coding": [{ "system": "http://snomed.info/sct", "code": "21522001:{363698007=27947004,364625007=410709000,406127006=255604002,CUSTOM:1=CUSTOM:103}", "display": "21522001:{363698007=27947004,364625007=410709000,406127006=255604002,CUSTOM:1=CUSTOM:103} (Finding site (attribute)=Epigastric region structure (body structure), Characteristic of pain (observable entity)=Cramping sensation quality (qualifier value), Pain intensity (observable entity)=Mild (qualifier value), Time since onset=a few weeks (1 weeks - 1 month))" }] }, "valueCodeableConcept": { "coding": [{ "system": "http://snomed.info/sct", "code": "52101004", "display": "Present" }], "text": "Symptom present" } } },	• Marked as "present" using the corresponding "52101004" SNOMED code (note that presenting complaints are always present) • Codec/coding/code contains the actual encoding of the symptom and its attributes in the same way as for all other symptoms
absent symptom	{ "fullUrl": "Complaint-Absent-0", "resource": { "resourceType": "Observation", "status": "final", "category": [{ "coding": [{ "system": "http://snomed.info/sct", "code": "409586006", "display": "Complaint" }] }], "code": { "coding": [{ "system": "http://snomed.info/sct", "code": "49727002", "display": "Cough" }] }, "valueCodeableConcept": { "coding": [{ "system": "http://snomed.info/sct", "code": "2667000",	• For symptoms by the patient explicitly reported as "not present" / "absent" • Marked as "Complaint" using a corresponding SNOMED code (409586006) • For „absent" we use „2667000". • Note that absent symptoms have fewer or no attributes i.e. only a "present" symptom can have an "intensity"

	"display": "Absent" }], "text": "Symptom absent" } } },	

unsure symptom	{
	"fullUrl": "Complaint-Unobtainable-0",
	"resource": {
	"resourceType": "Observation",
	"status": "final",
	"category": [
	{
	"coding": [
	{
	"system": "http://snomed.info/sct",
	"code": "409586006",
	"display": "Complaint"
	}
	]
	}
	],
	"code": {
	"coding": [
	{
	"system": "http://snomed.info/sct",
	"code": "29857009",
	"display": "Chest pain"
	}
	]
	},
	"valueCodeableConcept": {
	"coding": [
	{
	"system": "http://snomed.info/sct",
	"code": "876785008",
	"display": "Unobtainable"
	}
	],
	"text": "Symptom unobtainable"
	}
	}
	}
		• Similar to absent symptoms but with “876785008“ for „ Unobtainable“ instead of “absent”.
		• It is used whenever a user is not able or willing of providing an information, independent of the reason which is usually not asked by self-assessment applications

6.1.3.4 AI output data structure

While some AIs supported the optional FHIR endpoint, the AI output format for both API endpoints was the same as for MMVB 2.x and MMVB 1.x. The expected response was a JSON object with the fields described in Table 24. The MMVB 3.x AIs also implemented the health-check introduced for version 2.x. Table 25 shows the expected “healthy” response.

Table 24 – MMVB 3.x AI output structure for the FHIR and the non-FHIR benchmarking API endpoints

Field name	Example	Description

conditions	"conditions": [
	{
	"id": "ed9e333...bcde643",
	"name": "GERD"
	}
	]
		• The conditions the AI considers best explaining the presenting complaints.
		• Ordered by relevance descending

triage	"triage": "EC"	• The triage level the AI considers adequate for the given evidence

		• Uses the same abbreviations defined by the London-model EC, PC, SC, UNCERTAIN

Table 25 – Health-check endpoint API responses

Status	Example	Description
OK	{"data":"OK"}	• To signal that the AI API can accept benchmarking requests
ERROR	<ANYTHING ELSE>	• Anything else than the above OK message would be interpreted as error and might be used to stop sending further cases to the AI

6.1.3.5 Test data label/annotation structure

As raw input the MMVB 3.x uses case set format implemented for MMVB 2.2. For using it in the OCI benchmarking platform an additional script converted it into “devsplit” and a “testsplit” files required for the platform. Both files had the same format as the original case set. Table 26 shows the overall structure of a case set together with the labels to predict. Table 27 shows the relevant fields for encoding the MMVB 3.x case labels. For future versions we planed to migrate the case set format to FHIR documents too.

Table 26 – MMVB 3.x case set structure with label/annotation example


```
{
  "id": "f266e564-135f-4959-8a60-2f6d3f5cec9d",
  "name": "test10",
  "cases": [
    {
      "id": "29959f5c-c4e6-4341-9a1a-4802ce451629",
      "data": {
        "caseData": { ... },
        "metaData": {
          "name": "Synthesized case a5e425a2",
          "caseCreator": "MMVB Berlin model case synthesizer"
        },
        "valuesToPredict": {
          "correctCondition": {
            "id": "9d27455f69d907a7dad7bb471f3717a4",
            "name": "Appendicitis (disorder)",
            "standardOntologyUris": [
              "74400008"
```

```
            ]
          },
          "expectedCondition": {
            "id": "9d27455f69d907a7dad7bb471f3717a4",
            "name": "Appendicitis (disorder)",
            "standardOntologyUris": [
              "74400008"
            ]
          },
          "expectedTriageLevel": "EC",
          "impossibleConditions": [],
          "otherRelevantDifferentials": []
        }
      },
      "caseSets": [
        "f266e564-135f-4959-8a60-2f6d3f5cec9d"
      ]
    },
    ...
  ]
}
```


Table 27 – MMVB 3.x case with labels included

Field name	Example	Description

correctCondition	`"correctCondition": {`
	`"id": "2333...13c8",`
	`"name": "GERD",`
	`"standardOntologyUris": [ "http://snomed.info/id/235595009"`
	`]`
	`}`
		● The correct condition i.e. in MMVB 3.x

expectedCondition	Same as for correctCondition	● The conditions expected/accepted as top result for **explaining the presenting complaints** based on the given evidence. ● A list, but only one entry for mono-morbid cases as it is the case for MMVB 3.x
impossibleConditions	Same as for correctCondition	● An optional list of diseases that must not be contained in the results e.g.,because have contradict sex e.g.,male

		diseases for females in vice versa
otherRelevantDifferentials	Same as for correctCondition	• Other diseases that should be present in the result as relevant differentials to rule out.
expectedTriageLevel	"expectedTriageLevel": "PC"	• The expected triage level (EC, PC, SC, UNCERTAIN)
name	*"name"*: "BPPV test case #1",	• The name of the case. This is especially helpful for cases created by doctors.
caseSets	*"caseSets"*: ["4354...3a75"]	• The list of case sets containing this case. • Only used for case set management.

6.1.3.6 Scores and metrics

For the MMVB 3.x the focus was to learn how our current test metrics can be migrated to the OCI platform. For this the goal was to compare the results from our MMVB 2.x benchmarking platform with the new one to make sure the results are identical. We therefore did not update the metrics and continued to use the ones from the MMVB 2.x iterations. Table 28 shows the metrics as they have been configured in the OCI system.

Table 28 – The list of metrics migrated to the OCI system together with the descriptions used.

ID	Description	Range/Ordering
M1	Correct condition within the top 1 AI results.	• [0.0, 1.0] • More is better
M3	Correct condition within the top 3 AI results.	• [0.0, 1.0] • More is better
M10	Correct condition within the top 10 AI results.	• [0.0, 1.0] • More is better
Triage accuracy	Triage prediction correct.	• [0.0, 1.0] More is better
Triage similarity	Average similarity (1.0 = perfect match; 0.0 = maximum deviation (or uncertain))	• [0.0, 1.0] • More is better
Soft triage similarity	Average similarity (1.0 = perfect match; 0.0 = maximum deviation; counting 'uncertain' as 0.2 (slightly worse worst level similarity)).	• [0.0, 1.0] • More is better

6.1.3.7 Test data set acquisition

For the actual benchmarking we used the synthetic MMVB 2.2 data set sampled from the Berlin model for which the data set acquisition was described in 6.1.2.7.1. We also created a manually curated data set for testing the annotation tool, the annotation guidelines and the annotation process.

In several iterations the doctors created test sets facilitating the testing of the annotation tool and entered them the respective annotation tool version. It started with cases where only the symptoms could have been entered and then continued with cases where also attribute encoding in SNOMED was implemented. The different iterations involved different groups of doctors newly joining the topic group as this provided good opportunities to test the guidelines with doctors not previously involved. As part of the iterations the doctors collected feedback on the annotation tool, first in spreadsheets and then later directly inside the frontend using a corresponding feature implemented by the engineers of the topic group. The doctors' feedback was then discussed among the doctors in their own meeting stream and the also shared with the engineers. The relevant points have then been translated into Jira tickets and implemented by the engineers.

Following a decision by the doctors in TG-symptom workshop #3, the first iterations used the same 10 abdominal pain related diseases first used for the Berlin model in MMVB 2.x:

- Acute cholecystitis
- Acute Pyelonephritis
- Appendicitis
- Bladder Ca (first presentation)
- Ectopic Pregnancy
- Gastro-oesophageal reflux disease (GERD)
- Inflammatory Bowel Disease (first presentation, no flare)
- Irritable Bowel Syndrome (IBS)
- Simple urinary tract infection (Simple UTI)
- Viral Gastroenteritis (VG)
- (Abdominal pain idiopathic / no other symptoms (NOS))

As a 11th disease, also "idiopathic abdominal pain" was added. In contrast to MMVB 2.x, the restriction to a fixed set of 10 symptom was removed. The cases have been collected in a shared online spreadsheet[13]. This first set included 24 cases from 7 different doctors. Table 29 shows an example of a case included in this case set.

Table 29 – Case #1 of the cases prepared after workshop #3

Field	Example
Id	1
Company	…
Clinician	…
True Condition	Appendicitis
Triage Level	EC
Typicality	Typical
Biological sex	F

[13] https://docs.google.com/spreadsheets/d/14LR8XsH6gZdWAop3hkDOaABTHYQ_Vzz6PLahO6VP4HU/edit#gid=0

Age	21
Presenting Complaint(s)	Sharp right lower quadrant pain for since yesterday (12 hours or so), getting progressively worse.
Additional Evidence	Vomited once this morning, no blood. Currently feels nauseated. No diarrhoea or constipation. Felt feverish since last night. Temperature was 38.2C. No pv bleeding. Drinking fluids, urine output ok at present. Mouth feels dry. No dysuria. Feels generally unwell. Last menstrual period was 2 weeks ago. Denies recent unprotected sexual intercourse. Smoker. Nil other PMHx.

For the later iterations the doctors created a new set of 35 cases. As in the first case set, the cases have been fictional and covered the following 12 conditions:

- Crohn's disease
- Ulcerative colitis
- Gastro-oesophageal reflux disease
- Simple urinary tract infection
- Viral gastroenteritis
- Bladder cancer (first presentation)
- Acute cholecystitis
- Appendicitis
- Ectopic pregnancy
- Irritable bowel syndrome
- Acute pyelonephritis
- Abdominal pain not otherwise specified (idiopathic)

Crohn's disease and Ulcerative colitis have been added since their differentiation relies more on symptom attributes which was a focus for testing the annotation tool. Table 30 shows an example from this case set that was also shared as an online spreadsheet[14].

Table 30 – Case #33 of the MMVB 3.x case corpus

Field	Example
ID	33
Case	**Case ID:** 33 **Expected disease:** Acute pyelonephritis **Triage:** PC **Age:** 2 **Sex:** F **Presenting complaint:** 3 days of: fever anorexia dysuria urinary frequency **Present symptoms:** right-sided abdominal pain vomiting fatigue **Absent symptoms:** diarrhoea

For generating the actual benchmarking data the topic group's doctors then used the annotation tool to encode the free text using SNOMED. Figure 29 shows the same case as Table 30 entered by one of the topic group's doctors.

As part of the final iteration of encoding cases and testing them with the annotation tool this case was encoded by several doctors during the parallel case encoding test PCET. For this test three cases have been picked from the case set and independently encoded. The doctors then met several times to compare and analyse the nature of the difference in encoding, its potential impact on the AI responses and ways of minimizing the inter-annotator-noise both technically and by refining the guidelines.

6.1.3.8 Benchmarking system dataflow

As the benchmarking process description already implied, the MMVB 3.x dataflow through benchmarking architecture had the following relevant stages:

Model Generation

Same as for MMVB 2.x:

- The medical domain model was defined by the doctors direct in a google spreadsheet.
- From there it was exported as CSV file.

[14] https://docs.google.com/spreadsheets/d/1p1NbjkBOV3WLMBxg7Wx67qbRoPdq6N-UZK_K125B880

- The CSV was then pre-processed and converted into a JSON file by python script.
- This JSON model was then used by the MMVB 2.x benchmarking system as Domain Model for the case synthesizer.

Synthetic Case Generation

- The topic driver created the "Official FG AI4H Meeting I Benchmarking test data set" (71e9c086-2d8f-4cf9-bbe4-106117c95c5c) case set containing 1000 cases sampled from the domain model
- The generated case set was then stored to the Case Storage.

Case Set Export

- JSON export of the case set using the export function inside the MMVB 2.2 benchmarking platform.

Audit-Script integration

- Insertion of the case set into the TG-symptom audit script's /mmvb_datasets/ folder

Benchmarking data split creation and AI response computation

- Application of the mmvbAuditDataGenerator.py script to split the data into dev and test sets
- Export of AI input files into /ai-input-data folder
- Export of annotation files into /annotations
- Sending of ai inputs to all registered ai's and storage of the responses in the /submissions folder

Upload of the submissions to the audit platform

- Manual upload of the different AI submissions to the audit platform by the topic driver

Evaluation / metrics computation by the audit platform

- Computation of the metrics for all submissions

Result export

- Extraction of the results by copy paste from the leader-board screen of the audit platform.

6.1.3.9 Data sharing policies

For the benchmarking the MMVB 3.x iterations used the same synthetic case data as the MMVB 2.2 system which was freely accessible via the MMVB 2.2 benchmarking platform[15]. For the MMVB 3.x iterations the doctors working with the topic group also created a set of case vignettes in a shared online document, which was freely accessible too[16].

[15] http://35.228.161.168:3000/
[16] https://docs.google.com/spreadsheets/d/14LR8XsH6gZdWAop3hkDOaABTHYQ_Vzz6PLahO6VP4HU/edit#gid=0

6.1.3.10 Baseline acquisition

For MMVB 3.x no baseline for the human performance on the AI tasks was acquired.

6.1.3.11 Reporting methodology

As for MMVB 2.2, for MMVB 3.x no special reporting methodology was applied since it was still not the final MVB benchmarking. The results could only be accessed by the topic drive through the benchmarking platform and have been shared with the topic group directly and via presentations during the focus group meetings.

6.1.3.12 Result

Table 31 show the results from the evaluation by audit platform for several toy-AI versions on the "Official FG AI4H Meeting I Benchmarking test data set". It is important to note that the AIs used have been toy-AIs applied to data set that was sampled from a model that was knows to all participants. As expected from MMVB 2.2, on M1 the Ada Sampler performed best while the Random Sampler performed worst. We ran the evaluation both locally and in the audit platform to compare the results. Since the V1.1 of the Ada Sampler used in the audit platform was not available anymore, we included the local performance of V1.2 too for reporting local results for "Soft triage". While the local results for the V1.2 of the Ada Sample could not be compared with the results from the audit platform, for all other AIs the audit platform reported identical evaluation results in both scenarios. The only exception was the "Soft triage" metric which for an unknown reason could not be computed on the cloud hosted benchmarking platform. The comparison with the MMVB 2.2 system numbers was not possible as this systems did not support the separation into dev and test set. The numbers have been computed on the devsplit containing 100 of the 1000 cases in the case set used.

Table 31 also includes the results for the first test implementation of an FHIR/SNOMED endpoint. Among the non-random toy-AI it performed worst due to the incomplete attribute mapping. While this still shows that the general setup worked, this also underlines the further work is be needed.

It is also noteworthy that the Random Sampler performed worse than randomly picking a disease should perform, implying an issue that would need to be investigates to rule out an error in the current metric implementation.

Table 31 – Toy-AI benchmarking results for the "Official FG AI4H Meeting I Benchmarking test data set". Please note that the toy-AIs have nothing to do with the company's production AIs.

Toy AI Name	M1	M3	M10	Triage accuracy	Triage similarity	Soft triage distance audit platform	Soft triage distance audit local
Ada Berlin 1k/200k Sampling Toy AI V1.1	0.77	0.89	0.96	0.88	0.94	-	-
Infermedica Toy AI 2	0.74	0.89	0.97	0.85	0.93	-	0.925
Healthily Toy AI	0.74	0.87	0.92	0.74	0.86	-	0.864
Random Sampler	0.04	0.08	0.11	0.18	0.42	-	0.485

Ada Berlin 1k/200k Sampling Toy AI V1.2	0.79	0.91	0.98	0.89	0.945	-	0.945
Ada Berlin 1k/200k Sampling Toy AI V1.2 FHIR	0.56	0.76	0.85	0.85	0.925	-	0.925

6.1.3.13 Discussion of the benchmarking

Since MMVB was still not the final benchmarking with real AIs and real data the learnings and conclusions will be discussed in chapter 7.

6.1.3.14 Retirement

Since the MMVB 3.x iterations have not been real MVB benchmarking, similar to the MMVB 1 and 2 version no dedicated retirement protocol needs to be followed. The annotation tool frontend and backend are still online, including the databases. The retirement of the AIs that participated in this phase is up to the corresponding developers.

7 Overall discussion of the benchmarking so far

This section discusses the overall insights gained from benchmarking work in this topic group.

7.1 Overview of work done

The goal of the topic group on AI-based symptom assessment inside the ITU/WHO Focus Group on AI for Health was to specify a standardized benchmarking for AI-based symptom assessment systems. For working towards this goal, the topic group brought together 22 companies and 11 individual contributors from all over the world and explored the specific requirements of benchmarking such systems.

As part of this work several workshops have been held to identify the key topics to work on. Given that most topic group members represented companies already having their own benchmarking systems in place, the primary task was to agree on an approach that could benchmark all systems together in a standardized and unified way. The key points identified have been:

- Implementing a benchmarking platform that facilitates the
  - creation/annotation of benchmarking data
  - execution of the benchmarking
  - computation of the relevant metrics
  - reporting
- The specification of how to encode a benchmarking case that could be processed by all participating AIs
- The specification of a process for the creation of a benchmarking data set including
  - the corresponding guidelines for creating cases
  - a process for defining the conditions for which to create cases
- The creation of benchmarking data sets
- The creation and execution of a benchmarking challenge

Since the benchmarking platform was an essential part of the benchmarking, the work started with the design and implementation of a new benchmarking platform after the evaluation of existing

platforms showed limitations. The implementation work was set up in an iterative way leading to a working benchmarking platform that allowed benchmarking of AI systems against a data set. The main limitation at this stage was that the benchmarking platform was implemented using a simple medical domain model with only 10 conditions and 10 symptoms from the abdominal domain. The next phase therefore focused on lifting this limitation. The main challenge here was to find a way of encoding cases that could include all patient-reportable symptoms. The topic group decided to evaluate SNOMED as ontology for symptom encoding and implemented a dedicated annotation tool. The group also refined the interface to AI systems to support the SNOMED encoding. In parallel the doctors of the topic group created corresponding annotation guidelines.

At the time the focus group and the topic group ended in 2023 it was possible to encode cases using SNOMED/FHIR following the annotation guidelines and to run a basic benchmarking challenge in the OCI audit platform. This document reflects the key aspects of benchmarking symptom assessment systems. Work remaining for the future includes the integration of the symptom assessment annotation tool into the OCI annotation package, the definition and operational execution of a benchmarking data set collection call, the operational execution of a benchmarking and the subsequent publication of the results.

7.2 Learnings

The work on the benchmarking for AI-based symptom assessment systems lead to many relevant insights that are key to informing the way forward.

Symptom assessment AIs are often not based on ML

The work in the focus group showed that there is a difference between AI and machine learning as a specific sub field of it. Many challenges discussed in the focus group overfitted to machine learning and did not generalize to other AI paradigms even if this would have been possible. Working with the same materials, for instance in context of the audit group, was therefore slower than necessary. While giving guidance on today's ML challenges is important, it would be useful to generalize the work in the focus group and future global initiatives in a way that does also apply to new generations of AI.

AI product manufacturers have business constraints

Symptom assessment AIs are already marketed by companies as products. In contrast to the majority of topic groups where the AI systems developed are still part of research in an academic environment, symptom assessment benchmarking has to take business constraints into account. Some of them require different technical benchmarking solutions that have not fully been integrated into and supported by the OCI platform. The most important constraints included:

- Due to IP limitations and due to the complexity of real world medical device production systems, AIs cannot be submitted as containers for a benchmarking that could be evaluated in a sandbox. AIs need to be benchmarked while running in their own production environment. This however implies that benchmarking cases need to be considered public in the moment the first AI is tested with it. This in turn makes it more complex to protect the benchmarking against fraud and also requires the ongoing creation of new benchmarking data sets.
- The interest of companies appearing in benchmarking results and leader-boards where they are not at the top is limited.

Conflict of interest

Another implication that applies more to this topic group than to more academically driven ones is that significant parts of the knowledge about benchmarking symptom assessment systems lies within the companies and was not widely discussed in the academic community. While having this unique knowledge and experience to make the benchmarking viable, the companies are also not neutral and must not be involved in the organization and operational execution of the benchmarking. For the operation of the real benchmarking after the preparation and specification by the topic group driven by the companies, another organizational structure is needed, which the current focus group or topic group could not provide.

Symptom assessment AIs are medical devices

With the introduction of the medical device regulation in the EU, symptom assessment systems have to be classified as medical devices. This implies that the products must be compliant with this regulation in order to be marketed within the EU. The MDR already implies a process of immense complexity. A company already having a symptom assessment system as medical device on the market reduces need but also the degree of freedom to adopt any regulatory guidance from the focus group.

Semantic case encoding is complex and leads to mapping friction

Encoding self-reported patient cases in an unambiguous and complete way is challenging. First, creating and maintaining an ontology for the topic group or focus group that is closer to the self-reporting symptom ontologies of the companies is not feasible with the resources at hand. Second, even the best existing ontologies for symptoms are not as clean, precise and complete in terms of self-reportable findings and their attributes as the company ontologies. Mapping from real cases to any ontology and then from this ontology the individual company ontologies will always produce some “mapping friction”, i.e. AIs having lower performance on the test data than in the real world making the interpretation and comparison of the results mode difficult. Independent of mapping friction, the mapping from a joint ontology to the company ontologies for the 1,000 or so symptoms needed to encode cases is cost intensive.

Interactive assessment dialog benchmarking

The focus of the work was on benchmarking with case vignettes which provided the AIs with all symptoms at once. Most applications however ask a series of dynamic questions once a user has shared the problem they seek advice for. Collecting the right evidence for making decisions is an important capability of intelligent symptom assessment systems. The benchmarking should therefore include this dialog in the benchmarking.

GPT/LLM impact on symptom assessment and its benchmarking

In the first topic group workshop we anticipate that mid-tern all symptom assessment systems will support free-text conversations. The widespread uptake of GPTs early 2023, however came slightly faster than expected. It will a have a significant impact on symptom assessment systems and their benchmarking:

- The interaction of GPTs with symptom assessment systems will quickly allow conversational interfaces across the board.

- While it will take time to build medical devices with the sufficient medical safety, the near future will bring up many GPT/LLM based symptom assessment applications that need to be supported by the benchmarking.
- Using GPTs, the benchmarking can be radically simplified by switching to free-text vignettes and thereby skipping the entire ontology mapping – the hardest problem in symptom assessment benchmarking.
- GPTs will also enable testing assessment dialogs – a still missing feature of the current benchmarking.

7.3 Next steps

As this document outlined, the topic group made significant progress towards specifying many relevant details to consider when benchmarking symptom assessment systems. However, for the ultimate goal of establishing the continuous standardized benchmarking of symptom assessment system executed on a regular basis more work needs to be done. Work that, as discussed above, the focus group with its limited life span as a "speed boat" exploration structure cannot provide a sustainable home for. For that reason, with the focus group now coming to an end we aim to continue the journey as part of the new WHO AI4H Global Initiative that with its currently anticipated structure can provide the framework that is needed. To establish and run the standardized benchmarking inside the GI a few important topics need to be worked on:

Text vignette friendly case annotation tool

As outlined above, 2023 marks the start of a new era with generative AI and in particular GPTs arriving in every aspect of life. For this new world we need to adjust the benchmarking to evaluate symptom assessment based on this technology and at the same time simplify the symptom assessment benchmarking as far as possible to increase acceptance and impact. For this we will rewrite the annotation tool and the underlying data structures and APIs to enable free-text vignettes and remove the entire ontology mapping approach.

Annotation package integration

While the annotation tool allows the description of a case for the benchmarking, there also needs to be all the infrastructure around it where annotation rounds can be scheduled, annotators can login, see their assigned tasks and open tasks to work on them. It also needs to take care of the peer reviewing process and arbitration in case of annotation conflicts. Since it deals with sensitive data that must not leak before the benchmarking it has to have a high degree of safety, security, reliability and quality. To avoid every topic group needing to invest time in building a suitable platform, the OCI created the annotation package that serves this purpose. The new annotation tool needs to be integrated with the annotation package so that it is possible to open symptom assessment case creation tasks in this annotation tool.

Annotation guideline update

Once the new annotation tool is ready and integrated into the annotation package the annotation tool handbook and the annotation guidelines need to be updated accordingly. With the transition to structured text vignettes the annotation task gets significantly simpler but still the annotation guidelines are the only communication between the topic group and the annotators than can influence the case quality in a positive way.

AI interface update

So far the AI interface has only been used by topic group members. For a general call for participation in a symptom assessment benchmarking there is a need for a dedicated AI API interface specification and a corresponding reference implementation that can serve as a template for participants. Part of this work would also be the creation of a data set to allow testing of the AI interface.

Annotation round execution
One of the two important steps potentially more realistic in the GI is the administrative side of setting up an annotation round – ranging from determining together with the topic group a set of conditions and generating corresponding case creation/annotation/peer-reviewing/base-line establishment tasks in the annotation package, to then inviting doctors to work in them. There should be no connection between the doctors and any of the participants in the benchmarking so that the results are not compromised.

Benchmarking challenge setup and execution
Once the data is collected the GI could setup the corresponding benchmarking challenge in the OCI benchmarking platform based on the materials provided by the topic group. It would then also finally run the benchmarking and record the official results. The actual operational execution of the benchmarking is again a task that must not be performed by the topic group so that the results can be trusted.

Result publication
After benchmarking was completed, the results should be published. As discussed above, business constraints might require that participants can stay anonymous in the leader-board. The specification of how the results can be published is the final open task that requires further work. This again would need to happen in close cooperation with the GI since the topic group members must not know the identity of benchmarking participants that prefer to stay anonymous.

In general, we see the GI as an important step towards implementing the benchmarking for AI based symptom assessment systems which in turn will simplify and foster their application in the field.

7.4 A comment on the LLMs in context of symptom assessment

The following sections comment on the impact that the rise of LLM models end of 2022 / beginning of 2023 could have on symptom assessment and its benchmarking.

7.4.1 The possible place of LLMs in symptom assessment systems

Artificial intelligence (AI) as a concept has been around since the 1940s. Novel ideas and improvements in the computer's ability to simulate human intelligence have resulted in this artificial intelligence technology moving forward incrementally. Then in 2017, scientists at Google published a paper on a truly sea change concept in machine learning called transformer and the world changed [42]. This section provides a high-level overview and should act as a framework from which one can interpret ongoing developments in this space in general and with respect to symptom assessment and its benchmarking in particular.

Recently the development of *generative* AI, not yet *general* AI and large language models (LLMs) has been the top of mind topic of scientific forums, social media, legitimate news sources for not only professionals, but for lay audiences as well. Generative AI is the inflection point that made

arcane AI real to the public due to its conversational interface and ability to create text and images based on data scrubbed from universe of sources available in written and digital forms. Quite suddenly, data became information, usable and actionable in seemingly myriad applications in professional and daily life. ChatGPT (GPT3) from OpenAI was available at the end of 2022 and became the most widely and rapidly deployed technology in human history. It used a model with about 175 billion parameters and in widely published demonstrations produced astonishing output that captivated the attention of the world. In March 2023, OpenAI released GPT4, without announcing further details on how and on what it was trained, but estimates are that it may have an order of magnitude more data in its training data. Additionally, it has multimodal capabilities beyond just text as in ChatGPT. It "knows" many languages and can code from text instructions/prompts and translate between words and multiple code languages seamlessly.

Much has been made, justifiably, of GPT4's ability to pass professional licensure examinations in medicine and law without specialised training. Indeed, its savant like performance in multiple arenas is breathtaking. And GPT4 is not the only generative AI/LLM with these capabilities, although it is certainly the most notorious. More generative AI/LLM models are announced in news cycles and any enumeration becomes obsolete upon publication. The best known of these other models include Bard, Gorilla, LLaMa, PaLM, PaLM2, Falcon as well as other variations. Moreover, other countries active in AI are developing their own LLMs with unknown capabilities.

With all of this notoriety and publicity, concerns in regard to limitations and faults have become known and controversial. It is likely that these issues will be significantly overcome, though not ever truly eliminated. We need to keep the perspective that errors, even if asymptotically reduced to minor issues, may be acceptable within the limits of a proper risk management approach.

LLMs give the appearance of understanding what they are writing, but the word choices are actually mathematically derived using vector representations of words based on training data, while paying attention to context. This "fill in the blank" model depends on the sophistication of the received prompts…the LLM equivalent to being asked the right question.

With the earlier ChatGPT/GPT3 version, the information base was limited to the period in which it was trained, thus it could respond factually up to the end of 2021. With Microsoft's investment and collaboration with OpenAI, GPT4 has been integrated into a modern search engine and thus is able to access contemporary information. Apparently, many of the other LLMs have or will have that ability as well. Other features available in GPT4 and Bard and others to come are the ability to engage in modalities other than just text. When we think of the images, video streams, audio and other digital data streams that exist in healthcare, the implications of multimodal competence for medical scientists, clinicians and patients are mind boggling. AI has established the ability to predict protein folding based on amino acid sequencing in various physiologic environments. The implications for biopharmaceutical development of new investigational products when AI can accurately characterise and potentially enhance the receptors, active site of drugs and parameters that impact bioavailability will be wondrous to behold.

In context of this document it is also relevant to consider the impact of LLMs on the assessment of patient symptoms. Adoption and adaptation of these generative AI/LLM technologies will pretty soon be integrated into products and services to varying degrees. Some of these may be real and impactful, others less so, but nonetheless purporting to deploy these capabilities will be rampant to avoid appearing behind the competition. It is clear that many companies will be moving in this

direction in order to stay relevant. An important part of this work especially in the medical domain however has to be centred around medical safety – an aspect that in the current excitement about LLMs tends to be ignored.

Before the advent of generative AI and large language models, assessment of patient symptoms was dependent on defining and clarifying a specific symptom, identifying clinical synonyms and then linking them to possible diagnoses. This clearly was a labour intensive effort fraught by oversimplification when viewed through a clinician's perspective, especially given the co-morbidities, demographic and cultural variations encountered in what we clinicians arrogantly refer to as "real life". The process of validation of even these stick-drawing models of symptoms has been problematic and subject to enormous bias.

LLMs provide the opportunity to leapfrog these obstacles by surveying the essential totality of human written knowledge. By generating text of probabilities, LLMs amazingly predict what any given collection of symptoms may mean. It uses a form of intelligence that is not human based. Classically, an experienced clinician integrates what the clinician knows about diseases and human pathophysiology, what the clinician has learned from the fraction of available research material he/she has been able to digest and the sum of the clinician's experiences to come up with what any given set of symptoms may mean in any given context. It seems that the LLM may come to similar conclusions without any true understanding of the presenting symptoms and underlying diseases, but rather a complex and mostly unknown mathematical correlation of word vectors, presented in a conversational exchange. Similar result, different path to get there, although starting from the same place, i.e., a set of presenting symptoms in a given clinical context.

Generative AI/LLMs seem to have another advantage when it comes to assessment of symptoms. LLMs can pay attention to context such as patient demographics, concomitant conditions, past history and active or past medications that may have confounding adverse effects and other variables that may impact symptom presentation. These all become part of the sophisticated "prompt" used to interrogate the LLM and thus can be factored into the output. The ability to handle complex symptom inputs and outputs, common in real medicine, make LLMs more effective than traditional computerised models and more closely resemble what clinicians face daily. LLMs may minimise the time lost in unnecessary and costly diagnostic testing, by directing and focusing clinicians on those strategies that can confirm or deny possible diagnoses or suggest additional considerations.

A critique raised about LLMs has been that technology will never have the empathy that humans seek when distressed. It can be argued that the human touch would always secure the role of the human clinician. Yet GPT4 has shown empathy previously thought not possible [43]. The simulation of empathy will further lower the threshold of patient adoption. Although a human touch may still be required in healthcare, empathy may not be the domain of humans alone.

And, at the end of all this, the human clinician bears the ultimate responsibility for the care delivered. The medically enhanced LLM can prompt the clinician to delve into further investigations by suggesting other diagnostic possibilities based on patient symptoms, but that last mile/last kilometre care is the purview of the attending clinician. If optimal patient care is indeed our common goal, then we should welcome this technology. This does leave problems in the area of consumer-facing symptom checkers, where liability may rest with the producer of the application in

the absence of supervision by a clinician, which is likely to mean that LLM-based symptom checkers appear in clinical settings long before they are considered ready and safe for general use.

It certainly will help level the clinical playing field. If the patient outcome is based on the knowledge and experience of the clinician, then tools that give that brand new clinician the knowledge and wisdom of the most senior clinician should be mandatory. AI then becomes *augmented* intelligence that whispers into the ear of every clinician to make sure that state of the art knowledge is presented in a human voice that is easily understood and can be challenged because the interface is conversational. It also means that this degree of sophisticated information can be tailored to the available resources and that knowledge is not cloistered in some specialty facility not accessible to a generalist. The clinician can then take the relevant action required to best serve the patient in front of the clinician. Due to the conversational nature of LLMs, students and less experienced clinicians can practice their diagnostic skills from presenting symptoms and the LLMs can adjust the degree of difficulty commensurate with the needs of the student. At the end, clinical care and health outcome will be improved at a hopefully reduced cost. Value based care will be closer to reality.

It is important to notice that the uptake of LLMs early 2023 was only the beginning. Advances of varying significance will be reported almost daily. Legal and moral codes will just need to follow, as they always do when technological advances outstrip our ethical envelope. Legal, regulatory and ethical overreach may result if egregious events attributable to AI are documented, but will settle down as adjustments are made and common sense prevails.

7.4.2 Implications for benchmarking

At present, no symptom checkers driven by LLM-based reasoning are available, but it can be expected that during 2023 several solutions will appear. At the same it is clear that these first prototypes will not easily gain medical device certification. Any future use of Large Language Models as the basis of symptom checkers leaves some aspects of benchmarking untouched, but will modify others to varying degrees.

The creation of test cases for vignettes and establishing ground truth, will be just as important with LLMs as with the current generation of symptom checkers. The annotation of the vignettes is likely to change, however, with the Natural Language Processing capabilities of LLMs rendering obsolete the mapping of signs and symptoms to a standard ontology. LLMs may also provide an alternative solution to constructing interfaces between a benchmarking platform and the symptom checkers under test, with LLMs being theoretically capable of taking the output of a symptom checker (for example triage advice or follow up questions to be asked of the user) and using the vignette to determine whether the output matches the ground truth or how questions should be answered. More work will be needed to determine the accuracy of this approach.

8 Regulatory considerations

For AI-based technologies in healthcare, regulation is not only crucial to ensure the safety of patients and users, but also to accomplish market acceptance of these devices. This is challenging because there is a lack of universally accepted regulatory policies and guidelines for AI-based medical devices. To ensure that the benchmarking procedures and validation principles of FG-AI4H

are secure and relevant for regulators and other stakeholders, the working group on *"Regulatory considerations on AI for health" (WG-RC)* compiled requirements that consider these challenges.

The deliverables with relevance for regulatory considerations are DEL2 *"AI4H regulatory considerations"* (which provides an educational overview of some key regulatory considerations), DEL2.1 *"Mapping of IMDRF essential principles to AI for health software" and* DEL2.2 *"Guidelines for AI based medical device (AI-MD): Regulatory requirements"* (which provides a checklist to understand expectations of regulators, promotes step-by-step implementation of safety and effectiveness of AI-based medical devices and compensates for the lack of a harmonized standard). DEL04 identifies standards and best practices that are relevant for the "*AI software lifecycle specification."* The following sections discuss how the different regulatory aspects relate to the TG-Symptom.

8.1 Existing applicable regulatory frameworks

Most of the AI systems that are part of the FG-AI4H benchmarking process can be classified as *software as medical device* (SaMD) and eligible for a multitude of regulatory frameworks that are already in place. In addition, these AI systems often process sensitive personal health information that is controlled by another set of regulatory frameworks. Symptom assessment AI systems are here no exception.

They take general health profile information (like age, sex, risk factors), presenting complaints the user seeks advice for and answers to dynamic AI follow-up questions as input and generate outputs such as a pre-clinical triage, advice on general next steps, possible underlying conditions often combined with probability estimates, condition specific advice and information and further diagnostic steps as output. The intended users are for most systems "normal users" (”layperson", ”patients", people without special medical professional background or knowledge). They are used primarily in a home setting on smart phones or in the web browser. The "State of healthcare situation or condition" is primarily non-serious, sometimes serious. The "Significance of information provided by AI system to healthcare decision" falls mostly in the "Treat or diagnose" category. Depending on the specific intended use of each system according to the IMDRF AI system risk classification scheme (Table 32) symptom assessment systems could therefore be classified as class II-III.

Table 32 – IMDRF AI system risk classification scheme

State of healthcare situation or condition	**Significance of information provided by AI system to healthcare decision**		
	Treat or diagnose	**Drive clinical management**	**Inform clinical management**
Critical	IV	III	II
Serious	III	II	I
Non-serious	II	I	I

In most cases symptom assessment systems have to be considered to be “software as a medical device” and have to implement the regulatory frameworks of organizations overseeing the markets in which the AI device manufacturer plans to market their products.

8.2 Regulatory features to be reported by benchmarking participants

In most countries, benchmarked AI solutions can only be used legally if they comply with the respective regulatory frameworks there. To enable screening of benchmarked solutions to identify the ones that can be legally applied in a given context the submitted AI should provide dedicated metadata on regulatory compliance.

The concrete regulatory metadata that have to be provide for a benchmarking will be updated before every benchmarking and tailored to its specific needs. The list should be published in advance as an RFC document so that potential participants and stakeholders can comment and make sure that all regulatory dimensions relevant for them will be considered. As part of the publication of a benchmarking call the concrete intended use to be benchmarked should also be published to make sure that AIs only participate where their developed/registered intended use is compatible. Table 33 lists criteria that during the time of the topic group have been discussed as candidates for audit metadata fields to be provided on AI submission.

Table 33 – Candidates for AI regulatory metadata fields

Metadata field	Type	Description / Comment
General company certification		
Company GDPR compliance	yes/no	The General Data Protection Regulation mandatory for all EU based AI manufacturers
Company HIPPA compliance	yes/no	Health Insurance Portability and Accountability Act
Company ISO 27001 compliance	yes/no	Information security management
Medical device		
Intended use	string	Description of the intended use/purpose/task the AI will implement
Intended users	string	Description of who will use the AI (level of expertise required/assumed)
Intended application context	string	Description of the context where the AI is supposed to be used by its intended user
Certified MDD class	[I, I*, IIa, IIb, III]	Medical Device Directive (MDR predecessor)
Certified MDR class	[I, IIa, IIb, III]	Medical Device Regulation
IMDRF Risk class	[I, II, III, IV]	See Table 32
MDSAP	yes/no	Medical Device Single Audit Program by Australia, Brazil, Canada, Japan and USA – to reduce certification effort (not fully established yet)
ISO 13485:2016	yes/no	Quality management system
21 CFR part 820	yes/no	FDA Quality System Regulation QSR; is likely to be migrate to ISO 13485
IEC 62304	[A, B, C]	Software Safety Classes
Other AI / Application certifications		
ADA compliance	yes/no	American Disability Act compliance is required for symptom assessment applications marketed in the USA

8.3 Regulatory requirements for the benchmarking systems

While the benchmarking system for symptom assessment AIs is not itself a medical device, it should enable the collection of evidence that is trustworthy enough to be used as part of regulatory processes. It is therefore recommended to follow similar guidelines for the implementation of the benchmarking system.

With the migration from the TG-symptom's own benchmarking platform to the platform implemented by the Open Code Initiative of the focus group, most of the regulatory relevant code resides on the OCI side where the OCI is working towards implementing the necessary quality standards.

For the parts of the software specific for symptom assessment benchmarking like the annotation tool, the AI interfaces, the metrics implementation and the custom parts of the audit script, the same quality standards should be used. With the transition to the GI a unified QMS system should be introduced. With the setup of such a shared QMS and corresponding SOPs, the TG-symptom components might need to be reimplemented following the process.

The minimal measures that need to be taken for the TG-symptom specific parts of the benchmarking systems will likely include among other points:

- **Processes & Procedures:** Defining and implementing processes and procedures (SOPs) for all aspects of changes made to the software or the benchmarking.
- **Document control and change management:** making sure that all artifacts used in the process are stored in a way where all changes are transparent and can be traced and audited.
- **Planning & Requirements:** Introduction of a structured planning process from requirements down to tickets that can be implemented.
- **Technical Documentation:** Creating and maintaining formal technical documentation for all components, data structures, APIs and architecture. Chapter 6 contains first steps in this direction.
- **Risk Management:** Applying risk management for all components with risk/threat modelling.
- **Testing & QA:** A systematic quality testing approach e.g., complete test coverage, code review, code quality metrics and user testing e.g., letting doctors annotate cases during testing as was already done by the topic group.
- **Monitoring & Incident handling:** Monitoring of the benchmarking system while data is collected or a challenge is running – connected to a corresponding process for handling incidents.

These points are only a high level selection and in larger companies every point is handled by entire teams and is cost intensive. A key task inside the GI would therefore be to find ways of implementing them for all topic groups in as lean and streamlined a way as possible.

8.4 Regulatory approach for the topic group

As outlined above, symptom assessment systems have to be considered medical devices. Implementing symptom assessment AIs therefore requires in most cases the full scale compliance with a medical device regulation framework like for instance the Medical Device Regulation in the EU which implies that they have to take care of virtually every conceivable regulatory aspect also outlined in DEL2 *"AI4H regulatory considerations."* Beside asking for the concrete frameworks implemented, no further guidance is needed by the topic group.

For the TG-symptom specific parts of the benchmarking platform we will follow and implement the guidance and standards that we expect to be set by the GI for all topic groups covering all relevant points at a level the GI and its stakeholder consider adequate.

9 References

[1] *Tracking Universal Health Coverage: 2017 Global Monitoring Report.* World Health Organization and International Bank for Reconstruction and Development / The World Bank. 2017. http://pubdocs.worldbank.org/en/193371513169798347/2017-global-monitoring-report.pdf.

[2] *Global health workforce shortage to reach 12.9 million in coming decades.* from WHO: https://www.who.int/mediacentre/news/releases/2013/health-workforce-shortage/en/

[3] Global Strategy on Human Resources for Health: Workforce 2030: Reporting at Seventy-fifth World Health Assembly https://www.who.int/news/item/02-06-2022-global-strategy-on-human-resources-for-health--workforce-2030

[4] PushDoctor. UK Digital Health Report 2015. https://www.pushdoctor.co.uk/digital-health-report

[5] Pillay N. The economic burden of minor ailments on the national health service in the UK. SelfCare 2010; 1:105-116

[6] United Nations. Goal 3: Ensure healthy lives and promote well-being for all ages. https://www.un.org/sustainabledevelopment/health/

[7] Sebastian Köhler, Michael Gargano, Nicolas Matentzoglu, Leigh C Carmody, David Lewis-Smith, Nicole A Vasilevsky, Daniel Danis, Ganna Balagura, Gareth Baynam, Amy M Brower, Tiffany J Callahan, Christopher G Chute, Johanna L Est, Peter D Galer, Shiva Ganesan, Matthias Griese, Matthias Haimel, Julia Pazmandi, Marc Hanauer, Nomi L Harris, Michael J Hartnett, Maximilian Hastreiter, Fabian Hauck, Yongqun He, Tim Jeske, Hugh Kearney, Gerhard Kindle, Christoph Klein, Katrin Knoflach, Roland Krause, David Lagorce, Julie A McMurry, Jillian A Miller, Monica C Munoz-Torres, Rebecca L Peters, Christina K Rapp, Ana M Rath, Shahmir A Rind, Avi Z Rosenberg, Michael M Segal, Markus G Seidel, Damian Smedley, Tomer Talmy, Yarlalu Thomas, Samuel A Wiafe, Julie Xian, Zafer Yüksel, Ingo Helbig, Christopher J Mungall, Melissa A Haendel, Peter N Robinson, The Human Phenotype Ontology in 2021, Nucleic Acids Research, Volume 49, Issue D1, 8 January 2021, Pages D1207–D1217, https://doi.org/10.1093/nar/gkaa1043, https://hpo.jax.org/app/

[8] *Ethics and governance of artificial intelligence for health* https://www.who.int/publications/i/item/9789240029200

[9] Hamberg K. Women's Health. Gender Bias in Medicine. 2008;4(3):237-243. doi:10.2217/17455057.4.3.237

[10] J. Maude, Accuracy of a Machine Learning Based Ddx Generator, DEM, 10th International Conference, Boston, MA, Oct 8–10, 2017 https://www.isabelhealthcare.com/pdf/DEM_2017_Isabel_Accuracy.pdf

[11] Paul Manicone MD, Claire Stewart MD, Jeremy Kern MD, Mary Ottolini, MD MPH Children's National Medical Center, Washington, DC, „'ASKING ISABEL' FOR DIAGNOSTIC DILEMMAS IN PEDIATRICS: HOW DOES A WEB BASED DIAGNOSTIC CHECKLIST PERFORM?", PAS Poster 2013, https://www.isabelhealthcare.com/pdf/ISABEL_PAS_POSTER_2013.pdf

[12] Mark L. Graber, MD, Ashlei Mathew, Performance of a Web-Based Clinical Diagnosis Support System for Internists, J Gen Intern Med. 2008 Jan; 23(Suppl 1): 37–40., https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2150633/

[13] Semigran Hannah L, Linder Jeffrey A, Gidengil Courtney, Mehrotra Ateev. Evaluation of symptom checkers for self diagnosis and triage: audit study BMJ 2015; 351 :h3480 https://www.bmj.com/content/351/bmj.h3480

[14] Semigran, H. L., Levine, D. M., Nundy, S., & Mehrotra, A. (2016). Comparison of physician and computer diagnostic accuracy. JAMA Internal Medicine, 176(12), 1860-1861.

[15] Davies, B. M., Munro, C. F., & Kotter, M. R. (2019). A novel insight into the challenges of diagnosing degenerative cervical myelopathy using web-based symptom checkers. Journal of Medical Internet Research, 21(1), e10868.

[16] P Ramnarayan, A Tomlinson, A Rao, M Coren, A Winrow, J Brit. ISABEL:a web-based differential diagnostic aid for paediatrics: results from an initial performance evaluation ADC BMJ 2002 https://adc.bmj.com/content/88/5/408.long

[17] Hamish Fraser, Enrico Coiera, David Wong. Safety of patient-facing digital symptom checkers. Lancet Vol 392 Issu 10161 Correspondence https://www.thelancet.com/journals/lancet/article/PIIS0140-6736(18)32819-8/fulltext

[18] Moreno BE, Pueyo FI, Sánchez SM, Martín BM, Masip UJ. A new artificial intelligence tool for assessing symptoms in patients seeking emergency department care: the Mediktor application. Emergencias 2017; 29:391-396. https://www.ncbi.nlm.nih.gov/pubmed/29188913

[19] Nazario Arancibia, J. C., Martín Sanchez, F. J., Del rey Mejías, A. L., del Castillo, J. G., Chafer Vilaplana, J., Briñon, G., ... & Seara Aguilar, G. (2019). Evaluation of a diagnostic decision support system for the triage of patients in a hospital emergency department. International Journal of Interactive Multimedia & Artificial Intelligence, 5(4).

[20] Liang H, Tsui BY, Ni H et al. Evaluation and accurate diagnoses of pediatric diseases using artificial intelligence. Nat Med. 2019; 25(3):433-438. doi: 10.1038/s41591-018-0335-9. https://www.nature.com/articles/s41591-018-0335-9

[21] Ronicke S, Hirsch MC, Türk E, Larionov K, Tientcheu D, Wagner AD. Can a decision support system accelerate rare disease diagnosis? Evaluating the potential impact of Ada DX in a retrospective study. Orphanet Journal of Rare Diseases 2019; 14:69.

[22] Bisson LJ, Komm JT, Bernas GA, et al. Accuracy of a computer-based diagnostic program for ambulatory patients with knee pain. Am J Sports Med. 2014;42:2371–2376.

[23] Bisson, L. J., Komm, J. T., Bernas, G. A., Fineberg, M. S., Marzo, J. M., Rauh, M. A., ... & Wind, W. M. (2016). How accurate are patients at diagnosing the cause of their knee pain with the help of a web-based symptom checker?. Orthopaedic Journal of Sports Medicine, 4(2), 2325967116630286.

[24] Powley, L., McIlroy, G., Simons, G., & Raza, K. (2016). Are online symptoms checkers useful for patients with inflammatory arthritis?. BMC musculoskeletal disorders, 17(1), 362.

[25] Salman Razzaki, Adam Baker, Yura Perov, Katherine Middleton, Janie Baxter, Daniel Mullarkey, Davinder Sangar, Michael Taliercio, Mobasher Butt, Azeem Majeed, Arnold DoRosario, Megan Mahoney, Saurabh Johri: A comparative study of artificial intelligence and human doctors for the purpose of triage and diagnosis https://arxiv.org/abs/1806.10698.

[26] Liu, X., Cruz Rivera, S., Moher, D. *et al.* Reporting guidelines for clinical trial reports for interventions involving artificial intelligence: the CONSORT-AI extension. *Nat Med* **26,** 1364–1374 (2020). https://doi.org/10.1038/s41591-020-1034-x

[27] Poote AE, French DP, Dale J, Powell J. A study of automated self-assessment in a primary care student health centre setting. J Telemed Telecare 2014 Apr; 20(3):123-127 https://pubmed.ncbi.nlm.nih.gov/24643948/

[28] Verzantvoort NC, Teunis T, Verheij TJ, van der Velden AW. Self-triage for acute primary care via a smartphone application: practical, safe and efficient? PLoS One 2018 Jun 26;13(6) https://dx.plos.org/10.1371/journal.pone.0199284

[29] Berry AC, Cash BD, Wang B, Mulekar MS, Van Haneghan AB, Yuquimpo K, et al. Online symptom checker diagnostic and triage accuracy for HIV and hepatitis C. Epidemiology and Infection 2019 Jan; 147:e104 https://europepmc.org/article/MED/30869052

[30] Gilbert S, Mehl A, Baluch A, Cawley C, Challiner J, Fraser H, et al. How accurate are digital symptom assessment apps for suggesting conditions and urgency advice? A clinical vignettes comparison to GPs. BMJ Open 2020 Dec 16;10(12):e040269 https://bmjopen.bmj.com/content/10/12/e040269.long

[31] Hill MG, Sim M, Mills B. The quality of diagnosis and triage advice provided by free online symptom checkers and apps in Australia. Med J Aust 2020 Jun; 212(11):514-519 https://pubmed.ncbi.nlm.nih.gov/32391611/

[32] Yu SW, Ma A, Tsang VH, Chung LS, Leung SC, Leung LP. Triage accuracy of online symptom checkers for accident and emergency department patients. Hong Kong J Emergency Med 2020 Jul; 27(4):217-222 https://journals.sagepub.com/doi/10.1177/1024907919842486

[33] Ceney A, Tolond S, Glowinski A, Marks B, Swift S, Palser T. Accuracy of online symptom checkers and the potential impact on service utilisation. PLoS One 2021 Jul 15; 16(7):e0254088 https://pubmed.ncbi.nlm.nih.gov/34265845/

[34] Chan F, Lai S, Pieterman M, Richardson L, Singh A, Peters J, et al. Performance of a new symptom checker in patient triage: Canadian cohort study. PLoS One 2021 Dec 01; 16(12):e0260696 https://pubmed.ncbi.nlm.nih.gov/34852016/

[35] Delshad S, Dontaraju VS, Chengat V. Artificial intelligence-based application provides accurate medical triage advice when compared to consensus decisions of healthcare providers. Cureus 2021 Aug 06; 13(8):e16956 https://pubmed.ncbi.nlm.nih.gov/34405077/

[36] Gilbert S, Fenech M, Upadhyay S, Wicks P, Novorol C. Quality of condition suggestions and urgency advice provided by the Ada symptom assessment app evaluated with vignettes optimised for Australia. Aust J Prim Health 2021 Oct; 27(5):377-381 https://pubmed.ncbi.nlm.nih.gov/34706813/

[37] Dickson SJ, Dewar C, Richardson A, Hunter A, Searle S, Hodgson LE. Agreement and validity of electronic patient self-triage (eTriage) with nurse triage in two UK emergency departments: a retrospective study. Eur J Emerg Med 2022 Feb 01; 29(1):49-55 https://pubmed.ncbi.nlm.nih.gov/34545027/

[38] Summerton N. The medical history as a diagnostic technology. Br J Gen Pract. 2008; 58(549)

[39] Bertens LC, Broekhuizen BD, Naaktgeboren CA, et al. Use of expert panels to define the reference standard in diagnostic research: a systematic review of published methods and reporting. PLoS Med. 2013;10(10):e1001531.

[40] Hohmann E, Brand JC, Rossi MJ, Lubowitz JH. Expert Opinion Is Necessary: Delphi Panel Methodology Facilitates a Scientific Approach to Consensus. Arthroscopy 2018; 34(2):349-351 https://www.sciencedirect.com/science/article/pii/S0749806317314421

[41] Sendak MP, Gao M, Brajer N, Balu S. Presenting machine learning model information to clinical end users with model facts labels. NPJ Digit Med. 2020;3(1):41. https://doi.org/10.1038/s41746-020-0253-3.

[42] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aiden Gomez, Lukasz Kaiser, Illia Polosukhin. Attention is All You Need. Advances in Neural Information Processing Systems 30, 2017**.** arXiv:1706.03762

[43] Peter Lee, Carey Goldberg, Isaac Kohane. The AI Revolution in Medicine: GPT-4 and Beyond. Pearson Education 2023

[44] Gilbert, S., Harvey, H., Melvin, T. *et al.* Large language model AI chatbots require approval as medical devices. *Nat Med* (2023). https://doi.org/10.1038/s41591-023-02412-6

Annex A
Glossary

This section lists all the relevant abbreviations, acronyms and uncommon terms used in the document.

Acronym/Term	Expansion	Comment
AI	Artificial Intelligence	While the exact definition is highly controversial, in context of this document it refers to a field of computer science working on machine learning and knowledge-based technology that allows to *understand* complex (health related) problems and situations at or above human (doctor) level performance and providing corresponding insights (differential diagnosis) or solutions (next step advice, triage).
AI-MD	AI based medical device	
AI4H	Artificial intelligence for health	
AISA	AI-based symptom assessment	The abbreviation for the topic of this topic group.
API	Application Programming Interface	An interface following a defined structure, allowing computer systems to communicate.
AuI	Augmented Intelligence	
CC	Chief Complaint	See "Presenting Complaint".
CfTGP	Call for topic group participation	
CONSORT-AI	Consolidated Standards of Reporting Trials	
DD	Differential Diagnosis	
DEL	Deliverable	
FDA	Food and Drug administration	
FG	Focus Group	An instrument created by ITU-T providing an alternative working environment for the quick development of specifications in their chosen areas.
FGAI4H	Focus Group on AI for Health	
GDP	Gross domestic product	
GDPR	General Data Protection Regulation	
IIC	International Computing Centre	The United Nations data center that will host the benchmarking infrastructure.
IMDRF	International Medical Device Regulators Forum	
IP	Intellectual property	
ISO	International Standardization Organization	

ITU	International Telecommunication Union	The United Nations specialized agency for information and communication technologies – ICTs.
LMIC	Low-and middle-income countries	
MDR	Medical Device Regulation	
MMVB	Minimal minimal viable benchmarking	A simple benchmarking sandbox for understanding and testing the requirement for implementing the MVB. See chapter 5.2 for details.
MRCGP	Membership of the Royal College of General Practitioners	A postgraduate medical qualification in the United Kingdom run by the Royal College of General Practitioners.
MTS	Manchester Triage System	A commonly used system for the initial assessment of patients e.g.,in emergency departments.
MVB	Minimal viable benchmarking	
NGO	Non Governmental Organization	NGOs are usually non-profit and sometimes international organizations independent of governments and international governmental organizations that are active in humanitarian, educational, health care, public policy, social, human rights, environmental and other areas to affect changes according to their objectives. (from Wikipedia.en)
PC	Presenting Complaint	The health problems for which the user of a symptom assessment system seeks help.
PC	Primary Care	A pre-clinical triage level suggested by many symptom-checkers.
PII	Personal identifiable information	
PMCF	Post Market Clinical Follow Up	A requirement by regulators for Software as a medical device. This refers to clinical studies of the product in the real world that serve to show evidence of the claimed benefits of a medical device.
PROMs	Patient Reported Outcome Measures	Outcomes reported by patients (usually through questionnaires) about their quality of life
SaMD	Software as a medical device	
SDG	Sustainable Development Goals	The United Nations Sustainable Development Goals are the blueprint to achieve a better and more sustainable future for all. Currently there are 17 goals defined. SDG 3 is to "Ensure healthy lives and promote well-being for all at all ages" and is therefore the goal that will benefit from the AI4H focus groups work the most.

TDD	Topic Description Document	Document specifying the standardized benchmarking for a topic on which the FG AI4H topic group works. This document is the TDD for the topic group TG-Symptom
TG	Topic Group	
Triage		A medical term describing a heuristic scheme and process for classifying patients based on the severity of their symptoms. It is primarily used in emergency settings to prioritize patients and to determine the maximum acceptable waiting time until actions need to be taken.
WG	Working Group	
WHO	World Health Organization	

Annex B
Declaration of conflict of interests

In accordance with the ITU transparency rules, this section lists the conflict-of-interest declarations for everyone who contributed to this document. Please see the guidelines in FGAI4H-F-105 "ToRs for the WG-Experts and call for experts" and the respective forms (Application form & Conflict of interest form).

1DOC3

1DOC3 is a digital health startup based in Colombia and Mexico, was founded in 2014 and provide the first layer of access to affordable healthcare for spanish speaking people on their phone. 1DOC3 has developed a Medical Knowledge graph in Spanish and a proprietary AI assisted technology to improve user experience by effectively symptom checking, triaging and pre diagnosing, **optimizing doctors' time** allowing 1DOC3 to serve 350K consultations a month.

People actively involved: Lina Porras (linaporras@1doc3.com), Juan Beleño (jbeleno@1doc3.com) and María Fernanda González (mgonzalez@1doc3.com)

Ada Health GmbH

Ada Health GmbH is a digital health company based in Berlin, Germany, developing diagnostic decision support systems since 2011. In 2016 Ada launched the Ada-App, a DSAA for smartphone users, that since then has been used by more than 5 million users for about 30 million health assessments (beginning of 2023). The app is currently available in 11 languages and available worldwide. While Ada has many users in US, UK and Germany, it also launched a Global Health Initiative focusing on impact in LMIC where it partners with governments and NGOs to improve people's health.

People actively involved: Henry Hoffmann (henry.hoffmann@ada.com), Shubhanan Upadhyay (shubs.upadhyay@ada.com), Milan Jovanovic (milan.jovanovic@ada.com)

Involved before: Andreas Kühn, Clemens Schöll, Johannes Schröder, Sarika Jain, Isabel Glusman, Ria Vaidya, Martina Fischer, Ivan Lebovka

Babylon Health

Babylon Health is a London-based digital health company which was founded in 2013. Leveraging the increasing penetration of mobile phones, Babylon has developed a comprehensive, high-quality, digital-first health service. Users are able to access Babylon health services via three main routes: i) Artificial Intelligence (AI) services, via our chatbot, ii) "Virtual" telemedicine services and iii) physical consultations with Babylon's doctors (only available in the UK as part of our partnership with the NHS). Babylon currently operates in the U.K., Rwanda and Canada, serving approximately 4 million registered users. Babylon's AI services will be expanding to Asia and opportunities in various LMICs are currently being explored to bring accessible healthcare to where it is needed the most.

People actively involved: Saurabh Johri (saurabh.johri@babylonhealth.com), Adam Baker (adam.baker@babylonhealth.com)

Involved before: Nathalie Bradley-Schmieg (nathalie.bradley1@babylonhealth.com), Yura Perov

Baidu

Baidu is an international company with leading AI technology and platforms. After years of commercial exploration, Baidu has formed a comprehensive AI ecosystem and is now at the forefront of the AI industry in terms of fundamental technological capability, speed of productization and commercialization and "open" strategy. Baidu Intelligent Healthcare—an AI health-specialized division established in 2018—is seeking to harness Baidu's core technology assets to use evidence-based AI to empower primary health care. The division's technology development strategy was developed in collaboration with the Chinese government and industry thought leaders. It's building capacity in China's public health-care facilities at a grassroots level through the development of its Clinical Decision Support System (CDSS), an AI software tool for primary health-care providers built upon medical natural language understanding and knowledge graph technology. By providing explainable suggestions, CDSS guides physicians through the clinical decision-making process like diagnosis, treatment plans and risk alert. In the future, Baidu will continue to enhance user experience and accelerate the development of AI applications through the strategy of "strengthening the mobile foundation and leading in AI".

People actively involved: Yanwu XU (xuyanwu@baidu.com), Xingxing Cao (caoxingxing@baidu.com)

Barkibu

Barkibu is a pet health care and insurance company based in Coruña, Spain and founded in 2015. Through the Barkibu app, pet parents can get assistance on how to take care of their pets, check their symptoms and get immediate triage, talk to a live vet or find the best suited clinic for their pet's problem. We do this through a combination of an AI powered vet assistant that runs our proprietary algorithms fed with real case data, a chat & video telehealth platform and a comprehensive insurance coverage policy.

People actively involved: Francisco Cheda Pérez (fran@barkibu.com), Ernesto Hernández Cura (ernesto@barkibu.com)

Deepcare

Deepcare is a Vietnam based medtech company. Founded in 2018 by three co-founders. Actually, we provide a Teleconsultation system for vietnamese market. AI-based symptom checker is our core product. It actually is available only in vietnamese language.

People actively involved: Hanh Nguyen (hanhnv@deepcare.io), Hoan Dinh (hoan.dinh@deepcare.io), Anh Phan (anhpt@deepcare.io)

EQL

EQL is a digital health-tech organisation based in London, UK, which focuses on MSK conditions and physiotherapy. EQL's product, Phio Access, provides a conversational AI-enabled digital solution to support triage for MSK conditions. Phio Access is currently available to 9.5 million people in the UK and in active use by several major healthcare providers, including Circle, BMI,

Connect Health, Healthshare. EQL is currently working on its next-generation products, with the extended application of AI and ML technology for MSK medicine and physiotherapy.

People actively involved: Yura Perov (yura@eql.ai).

Flo Health

Flo Health is international company with offices London, Villnius, Minsk, Cyprus and USA. We are focused purely on women's health and our aim is to help women and girls prioritise their health by giving access to expert information, knowledge and support. We encourage our users to better understand how physiology affects their wellbeing. We are mainly a B2C company, available in 22 languages, we offer products including a menstrual cycle tracker, symptoms tracking and predictions and a dialog service that provides potential differentials for symptoms they are experiencing. We also have a very large content library for users to use and educate themselves on conditions and symptoms.

People actively involved: Dr Anna Klepchukova (CMO, a_klepchukova@flo.health) and Dr Saddif Ahmed (Medical Director, s_ahmed@flo.health)

Healthily Ltd

Healthily is a Norwegian company based in London. We have eight years' experience in the field, a team of 50 people and currently deliver next steps health advice based on symptoms and personal factors to 650,000 people a month. Healthily has worked on benchmarking with Leeds University's eHealth Department, NHS England and Imperial College London. We are keen to link all these initiatives together to create a globally recognised benchmarking standard.

People actively involved: Jonathon Carr-Brown (jcb@livehealthily.com), Matteo Berlucchi (matteo@livehealthily.com), Martin Cansdale (martin@livehealthily.com), Andras Meczner (andras@livehealthily.com)

Involved before: Aleem Qureshi, Audrey Menezes, Rex Cooper

Infermedica

Infermedica is a leading digital health company, specializing in AI-powered solutions for symptom analysis and patient triage. The company's mission is to make healthcare accessible, convenient and affordable for everyone worldwide, by automating primary care, from symptom to outcome. Infermedica has been adeptly interweaving medical and technical expertise into their technologies since 2012. Infermedica is now being used in more than 30 countries, in 20 languages and has completed more than 12 million successful health checks to date.

People actively involved: Dr. Irv Loh (irv.loh@infermedica.com), Piotr Orzechowski (piotr.orzechowski@infermedica.com), Mateusz Palczewski (mateusz.palczewski@infermedica.com), Mateusz Glod (mateusz.glod@infermedica.com)

Involved before: Jakub Winter (jakub.winter@infermedica.com), Michał Kurtys (michal.kurtys@infermedica.com)

Inspired Ideas

Inspired Ideas is a technology company in Tanzania that believes in using technology to solve the biggest challenges across the African continent. Their intelligent Health Assistant, Dr. Elsa, is powered by data and artificial intelligence and supports healthcare workers in rural areas through symptom assessment, diagnostic decision support, next step recommendations and predicting disease outbreaks. The Health Assistant augments the capacity and expertise of healthcare providers, empowering them to make more accurate decisions about their patients' health, as well as analyzes existing health data to predict infectious disease outbreaks six months in advance. Inspired Ideas envisions building a complete end-to-end intelligent health system by putting digital tools in the hands of clinicians all over the African continent to connect providers, improve health outcomes and support decision making within the health infrastructure that already exists.

People actively involved: Ally Salim Jr (ally@inspiredideas.io), Megan Allen (megan@inspiredideas.io)

Isabel Healthcare

Isabel Healthcare is a social enterprise based in the UK. Founded in 2000 after the near fatal misdiagnosis of the co-founder's daughter, the company develops and markets machine learning based diagnosis decision support systems to clinicians, patients and medical students. The Isabel DDx Generator has been used by healthcare institutions since 2001.Its main user base is in the USA with over 160 leading institutions but also has institutional users around the world, including emerging economies such as Bangladesh, Guatemala and Somalia . The DDx Generator is also available in Spanish and Chinese. The Isabel Symptom Checker and Triage system has been available since 2012. This system is freely available to patients and currently receives traffic from 142 countries. The company makes its APIs available so EMR vendors, health information and telehealth companies can integrate Isabel into their own systems. The Isabel system has been robustly validated since 2002 with several articles in peer reviewed publications.

People actively involved: Jason Maude (jason.maude@isabelhealthcare.com)

Kahun

Kahun is an Israeli based med-tech venture, founded in 2018, developed an AI virtual clinical in-take technology. Kahun has built an evidence-based medical knowledge graph (20M+ relations) and an AI engine that utilizes the graph to generate real-time insights. It enables Kahun to perform a patient interview and supply a patient decision support dashboard to the provider.

People actively involved: Michal Tzuchman Katz (michal@kahun.com)

Tom Neumark

I am a postdoctoral research fellow, trained in social anthropology, employed by the University of Oslo. My qualitative and ethnographic research concerns the role of digital technologies and data in improving healthcare outcomes in East Africa. This research is part of a European Research Council funded project, based at the University of Oslo, titled 'Universal Health Coverage and the Public Good in Africa'. It has ethical approval from the NSD (Norway) and NIMR (Tanzania); in accordance with this, the following applies: Personal information (names and identifiers) will be anonymized unless the participant explicitly wishes to be named. No unauthorized persons will

have access to the research data. Measures will be taken to ensure confidentiality and anonymity. More information available on request.

Visiba Group AB

Visiba Care supplies and develops a software solution that enables healthcare providers to run own-brand digital practices. The company offers a scalable and flexible platform with facilities such as video meetings, secure messaging, drop-ins and booking appointments. Visiba Care enables larger healthcare organisations to implement digital healthcare on a large scale and include multiple practices with unique patient offers in parallel. The solution can be integrated with existing tools and healthcare information systems. Facilities and flows can be added and customised as needed.

Visiba Care was founded in 2014 to make healthcare more accessible, efficient and equal. In a short time, Visiba Care has been established as a market-leading provider of technology and services in Sweden, enabling existing healthcare to digitalise their care flows. Through its innovative product offering and the value it creates for both healthcare providers and patients, Visiba Care has been a driving force in the digitalisation of existing healthcare. Through our platform, thousands of patients today can choose to meet their healthcare provider digitally. As of today, Visiba Care is active in 4 markets (Sweden, Finland, Norway and UK) with more than 70 customers and has helped facilitate more than 4.000.000 patient-care interactions.

We have been working specifically with AI-based symptom assessment and automated triage since 2018 and have developed and commercialised Red Robin – a medical device for automated anamnesis and triage.

People actively involved: Anastacia Simonchik (anastacia.simonchik@visibacare.com)

Annex C
Topic Group status updates for the focus group meetings

For the regular meetings of the focus group the TG-symptom prepare a status update which was then also presented by the topic driver at the meeting. The following sections contain the detailed status updates.

Status update for meeting D (Shanghai)

With the publication of the "call for participation" the current topic group members, Ada Health and Your.MD, started to share it within their networks of field experts. Some already declared general interest and are expected to join official via input documents at meeting D or E. Before the initial submission of the first draft of this TDD it was jointly edited by the current topic group members. Some of the approached experts started working on own contributions that will soon be added to the document. For the missing parts of the TDD where input is needed the topic group will reach out to field experts at the upcoming meetings and the in between.

Status update for meeting E (Geneva)

With Baidu joining at meeting D we introduced the topic group differentiation into the subtopics "self-assessment " and "clinical symptom assessment". The corresponding changes to this TDD have been started, however there at the current phase they are still quite close and will mainly differ in the symptom input space and condition output space. Shortly after meeting D Isabel Healthcare, one of the pioneers of the field for diagnostic decision support systems for non-academic use, joined the topic group for both subtopics. In the week before meeting E Babylon Health, a large London-based digital health company developing the popular Babylon symptom checker app, joint the topic group too.

With more than two participants, the topic group on 08.05.2019 started official online meetings. The protocol of the first meeting was distributed through the ai4h email reflector. We will also work on publishing the protocols in the website.

The refinement of the TDD involved primarily:

- adding the new members to the document
- adding the separation into two sub-topics
- the refinement of the triage section
- an improved introduction
- adding a section on benchmarking platforms including AICrowd

The detailed list of the changes is also listed in the "change notes" at the beginning of the document.

Status update for meeting F (Zanzibar)

During meeting E in Geneva, the topic group for the first time had a breakout session discussing the specific requirements for benchmarking of AISA systems in person. This meeting can be seen as the starting point for the multilateral work on a standardized benchmarking for this topic group.

It was decided that the main objective of the topic group for meeting F in Zanzibar was to create a Minimal Minimal Viable Benchmarking (MMVB). The goals of this step as an explicit step before the Minimal Viable Benchmarking (MVB) are:

- show a complete benchmarking pipeline for AISA
- with all parts visible so that we can all understand how to proceed
- get first benchmarking result numbers for Zanzibar
- learn relevant things for MVB that might follow in 1-2 meetings

For discussing the technical details of the MMVB the group held a meeting from 11 - 12 July 2019 in London. A first benchmarking system based on an Orphanet rare disease model was presented and discussed. The main outcomes of this meeting were as follows:

- An agreed-upon set of 11 conditions, 10 symptoms, 1 factor medical model to use for the MMVB.
- To use the pre-clinical triage levels "self-care", "consultation", "emergency", "uncertain" for MMVB
- The data structures to use for the inputs and outputs.
- The agreement on technology agnostic REST API calls for accessing AIs.
- The plan how to work together on drafting a guideline to create/annotate cases for benchmarking.

Based on the meeting outcomes in the following week a second Python based benchmarking framework using the agreed upon data structures and the 11 disease "London" model was implemented and shared via GitHub.

In addition to the London meeting the group had also 3 other phone calls. The following list shows all meetings together with their respective protocol links:

- 30.5.2019 - Meeting #2 - Meeting E Breakout Minutes
- 20.06.2019 - Meeting #3 - Telco Minutes
- 11-12.7.2019 - Meeting #4 - London Workshop Minutes
- 15.8.2019 - Meeting #5 - Telco Minutes
- 23.08.2019 - Meeting #6 - Telco Minutes

Since the last meeting the topic group was joined by Deepcare.io, Infermedica, Symptify and Inspired Ideas. Currently the topic group has the following members:

- Ada Health (Henry Hoffmann, Dr Shubhanan Upadhyay)
- Babylon Health (Saurabh Johri, Yura Perov, Nathalie Bradley-Schmieg)
- Baidu (Yanwu XU)
- Deepcare.io (Hanh Nguyen)
- Infermedica (Piotr Orzechowski, Dr Irv Loh, Jakub Winter)
- Inspired Ideas (Megan Allen, Ally Salim Jnr)
- Isabel Healthcare (Jason Maude)
- Symptify (Dr Jalil Thurber)

- Your.MD (Jonathon Carr-Brown, Rex Cooper)

At meeting E there was also the agreement that topic groups might have their own email reflector. Due to the significant number of members the topic group therefore decided to introduce fgai4htgsymptom@lists.itu.int as the groups email reflector.

Status update for meeting G (Delhi)

At the meeting F in Zanzibar the topic group presented a first MMVB - a "minimal minimal viable benchmarking". It showed a first benchmarking pipeline for AI-based symptom assessment systems using synthetic data sampled from a simplistic model and a collection of toy-AI. The main goal of the MMVB was to start learning what benchmarking for this topic group could look like. A simple model was chosen to gain insights in the first iteration, onto which more complex layers could be added for subsequent versions. For the latest iteration, the corresponding model and systems are called MMVB 2.0. In general, we expect to continue with further MMVB iterations until all details for implementing the first benchmarking with real data and real AI have been investigated - a version that is then called MVB.

As for the first MMVB iteration we have chosen a workshop format for discussing the technical details of the next benchmarking iteration. The corresponding workshop was held from 10-11.10.2019 in Berlin. As inclusiveness is a key priority for the focus group as a whole we also supported remote participation. In the meeting we agreed primarily on:

- Having independent from the MMVB 2 a more cloud based MMVB 1 version benchmarking cloud hosted toy AIs.
- The structure for how to encode attributes of symptoms and findings - a feature that is crucial for benchmarking self-assessment systems.
- A cleaner approach towards factors as the MMVB version.
- An approach how to continue with creation of benchmarking data.
- Exploring whether a 'pruned' subset within SNOMED exists for our use case (to map our symptom ontologies to)

Over the next weeks after the workshop the technical details have then been further refined. All together the have been the following meetings since meeting F:

- 03.08.2019 – Meeting #7 – Meeting F Breakout Minutes
- 27.09.2019 – Meeting #8 – Telco Minutes
- 10-11.10.2019 – Meeting #9 – Berlin Workshop Minutes
- 17.10.2019 – Meeting #10 – Telco Minutes
- 20.10.2019 – Meeting #11 – Telco Minutes
- 25.10.2019 – Meeting #12 – Telco Minutes
- 30.10.2019 – Meeting #13 – Telco Minutes

At the time of submission, the MMVB 2 version of the benchmarking software has not been completed yet. The plan is to present a version running on the new MMVB 2 model (also called the "Berlin Model") by the start of meeting G in Delhi.

While the Berlin Model relies on custom symptoms and condition the MVB benchmarking needs to use an ontology all partners can map to. In a teleconference call with SNOMED expert (Ian Arrowsmith) who had, in a prior role, been involved in creating SNOMED findings (minutes in meeting 12 as an addendum), discussion provided some avenues and contacts to help us discover whether it is indeed possible to find a refined subset of SNOMED for our use case to map common symptom and attribute ontologies to.

Beside the work on a MMVB 2 version of model and software we also started to investigate options for funding the independent creation of high-quality benchmarking data. Here we reached out to the Botnar Foundation and the Wellcome trust who have followed and supported the focus group since meeting A in Geneva. We expect to integrate their feedback for the funding criteria and requirements in one of the upcoming iterations of this document.

Since meeting F the group was joined by a new company Buoy (Eddie Reyes), mfine (Dr Srinivas Gunda), MyDoctor (Harsha Jayakody), Visiba Care (Anastacia Simonchik). For the first time the group was also joined by the individual experts Muhammad Murhaba (Independent Contributor, NHS Digital) and Thomas Neumark (Independent Contributor, University of Oslo) who supported the group with outreach activities and contributions.

Currently the topic group has the following 10 companies and 2 individuals as members:

- Ada Health (Henry Hoffmann, Dr Shubs Upadhyay)
- Babylon Health (Saurabh Johri, Yura Perov, Nathalie Bradley-Schmieg)
- Baidu (Yanwu XU)
- Buoy (Eddie Reyes)
- Deepcare.io (Hanh Nguyen)
- Infermedica (Piotr Orzechowski, Dr Irv Loh, Jakub Winter, Michal Kurtys)
- Inspired Ideas (Megan Allen, Ally Salim Jnr)
- Isabel Healthcare (Jason Maude)
- Muhammad Murhaba (Independent Contributor)
- MyDoctor (Harsha Jayakody)
- Symptify (Dr Jalil Thurber)
- Thomas Neumark (Independent Contributor)
- Visiba Care (Anastacia Simonchik)
- Your.MD (Jonathon Carr-Brown, Rex Cooper, Martin Cansdale)

The topic group email reflector fgai4htgsymptom@lists.itu.int altogether has currently 44 subscribers. The latest Meeting G version of this Topic Description Document lists 20 contributors.

Status update for meeting H (Brasilia)

Due to limited development resources (vacation, Christmas-break) since the last meeting, our work concentrated on extending the MMVB 1 system. We focused on a feature supporting the benchmarking of the cases defined by our doctors, in addition to the benchmarking with synthetic cases. The updated version has been published to GitHub and deployed to the demo system. The work also included adding another toy AI from the topic group member Inspired Ideas.

In the time since the last meeting the topic group had primarily one telco for aligning on the steps for meeting H:

- 06.12.2019 – Meeting #14 – Telco Minutes
- 06.01.2020 – Meeting #15 – Telco Minutes

In addition to this, our topic group also joined with three representatives the workshop of the DAISAM and DASH working groups from 8-9 of January 2020 in Berlin. We contributed there to all tracks and put emphasis on the special requirements of the benchmarking of systems for AI based symptom assessment. The results from these discussions will be reflected in this document over the next versions.

Since the last meeting, the topic group approached the Wellcome Trust and the Botnar foundation exploring funding options for the creation of case cards (for more info see 5.5 below). An initial phone call with the Wellcome Trust including Alexandre Cuenat (who previously attended the ITU/WHO AI4H meetings) was arranged. Mr. Cuenat offered to look into opportunities with Wellcome Centres. It was recommended that we look into direct funding options of the Wellcome Innovation stream e.g.,applying for an Innovator Award. The topic group also received an email from the Botnar foundation, stating that they would get back to us in January. Both opportunities require further exploration in the time after meeting G.

For the Meeting H version of this document we also merged the reformatting done by ITU and revised indexing and descriptions of tables and figures. With the introduction of the new SharePoint folder for all topic groups, our topic group started migrating all documents to the corresponding TG-Symptom folder https://extranet.itu.int/sites/itu-t/focusgroups/ai4h/tg/SitePages/TG-Symptom.aspx. As part of this, the latest TDD draft can always be found under https://extranet.itu.int/sites/itu-t/focusgroups/ai4h/tg/symptom/FGAI4H%20TG%20Symptom%20TDD%20draft.docx?d=wb569618c24f1445daa93f93aca2bb875. The protocols of all topic group internal meetings have also been uploaded to the folder and the references in this TDD have been updated accordingly.

Since meeting G there has also been some exchange with Baidu, who joined the topic group with a focus on the clinical symptom assessment. We are looking forward to integrating material on the benchmarking of AI systems in the clinical context for meeting I.

As our topic group is now one of the largest and longest existing ones, we have also been more involved in supporting the onboarding of new topic groups. For this we met with members of the newly formed topic group Dental Imaging to share insights on starting a topic group.

Since the submission for this TDD for meeting G, the topic group was joined by 1Doc3, Buoy, mFine and MyDoctor. MyDoctor and mFine joined meeting G and have been onboarded by the group during this meeting. With the new topic group members Buoy and 1Doc3 we conducted online onboarding meetings.

Currently the topic group has the following 14 companies and 2 individuals as members:

- 1Doc3 (Lina Porras)
- Ada Health (Henry Hoffmann, Dr Shubhanan Upadhyay, Dr Martina Fischer)
- Babylon Health (Saurabh Johri, Yura Perov, Nathalie Bradley-Schmieg)
- Baidu (Yanwu XU)
- Buoy (Eddie Reyes)

- Deepcare.io (Hanh Nguyen)
- Infermedica (Piotr Orzechowski, Dr Irv Loh, Jakub Winter, Michal Kurtys)
- Inspired Ideas (Megan Allen, Ally Salim Jnr)
- Isabel Healthcare (Jason Maude)
- Mfine (Dr Srinivas Gunda)
- Muhammad Murhaba (Independent Contributor)
- MyDoctor (Harsha Jayakody)
- Symptify (Dr Jalil Thurber)
- Thomas Neumark (Independent Contributor)
- Visiba Care (Anastacia Simonchik)
- Your.MD (Jonathon Carr-Brown, Rex Cooper, Martin Cansdale)

The topic group email reflector fgai4htgsymptom@lists.itu.int altogether has currently 56 subscribers (12 more than for Meeting G). The latest Meeting H version of this Topic Description Document lists 22 (2 more) contributors.

Status update for meeting I (Online E Meeting)

As the update for meeting H outlined, the work there was focused on extending the current MMVB version to support doctor cases and to connect more toy-AIs. With some new developers joining the topic group, since then we could focus more on the next important step of implementing the changes agreed upon at the Berlin workshop in November 2019. Beside a strong focus on the Berlin model extending the London model by symptom attributes and factors this also included more flexible frontend result report drill down, a more refined scoring and metric systems and in general moving the benchmarking system closer to the one needed for the MVB. Given the requirements of the Berlin model it became clear that implementing them would be easier if the software would be separated into dedicated frontend and backend applications, both using tech-stacks allowing to implement more complex features in a more stable and future-proof way. At the time of Meeting I this reimplementation is almost finished.

At meeting H the topic group was also joined by Alejandro Osornio, an expert for ontologies. In the weeks following he proposed a technical solution for how to use SNOMED CT for encoding the symptoms of the Berlin model. An overview of this work will be outline in section “Ontologies for encoding input data” (not in version yet) and based on this the current implementation work will integrate a mapping to an ontology earlier than expected. Continuing the ontology mapping after meeting I will be one of the priorities.

As suggested in the last meeting the focus group started the work on updating the FGAI4H-C-105 template for TDDs. Our topic group reviewed the draft and contributed the insights from working on this TDD. Once a new version is adopted by the focus group we will adjust this TDD to the new structure.

During meeting H the focus group discussed the possibility of working on a joint topic group overarching tool for creating and annotating benchmarking test data. As part of this discussion our topic group also contributed to an initial requirements document. After the meeting this discussion was continued in several online meetings with WG-DASH.

Since the last meeting we also intensified our online collaboration. For coordinating the implementation work we introduced a weekly tech telco. For bringing the clinical discussion on

scores and metrics forward the doctors inside the group also started a meeting series. The following list shows all the online meetings since the meeting H:

- 28.03.2020 – Meeting #17 – Telco Minutes
- 12.03.2020 – Meeting #18 – Tech Telco Minutes
- 13.03.2020 – Meeting #19 – Telco Minutes
- 20.03.2020 – Meeting #20 – Tech Telco Minutes
- 27.03.2020 – Meeting #21 – Telco Minutes
- 15.04.2020 – Meeting #22 – Tech Telco Minutes
- 22.04.2020 – Meeting #23 – Tech Telco Minutes
- 21.04.2020 – Meeting #24 – Clinical Telco (no minutes)
- 24.04.2020 – Meeting #25 – Telco Minutes

All the meeting notes can also be found in the official TG-Symptom SharePoint folder: https://extranet.itu.int/sites/itu-t/focusgroups/ai4h/tg/SitePages/TG-Symptom.aspx

We also started to publish our TG internal focus group meeting reports the. The summary of meeting H can be found here:

- TG-Symptom update on Meeting H

In addition to the meetings, we also now use the TG slack channel more for ad-hoc communication around technical implementation details and also for the clinical discussion (please reach out to the Topic Driver for details on how to join). Currently it is used by 21 people in the group.

Since Meeting H, we have been joined by three independent contributors, namely Pritesh Mistry, Alejandro Osornio and Salman Razzaki. One company (XUND, represented by Lukas Seper) also joined. In addition, Yura Perov (previously at Babylon) also joined in an independent capacity.

Currently, our topic group has the following 15 companies and 6 independent contributors:

- 1Doc3 (Lina Porras and Maria Gonzalez)
- Ada Health (Henry Hoffmann, Dr Shubs Upadhyay, Dr Martina Fischer)
- Alejandro Orsonio (Independent Contributor)
- Babylon Health (Saurabh Johri, Nathalie Bradley-Schmieg, Adam Baker)
- Baidu (Yanwu XU)
- Buoy (Eddie Reyes)
- Deepcare.io (Hanh Nguyen)
- Infermedica (Piotr Orzechowski, Dr Irv Loh, Jakub Winter, Michal Kurtys)
- Inspired Ideas (Megan Allen, Ally Salim Jnr)
- Isabel Healthcare (Jason Maude)
- Mfine (Dr Srinivas Gunda)
- Muhammad Murhaba (Independent Contributor)
- MyDoctor (Harsha Jayakody)

- Pritesh Mistry (Independent Contributor)
- Dr Salman Razzaki (Independent Contributor)
- Symptify (Dr Jalil Thurber)
- Thomas Neumark (Independent Contributor)
- Visiba Care (Anastacia Simonchik)
- XUND (Lukas Seper, Tamás Petrovics, Sophie Pingitzer)
- Your.MD (Jonathon Carr-Brown, Rex Cooper, Martin Cansdale)
- Yura Perov (Independent Contributor)

The topic group email reflector fgai4htgsymptom@lists.itu.int altogether has currently 83 subscribers (27 more than for Meeting H). The latest Meeting I version of this Topic Description Document lists 28 (6 more) contributors.

Status update for meeting J (Online E Meeting)

The work between meeting I and meeting J is divided into two large areas. The first focus was on the finalization of the implementation of the Berlin model. With the separation of the benchmarking system in frontend and backend the implementation was also finished by two teams, one on the backend side. While on both sides the data structures and interface had to be extended to the Berlin models more complex attribute and factor model, the frontend also improved usability and design. The backend had an additional focus to extend the case synthesizer generating the synthetic toy data used for testing the benchmarking system. Building on the new systems the members of the topic group started adapting their toy AIs to the new changed backend API interfaces and protocols. At the time of submission of the TDD version for meeting J three toy AIs have be completed with the others to follow in the weeks after meeting J.

With the current version of the software we also introduced the separation between the benchmarking system and the system for annotating/creating new cases by doctors. The corresponding annotation tool was also extended to support the Berlin model. Based on it we expect doctors to start creating benchmarking case vignettes before meeting J and continuing for the weeks after so that we again have the results for both synthetic and real cases. In anticipation of the upcoming next steps on extending the toy model with only 12 diseases and 12 symptoms to a fully condition and symptom space, we have already started to use SNOMED identifiers in the benchmarking system.

The second large area of work was dedicated to scores and metrics. For driving this forward the doctors inside the topic group formed a temporary breakout group working on a document covering all relevant aspects on this topic in full details.

After meeting I we also continued our contribution to a new template for a topic description documents. The resulting document was submitted as FGAI4H-J-004 to meeting J.

All the work in the topic group was organized online. The following list shows all the online meetings since the since meeting I:

- 29.05.2020 – Meeting #26 – Telco Minutes
- 11.06.2020 – Meeting #27 – Telco Minutes
- 26.06.2020 – Meeting #28 – Telco Minutes

- 10.07.2020 – Meeting #29 – Telco Minutes
- 07.08.2020 – Meeting #30 – Telco Minutes
- 21.08.2020 – Meeting #31 – Telco Minutes
- 04.09.2020 – Meeting #32 – Telco Minutes
- 18.09.2020 – Meeting #33 – Telco Minutes

For coordinating the implementation work we also continued the weekly tech telco, however having meeting minutes for them proved impracticable. For bringing the clinical discussion on scores and metrics forward the doctors of topic group also had additional telcos not listed here.

All the meeting notes can also be found in the official TG-Symptom SharePoint folder: https://extranet.itu.int/sites/itu-t/focusgroups/ai4h/tg/SitePages/TG-Symptom.aspx

We also published a topic group internal summary of meeting I that can be found here:

- TG-Symptom update on Meeting I

Since Meeting I, we have been joined by:

- Barkibu (Ernesto Hernandez and Francisco Cheda)
- EQL (Yura Perov)
- Dr Reza Jarral (Independent contributor)

Currently, our topic group has the following 17 companies and 6 independent contributors:

- 1Doc3 (Lina Porras and Maria Gonzalez)
- Ada Health (Henry Hoffmann, Dr Shubhanan Upadhyay, Ivan Lebovka, Nils Strelow)
- Alejandro Orsonio (Independent Contributor)
- Babylon Health (Saurabh Johri, Adam Baker)
- Baidu (Yanwu XU)
- Barkibu (Ernesto Hernandez)
- Buoy (Eddie Reyes)
- Deepcare.io (Hanh Nguyen)
- EQL (Yura Perov, who moved from Babylon to EQL)
- Infermedica (Piotr Orzechowski, Dr Irv Loh, Jakub Winter, Michal Kurtys)
- Inspired Ideas (Megan Allen, Ally Salim Jnr)
- Isabel Healthcare (Jason Maude)
- Dr Reza Jarral (Independent contributor)
- Mfine (Dr Srinivas Gunda)
- Muhammad Murhaba (Independent Contributor)
- MyDoctor (Harsha Jayakody)

- Pritesh Mistry (Independent Contributor)
- Dr Salman Razzaki (Independent Contributor)
- Symptify (Dr Jalil Thurber)
- Thomas Neumark (Independent Contributor)
- Visiba Care (Anastacia Simonchik)
- XUND (Lukas Seper, Tamás Petrovics, Sophie Pingitzer)
- Your.MD (Jonathon Carr-Brown, Rex Cooper, Martin Cansdale, Dr Audrey Menezes)

The topic group email reflector fgai4htgsymptom@lists.itu.int altogether has currently 99 subscribers (16 more than for Meeting I). The latest meeting I version of this Topic Description Document lists 29 (1 more) contributors.

Status update for meeting K (Online E Meeting)

With the finalization of the MMVB version implementing the Berlin model and a new benchmarking frontend and backend, the focus of work between Meeting J and Meeting K was one critical task of the topic group: agreeing on an ontology and approach for encoding realistic case data for the benchmarking.

As already for the London Model and Berlin Model iterations the work started with organizing the third topic group internal workshop from 12.11.2020 – 13.11.2020.

In preparation of the workshop all participants have been asked to prepare answers to the following questions:

1) **Procedure for agreeing on ontologies:** *How would you approach organizing the creation of a joined SNOMED-based ontology for symptoms, factors, attributes subset, profile details, expected conditions + all the necessary relations for the benchmarking?*
2) **Available Resources:** *What are the resources you can contribute until the next meeting for technical implementation, working on the joint ontology, creating case data for the benchmarking or updating/migrating the TDD to the new template?*
3) **Next MMVB iteration AIs:** *Under which conditions could you imagine to use already real AIs in the next MMVB iteration? Or should we just stick to toy-AIs for the time being?*
4) **Next MMVB and MVB Disease sets:** *Which set of diseases should we use for the next MMVB version?*
5) **AI metadata:** *What are the relevant metadata-fields that would be needed to describe the context you designed your AI for?*
6) **Benchmarking result sharing:** *How would you like the results of a benchmarking to be shared with the general public, stakeholders, your partners, internally etc.?*
7) **TDD Update work:** *Which sections of the TDD could you imagine to migrate/write/update?*

During the workshop the questions have then been discussed in detail. The main part of the discussion focused on question 1 about the approach for agreeing on a joint ontology. The key points form this have been:

1) We need to try the process of aligning with a few symptoms to see how this works and how to then use this as a blueprint for the general agreement process

2) We will use the same 11 abdominal related diseases we used in the Berlin Model, but extend the symptom space from the only 11 symptoms in the Berlin Model to all symptoms relevant to these disease
3) As a next step all companies create full detailed case for these diseases and/or lookup the symptoms the consider relevant for any of these diseases.
4) Based on these cases the symptoms and their attributes would be grouped/unified to identify the relevant information that needs to be encoded.
5) Based on this symptom/attribute set we would then meet again and try map them to SNOMED concepts and agree on the level of pre/post-coordination i.e. if it is "pain" + location "right lower quadrant of abdomen" or "abdominal pain" + location "right lower quadrant of abdomen" etc.

Following the workshop, the doctors in the topic group then created the corresponding case vignettes. The grouping of the symptoms and first steps towards mapping them to SNOMED have then be performed in corresponding follow-up meetings – a process that will continue after the submitting the first draft of the TDD migration work.

Beside the ontology there was also a discussion on point 3 where the consensus was that it should be open to everyone to use their real AIs and whether they do so via the public benchmarking API endpoints or only internally with a local test system. All other points had not been touched in greater detail and the TDD discussion was moved to a dedicated TDD related meeting.

In reporting period the topic group also contributed to the creation of the new TDD template FGAI4H-J-105 refined since Meeting J. In reflects many of the learnings from writing the earlier versions of this TDD and the way it was necessary to deviate from the original TDD template submitted by this topic group as FGAI4H-C-105 to Meeting C.

Based on TDD templated that was then accepted by the focus group via the online approval process out topic group also started the migration of this TDD document to the new format. Given the vacation, the late final approval and the size of the TG-Symptom TDD (97 pages) the version submitted for Meeting K is a work in progress version. In particular the sections 4 on ethics and on the theoretical background and the detailed descriptions of the latest benchmarking iterations could not be completed yet. The topic group also reviewed the ethics document FGAI4H-K-028, even though this took longer as requested.

The topic group also had some contact with the Open-Source initiative, however due to capacity limitations on both sides the original plan to implement a symptom-assessment benchmarking similar to the Berlin Model MMVB version was not realized until Meeting K.

All the work in the topic group was organized online. The following list shows all the online meetings since the since meeting J:

- 16.10.2020 – Meeting #34 – Telco Minutes
- 30.10.2020 – Meeting #35 – Telco Minutes
- 12.-13.11.2020 – Meeting #36 – Workshop #3 Minutes
- 25.11.2020 – Meeting #37 – Ontology Telco Minutes
- 27.11.2020 – Meeting #38 – Telco Minutes
- 11.12.2020 – Meeting #39 – TDD Telco Minutes
- 14.12.2020 – Meeting #40 – Ontology Telco Minutes
- 22.12.2020 – Meeting #41 – Ontology Telco (continued meeting #40 notes)

- 15.01.2020 – Meeting #42 – Telco Minutes

All the meeting notes can also be found in the official TG-Symptom SharePoint folder: https://extranet.itu.int/sites/itu-t/focusgroups/ai4h/tg/SitePages/TG-Symptom.aspx

Since Meeting J, we have been joined by:

- PnP (Opeoluwa Ashimi)

Currently, our topic group has the following 18 companies and 6 independent contributors:

- 1Doc3 (Lina Porras and Maria Gonzalez)
- Ada Health (Henry Hoffmann, Dr Shubs Upadhyay, Ivan Lebovka, Nils Strelow)
- Alejandro Orsonio (Independent Contributor)
- Babylon Health (Saurabh Johri, Adam Baker)
- Baidu (Yanwu XU)
- Barkibu (Ernesto Hernandez)
- Buoy (Eddie Reyes)
- Deepcare.io (Hanh Nguyen)
- EQL (Yura Perov, who moved from Babylon to EQL)
- Infermedica (Piotr Orzechowski, Dr Irv Loh, Jakub Winter, Michal Kurtys)
- Inspired Ideas (Megan Allen, Ally Salim Jnr)
- Isabel Healthcare (Jason Maude)
- Dr Reza Jarral (Independent contributor)
- Mfine (Dr Srinivas Gunda)
- Muhammad Murhaba (Independent Contributor)
- MyDoctor (Harsha Jayakody)
- PnP (Opeoluwa Ashimi)
- Pritesh Mistry (Independent Contributor)
- Dr Salman Razzaki (Independent Contributor)
- Symptify (Dr Jalil Thurber)
- Thomas Neumark (Independent Contributor)
- Visiba Care (Anastacia Simonchik)
- XUND (Lukas Seper, Tamás Petrovics, Sophie Pingitzer)
- Your.MD (Jonathon Carr-Brown, Rex Cooper, Martin Cansdale, Dr Audrey Menezes)

The topic group email reflector fgai4htgsymptom@lists.itu.int altogether has currently 104 (duplicates not counted) subscribers (5 more than for Meeting J). The latest meeting I version of this Topic Description Document lists 31 contributors (2 more).

Status update for meeting L (Online E Meeting)

Following Meeting K the topic group continued the work on defining a symptom ontology which had started back in workshop #3 from 12.11.2020 – 13.11.2020. After the doctors from the topic group created cases containing all symptoms (Figure 32 shows some of these cases) and after the symptom's different presentations had been clustered (See Figure 33 for an example), the group met again to review the results and discuss next steps.

ID	Company	Clinician	True Condition	Triage Level	Typicality	Biological sex	Age	Presenting Complaint(s)	Additional Evidence
1	Ada	Shubs	Apendicitis	EC	Typical	F	21	Sharp right lower quadrant pain for since yesterday (12 hours or so), getting progressively worse.	Vomited once this morning, no blood. Currently feels nauseated. No diarrhoea or constipation. Felt feverish since last night Temperature was 38.2C. No pv bleeding. Drinking fluids, urine output ok at present. Mouth feels dry. No dysuria. Feels generally unwell. Last menstrual period was 2 weeks ago. Denies recent unprotected sexual intercourse. Smoker. Nil other PMHx.
6	Infermedica	Michal	Irritable Bowel Syndrome	SC	Typical	F	25	abdominal pain,	abdominal pain lasting for 10 months, gets better on the weekends, increases in stressfull situations, gets better after defecation rectal pressure flatulence constipations and diarrheas no vomitting no weight loss no anorexia no gastrointestinal bleeding no fever fatigue
12	Independent	Reza	Acute Pyelonephritis	EC	Atypical	F	33	Fever and vomiting	2 days of progressive loin pain Sweaty Temperature was 39.2C Nausea and vomiting Anorexia Negative pregnancy test Some dysuria in the past week Lethargic and bed bound No frequency
25	PnP	PnP clinician	Viral Gastroenteritis	SC	Typical	M	2	Watery stool (3 episodes) Vomiting (5 episodes) Symptoms started 48 hrs. ago	Temperature >39 o C Symptoms persistent despite the use of antimalaria and antibiotics by the mother, mild abdominal discomfort Stool watery, non-bloody, non-mucoid Child taken care of by maid when mother goes to work, attends kindergarten where one other kid in his play group has similar symptom No abdominal distention, no dysuria, no history of food allergy or intolerance, no previous history of abdominal surgery
29	Independent	Eva	Acute Pyelonephritis	EC	Typical	F	25	Increasing flank pain and fever	Fever (39 °C) and chills for 1 day Increasing flank pain since morning Costovertebral angle tenderness Dysuria, frequent urination and urgency for 2 days Feels generally unwell and weak nausea, vomiting denies pregnancy
32	1Doc3	Maria/Lina	Acute cholecystitis		Typical	F	42	6 hour of moderate abdominal pain and right subcostal tenderness, asociated to vomiting.	Feels feverish Is overweight Has polycystic ovary syndrome Is taking oral contraceptives Smoking (+) Nauseas (+)

Figure 32 – Some of the case vignettes created by the doctors after workshop #3

```
abdominal pain
      finding site
            left
            right lower quadrant
            flank
            periumbilical
      quality
            cramping
            sharp
      intensity
            moderate
            intensity (8/10),
      time since onset / duration
            Over the last week
            for since yesterday (12 hours or so)
            Over the last week
            8 hours of
      progression / clinical course
            getting progressively worse
            increasing
            progressive
```

Figure 33 – Attributes of symptom "abdominal pain" collected from the workshop #3 case vignettes

One of the outcomes of this meeting was that as the next step the doctors would explore expressing the cases using SNOMED CT concepts. The focus was here on the symptoms and the attributes have therefore been ignored. Figure 34 shows an example of how this manual mapping looked. The work on this step showed that in general symptom mapping is feasible. It was also noted that SNOMED CT has a very strong bias towards professionally used clinical findings and not all lay use patient-reported details would be available at the level of detail supported by symptom assessment applications.

Case Snippets	Symptom	Snomed Name	Snomed Id	Mapping Quality
No diarrhoea	diarrhoea	Diarrhea (finding)	62315008	neg.
no diarrhoea				neg.
5 weeks of on-off diarrhoea with mucus				attributes missing
Intermittent diarrhoea for the last 2 months, no blood no mucus.				attributes missing
Chronic Diarrhoea "Poorly formed predominantly type 6 stool for 1 year Worse after heavy drinking Worse with work stress				attributes missing
diarrhea				perfect
Watery stool (3 episodes)				attributes missing
Stool watery, non-bloody, non-mucoid				attributes missing
diarrheas				attributes missing
no ... constipation	constipation	Constipation (finding)	14760008	neg.
constipations				attributes missing
Felt feverish since last night Temperature was 38.2C.	fever	Fever (finding)	386661006	attributes missing
no fever				neg.
NO fever				neg.
Fever				perfect
Temperature was 39.2C				attributes missing
Fever of 38C				attributes missing

Figure 34 – Example of workshop #3 symptoms phrases mapped to SNOMED CT (ignoring attributes)

In a subsequent meeting the topic group then discussed how to approach the mapping of attributes. During the discussion it became clear that the effort of defining which attributes are allowed for which symptoms and which attribute states are valid for an attribute in context of a given symptom will be a lot of work and that keeping this mapping up to date would cause a lot of maintenance work too. For this reason, we decided to first test if it would be feasible to not create such an attribute mapping and rely only on the SNOMED CT findings “as is”, possibly extended by only the most common attributes “severity” and parts of “clinical course”.

Based on this idea the topic group implemented the minimalistic SNOMED CT case creation tool depicted in Figure 35. Its main purpose was to allow the doctors in the group to see how well mapping the symptoms mentioned in the workshop #3 cases only using a search function. For this task the tool provided some metadata fields for author, expected disease, the factors age and sex, a field for the case vignette text and a comment field for providing feedback on how well it worked. The main feature was a search field that allowed searching findings in SNOMED CT and to add them as “presenting complaint”, “present” or “not present”. For this feature it was agreed to restrict the search to the finding subtree. In addition to this the doctors in the group reviewed the tree and excluded selected sub-trees and findings matching certain rules to narrow the tree down to the symptoms relevant to modelling self-assessment cases mainly consisting of patient reportable findings.

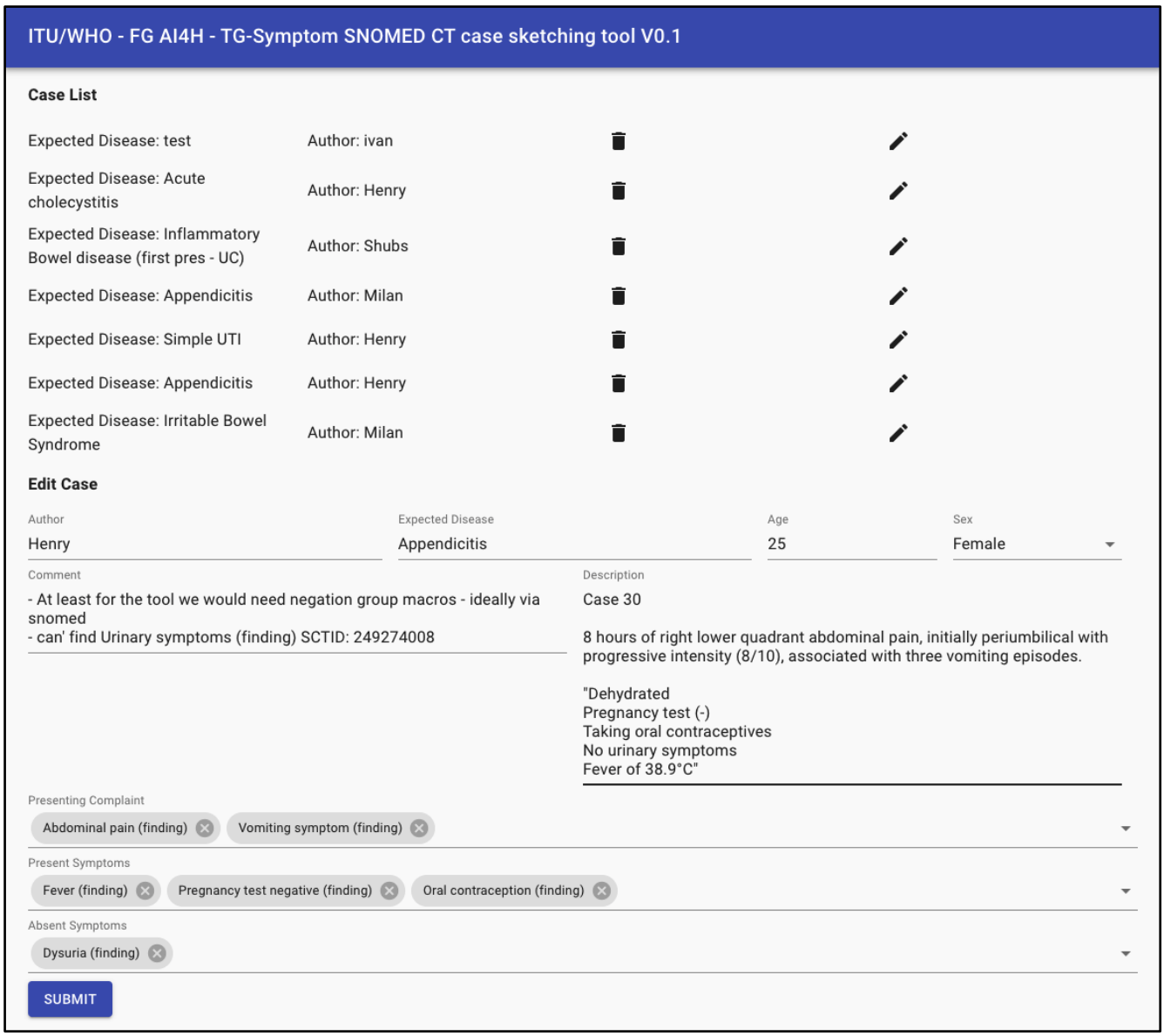


Figure 35 – Experimental first simple SNOMED CT based case creation tool

The tool was then tested by the doctors in the group. While this testing is still ongoing some of the insights so far are:

- For a realistic assessment of the approach, we need to integrate a real search engine supporting synonyms, ids, fuzzy search, shorter-match-preference, start-of-word-preference, perfect-match-preference etc. Even if fuzzy search is not supported the search provided by Snowstorm servers will likely provide a good starting point for this.
- We need to add a visualization of the hierarchies around the search concept to enable the selection of the right post-coordination level – especially for adding attribute details.
- We need to replace the plain tag-lists for the symptoms with a UI that allows to specify a few hand-picked attributes that apply to almost all findings.
- This might include a mechanism to visualize/edit the findings with attribute post-coordination of the same base finding together.
- We need to encourage the user to add “not present” symptoms as high as possible in the hierarchy.

- We need to provide easy means to negate common finding groups like “Urinary symptoms” which can consist of multiple concepts.
- We need to provide a built-in way to mark suggested findings as unappropriated for creating symptom-assessment cases for further improving the case creation tool.

Beside the work on the ontology workstream the group also continued the migration and extension of contents from our original TDD version into this new J-105 format. The main points added in in the submission for meeting L are:

- The migration and extension of the MMVB 2.0-2.2 descriptions summarized as the new section 6.1.2
- The migration and extension of the chapter on ethical considerations as chapter 4
- The migration of the existing work on benchmarking as chapter 5

Since meeting K there has also been some more exchange with the open-source initiative of the focus group, but so far, the test-wise integration of symptom-assessment as demo use-case has not been started.

All the work in the topic group was organized online. The following list shows all the online meetings since the since meeting K:

- 19.01.2020 – Meeting #43 – Telco Minutes
- 12.03.2020 – Meeting #44 – Telco Minutes
- 26.03.2020 – Meeting #45 – Telco Minutes
- 16.04.2020 – Meeting #46 – Telco Minutes
- 23.04.2020 – Meeting #47 – Telco Minutes
- 07.05.2020 – Meeting #48 – Telco Minutes

All the meeting notes can also be found in the official TG-Symptom SharePoint folder: https://extranet.itu.int/sites/itu-t/focusgroups/ai4h/tg/SitePages/TG-Symptom.aspx

Since Meeting K, we have been joined by:

- Nivi (David Tresner-Kirsch)

Yura Perov/EQL changed his role to “independent contributors”.

Currently, our topic group has the following 18 companies and 7 independent contributors:

- 1Doc3 (Lina Porras and Maria Gonzalez)
- Ada Health (Henry Hoffmann, Dr Shubhanan Upadhyay, Ivan Lebovka)
- Alejandro Orsonio (Independent Contributor)
- Babylon Health (Saurabh Johri, Adam Baker)
- Baidu (Yanwu XU)
- Barkibu (Ernesto Hernandez)
- Buoy (Eddie Reyes)

- Deepcare.io (Hanh Nguyen)
- Infermedica (Piotr Orzechowski, Dr Irv Loh, Jakub Winter, Michal Kurtys)
- Inspired Ideas (Megan Allen, Ally Salim Jnr)
- Isabel Healthcare (Jason Maude)
- Dr Reza Jarral (Independent contributor)
- Mfine (Dr Srinivas Gunda)
- Muhammad Murhaba (Independent Contributor)
- MyDoctor (Harsha Jayakody)
- Nivi (David Tresner-Kirsch)
- PnP (Opeoluwa Ashimi)
- Pritesh Mistry (Independent Contributor)
- Dr Salman Razzaki (Independent Contributor)
- Symptify (Dr Jalil Thurber)
- Thomas Neumark (Independent Contributor)
- Visiba Care (Anastacia Simonchik)
- XUND (Lukas Seper, Tamás Petrovics, Sophie Pingitzer)
- Your.MD (Jonathon Carr-Brown, Rex Cooper, Martin Cansdale, Dr Audrey Menezes)
- Yura Perov (Babylon and EQL before)

The topic group email reflector fgai4htgsymptom@lists.itu.int altogether has currently 109 (duplicates not counted) subscribers (5 more than for Meeting K). The latest meeting L version of this Topic Description Document lists 31 contributors.

Status update for meeting M (Online E Meeting)

Following meeting L, the work on the SNOMED annotation tool continued, starting with a meeting that structured the required next steps to address the learnings from meeting L. The focus was to improve the symptom search because it plays a central role in encoding test cases for benchmarking using SNOMED. To date, the search was implemented as a simple infix search over the normalized symptom name space. Testing this approach led to the conclusion that we need to improve the search by:

- supporting synonyms
- have a fuzzier search tolerant (also against typos)
- allowing search for identifiers
- shorter-match-preference
- start-of-word-preference
- perfect-match-preference

The group agreed that even if fuzzy search is not supported, the best course of action would be to update the annotation tool to use Snowstorm – the most frequently used server for hosting a SNOMED instance that provides a flexible API for querying all relevant concepts. (The swagger API specification of a Snowstorm instance can be found here: https://snowstorm-training.snomedtools.org/snowstorm/snomed-ct/swagger-ui.htm).

API access to the Snowstorm server has been implemented in the annotation tool backend. The corresponding search functionality was then provided via a new search API to the frontend application.

In the previous version of the case editor, the search was integrated as one tag lists for each group of symptoms: presenting complaints, present symptoms absent symptoms. While this was sufficient to test adding symptoms “as is” from SNOMED, it became clear that in addition we will need to explicitly support editing symptom details (such as attribute expression), meaning that the tag-list approach would be too simplistic. Together with the transition to the new search API, the UI was therefore separated into individual lists for each of the symptom categories and a dedicated section for the symptom search so that the search results could be added to any of the categories (see Figure 36)

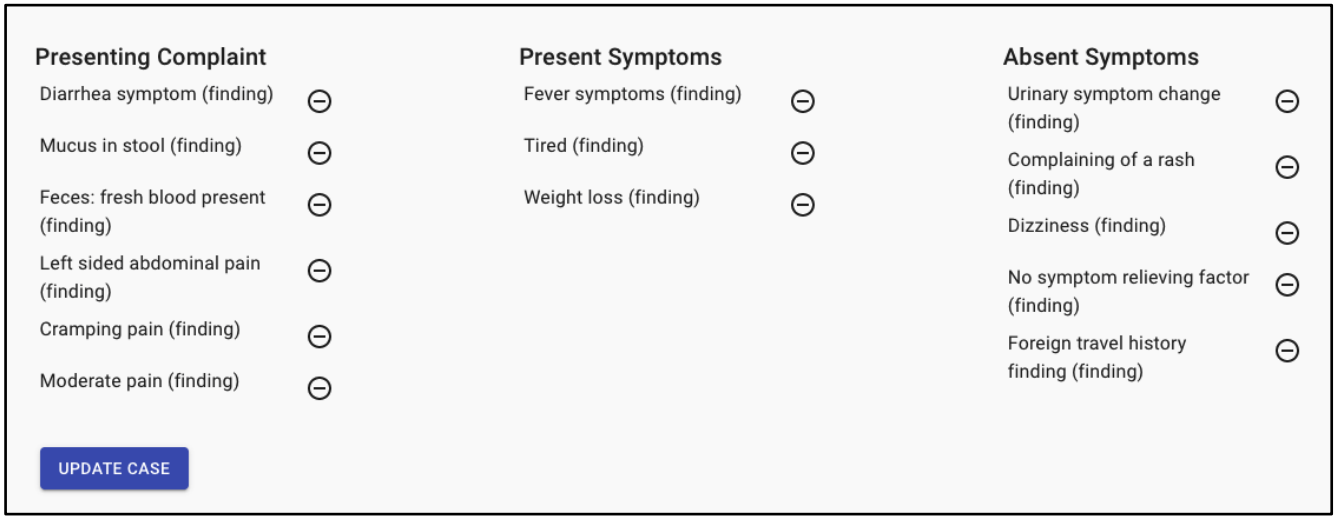


Figure 36 – Case symptoms separated by category

The design was aligned with the official SNOMED browser, separating the search result window and a window that shows the details for the selected search results (in particular the concept hierarchy around it as in many cases the desired concept is one of parents or children of the selected search result). Search, search results and the ancestors and children for the selected search result can be seen in Figure 37.

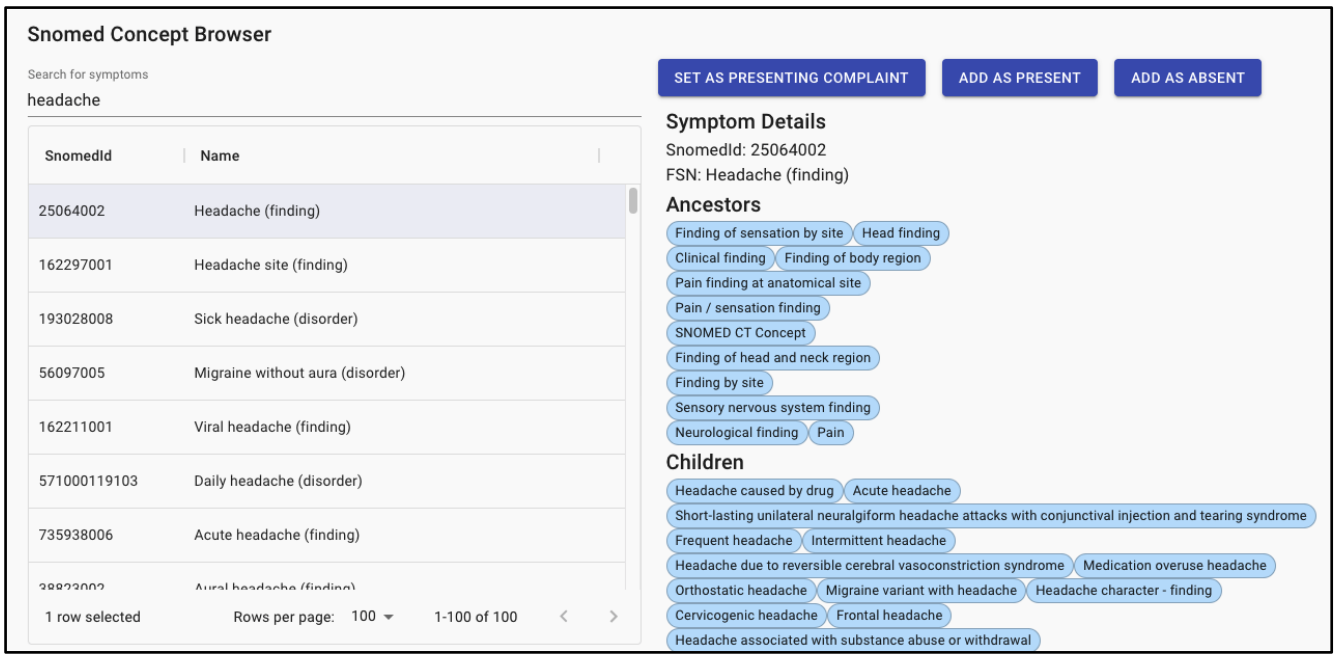


Figure 37 – SNOMED search results for "headache" (left side) and the ancestors and children for the selected "Headache (finding)" concept.

We moved the source code of the annotation tool to GitHub to work on the software in a cooperative way:

https://github.com/FG-AI4H-TG-Symptom/annotation_toy

The topic group also started to use the GitHub ticket system to organize the tasks:

https://github.com/FG-AI4H-TG-Symptom/annotation_toy/issues

Following meeting M, the next steps are to:

- implement an ECL based pre-filtering to filter better to concepts relevant for creating or annotating cases.
- add an attribute editor for the symptoms.
- add mechanisms marking and visualizing findings that have been used in other cases and/or should not be used.
- refine the hierarchy view.
- implement/integrate case and case set handling from previous versions.
- create realistic cases using the tool.

The next steps also include the support of cases created using the annotation tool in the MMVB benchmarking system implemented by the group. In addition to updating some of the toy-AIs, this includes the company internal implementation of a first mapping from the SNOMED symptom space to their native ontologies to confirm that the approach will generally work.

The end of this reporting period represented a milestone for the cooperation with the open-source initiative of the focus group. Even if TG-Symptom is not ML centric like most of the other topic groups, we have been chosen as one of the test use-cases. As part of this process, a group has been formed that will guide the implementation of the TG-symptom benchmarking in this framework. The current structure of this group can be found in this team allocation matrix:

https://docs.google.com/spreadsheets/d/17gDoEVA8qe_SBPyMlddl0dVzyTc29Y5O/edit#gid=1198500177

The first TG meeting #55 was already joined by a regulatory representative of this group (Carolin Prabhu) and further alignment meetings will likely take place soon after submitting this TDD iteration.

All the work in the topic group was organized online. The following list shows all the online meetings since the since meeting L:

- 11.06.2021 – Meeting #49 – Telco Minutes
- 15.07.2021 – Meeting #50 – Telco Minutes
- 30.07.2021 – Meeting #51 – Telco Minutes
- 20.08.2021 – Meeting #52 – Telco Minutes
- 23.08.2021 – Meeting #53 – Telco Minutes
- 03.09.2021 – Meeting #54 – Telco Minutes
- 17.09.2021 – Meeting #55 – Telco Minutes

All the meeting notes can also be found in the official TG-Symptom SharePoint folder: https://extranet.itu.int/sites/itu-t/focusgroups/ai4h/tg/SitePages/TG-Symptom.aspx

Since Meeting L there have been no new joiners. Within the group, Alejandro Orsonio now works for SNOMED International which will help the topic group's work, but in the meantime will continue his role as an Independent Contributor. Buoy's representative changed from Eddie Reyes to Sarah Hassonjee.

Currently, our topic group has the following 18 companies and 7 independent contributors:

- 1Doc3 (Lina Porras and Maria Gonzalez)
- Ada Health (Henry Hoffmann, Dr Shubhanan Upadhyay, Ivan Lebovka, Milan Jovanovic)
- Alejandro Orsonio (Independen Contributor)
- Babylon Health (Saurabh Johri, Adam Baker)
- Baidu (Yanwu XU)
- Barkibu (Ernesto Hernandez)
- Buoy (Sarah Hassonjee)
- Deepcare.io (Hanh Nguyen)
- Infermedica (Piotr Orzechowski, Dr Irv Loh, Jakub Winter, Michal Kurtys)
- Inspired Ideas (Megan Allen, Ally Salim Jnr)
- Isabel Healthcare (Jason Maude)
- Dr Reza Jarral (Independent contributor)
- Mfine (Dr Srinivas Gunda)

- Muhammad Murhaba (Independent Contributor)
- MyDoctor (Harsha Jayakody)
- Nivi (David Tresner-Kirsch)
- PnP (Opeoluwa Ashimi)
- Pritesh Mistry (Independent Contributor)
- Dr Salman Razzaki (Independent Contributor)
- Symptify (Dr Jalil Thurber)
- Thomas Neumark (Independent Contributor)
- Visiba Care (Anastacia Simonchik)
- XUND (Lukas Seper, Tamás Petrovics, Sophie Pingitzer)
- Your.MD (Jonathon Carr-Brown, Martin Cansdale, Dr Audrey Menezes)
- Yura Perov (Babylon)

The topic group email reflector fgai4htgsymptom@lists.itu.int altogether has currently 109 (duplicates not counted). The latest meeting M version of this Topic Description Document lists 31 contributors.

Status update for meeting N (Online E Meeting)

The work following meeting M was split into five workstreams. The first workstream focused on continuing the work on the annotation tool with the goal of finding a way to create annotated cases for benchmarking AI-based symptom assessment systems. Following meeting M, the tool supported symptom and finding search using the SNOMED ontology. Building on top of this, additional work was done to support the annotation of symptom attributes (e.g.,intensity or finding site). While SNOMED's support for findings is adequate, the precision and quality of attributes is not suboptimal to use for case encoding. Based on the approach outlined by our topic group's SNOMED expert, we implemented the following changes:

- Refactoring the case editor's tag-like symptom lists into tables to show more details for each symptom.
- Implementing a dedicated modal symptom editor to annotate attributes supported the symptoms/findings according to the SNOMED ontology.
- Implementing a sub-component to select the attributes states still available for post-coordinating the symptoms/findings (e.g.,a sub-structure of the abdomen if "abdominal pain" was selected).
- An additional comment field allowing our doctors to provide feedback on how well the attribute encoding worked.
- Implementing the necessary backend SnowStorm server API calls to query the pre-coordinated, post-coordinated attribute states and the attributes supported by symptoms/findings.

Figure 38 shows the modal finding editor with the new attribute selection feature.

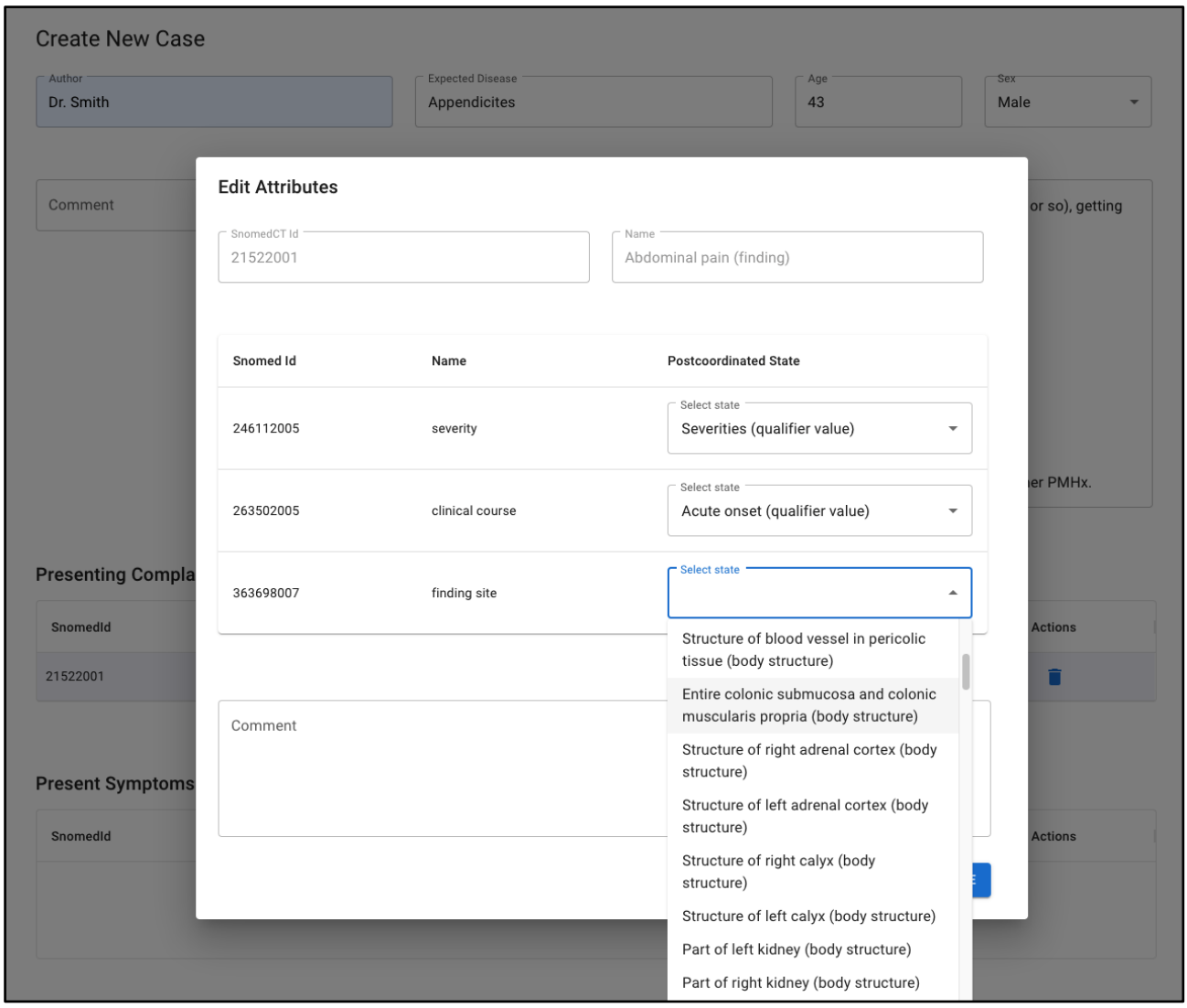


Figure 38 – Editor that describes attributes severity, clinical course and finding site of an abdominal pain finding.

We also implemented a feature that makes it easier for our doctors to identify the symptoms and findings that they most likely want to use in cases by highlighting the entities that have been used in other case vignettes. The feature also allows the explicitly marking of findings as inappropriate if they should be explicitly avoided (see Figure 39).

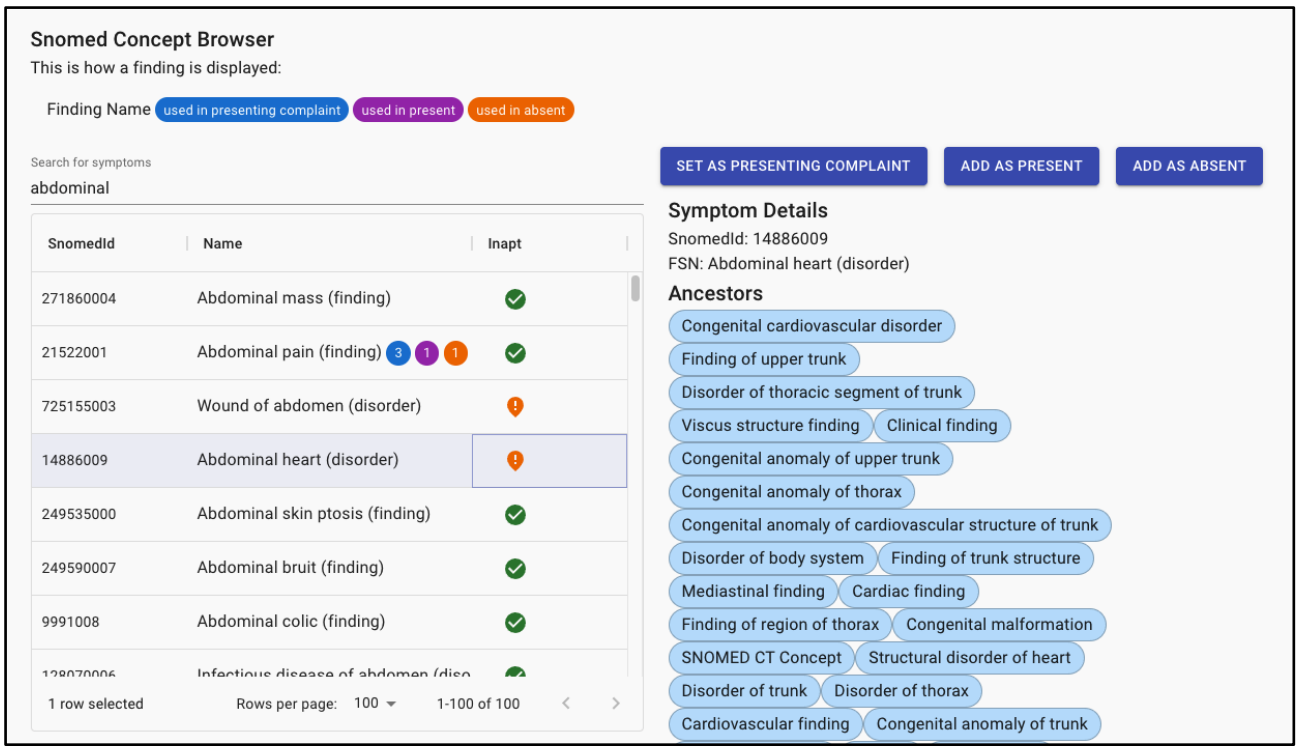


Figure 39 – SNOMED concept browser with the new feature showing how often concepts have been used in cases and if they are appropriate for use in case vignettes.

The technical work on the annotation tool was complemented by extending the clinical vignettes database that will be used for testing the annotation tool and performing a first benchmarking with the real AIs. The medical doctors of the topic group used two approaches to further enrich the database. First they reached out to the medical community to ask for support to create new clinical vignettes. The second approach consisted of internet search for high quality clinical vignettes that are freely available.

In the third workstream we explored the feasibility of integrating the TG-Symptom annotation tool with the recent developments on the "annotation package" developed by the focus group's open-code initiative. The topic group reached out to the group and discussed how the annotation tool could be registered and called as an external case editor. This included investigating how annotated TG-Symptom cases could be stored in the annotation package system. In preparation of calling our editor "stand-alone", the annotation tool was refactored to separate the case editor components from the case list and to accept case data via URL encoded parameters.

Shortly before meeting M, the TG-Symptom agreed to engage in the audit trial initiative as another showcase. While it was clearly stated that we will not be able to follow the proposed timeline and are unlikely to have the evaluation ready for publication before meeting N, we worked together with the corresponding TG-Symptom audit group on both an audit benchmarking script and on the audit process itself. As part of the work to implementan evaluation script we created an initial version published in the GitHub-repo that the audit group created for us:

- https://github.com/aiaudit-org/trial-audits-team-a-tg-symptoms

Any real audit trial would require the work on SNOMED-based encoding of cases to be completed. Here we focused on applying the audit framework on the MMVB 2.2 system using synthetic cases sampled from the Berlin model with 11 abdominal conditions and corresponding toy-AIs. Since these toy-AIs are hosted in the cloud to use them directly, the docker-based benchmarking approach + internet access would be needed, which was not available when the development was started.

Therefore, as an intermediate step, we implemented benchmarking based on the submission of solution files. To facilitate this approach, we created a helper script taking a data set exported from the MMVB 2.2. system's case-set page and converting into audit annotation files and AI input-files. We also implemented a script iterating all cloud hosted AI-systems and recording their response to the AI input cases in audit submission files that could then be used for manual solution upload and local offline testing. To evaluate the solutions, we extended the evaluation script to use the same TG-Symptom specific metrics also supported by the MMVB 2.2 system. The work reached the point where the scripts successfully run locally. We expect to present the benchmarking results in the audit system during meeting N. The general background and usage are described in detail in the corresponding readme file.

Independent of the technical audit work, we also worked with the TG-Symptom audit group on the formal side of the audit benchmark. We focused on the audit-checklist. Given that the TG-Symptom AIs largely apply non-ML AI-techniques, adapting the checklist to TG-Symptom needs was more challenging than expected. The current master is expected to be merged the week before meeting N. It is a combination of a more specific version of the original default checklist and technical points chosen from a list of TG-Symptom specific aspects relevant to understand the performance of systems for AI-based symptom assessment. The latest version of the merged checklist can be found here. The list of TG-Symptom specific technical details can be found here.

All the work in the topic group was organized online. The following list shows all the online meetings since the since meeting M:

- 12.10.2021 – Meeting #56 – Telco Minutes
- 15.10.2021 – Meeting #57 – Telco Minutes
- 29.10.2021 – Meeting #58 – Telco Minutes
- 03.11.2021 – Meeting #59 – Telco Minutes
- 12.11.2021 – Meeting #60 – Telco Minutes
- 26.11.2021 – Meeting #61 – Telco Minutes
- 02.12.2021 – Meeting #62 – Telco Minutes
- 10.12.2021 – Meeting #63 – Telco Minutes
- 21.01.2022 – Meeting #64 – Telco Minutes
- 04.02.2022 – Meeting #65 – Telco Minutes

All the meeting notes can also be found in the official TG-Symptom SharePoint folder:

https://extranet.itu.int/sites/itu-t/focusgroups/ai4h/tg/SitePages/TG-Symptom.aspx

In addition to the regular bi-weekly TG-Symptom meetings there have been weekly informal developer stand-ups to sync on technical details as well as informal management syncs. TG-Symptom also participated in several meetings of the TG-Symptom audit group.

Since meeting M, the topic group has been joined by:

- Flo (Anna Klepchukova, Saddif Ahmed)
- Kahun (Michal Tzuchman Katz)

Currently, our topic group has the following 20 (+2) companies and 7 independent contributors:

- 1Doc3 (Lina Porras and Maria Gonzalez)
- Ada Health (Henry Hoffmann, Dr Shubhanan Upadhyay, Ivan Lebovka, Milan Jovanovic)

- Alejandro Orsonio (Independent Contributor)
- Babylon Health (Saurabh Johri, Adam Baker)
- Baidu (Yanwu XU)
- Barkibu (Ernesto Hernandez)
- Buoy (Sarah Hassonjee)
- Deepcare.io (Hanh Nguyen)
- Flo (Anna Klepchukova, Saddif Ahmed)
- Infermedica (Piotr Orzechowski, Dr Irv Loh, Jakub Winter, Michal Kurtys)
- Inspired Ideas (Megan Allen, Ally Salim Jnr)
- Isabel Healthcare (Jason Maude)
- Kahun (Michal Tzuchman Katz)
- Dr Reza Jarral (Independent contributor)
- Mfine (Dr Srinivas Gunda)
- Muhammad Murhaba (Independent Contributor)
- MyDoctor (Harsha Jayakody)
- Nivi (David Tresner-Kirsch)
- PnP (Opeoluwa Ashimi)
- Pritesh Mistry (Independent Contributor)
- Dr Salman Razzaki (Independent Contributor)
- Symptify (Dr Jalil Thurber)
- Thomas Neumark (Independent Contributor)
- Visiba Care (Anastacia Simonchik)
- XUND (Lukas Seper, Tamás Petrovics, Sophie Pingitzer)
- Your.MD (Jonathon Carr-Brown, Martin Cansdale, Dr Audrey Menezes)
- Yura Perov (Babylon)

The topic group email reflector fgai4htgsymptom@lists.itu.int altogether has currently 117 (+8) (duplicates not counted) subscribers. The latest meeting N version of this Topic Description Document lists 34 (+3) contributors.

Status update for meeting O (Berlin)

In the reporting period between meeting N and meeting O we continued the work on the five active workstreams introduced in the previous report. After adding the new features for SNOMED-based symptom attribute specification, the work in the first workstream focused on testing these new features. For this the topic group doctors encoded about 30 cases and collected all observed bugs and usability issues, which then have been translated into corresponding tickets and discussed with the engineer responsible for the annotation tool. So far 50% of the issued have been resolved. The work is expected to be finished shortly after meeting O.

In preparation of the upcoming test-encoding of benchmarking cases, the topic group doctors also started to reach out to other doctors interested in contributing. This included both English native speakers as well as non-English native speakers to assess the robustness of the case annotation and process in this respect. In preparation of this work, we also started the revision of the necessary annotation guidelines.

In workstream three we continued the work on integrating TG-Symptom annotation tool with the annotation package developed by the focus group's open-code initiative. The topic group's project plan now includes migrating the annotation tool unchanged to the infrastructure of the open code initiative, integrating with the annotation package API and migrating storage of cases to the annotation package. Remaining tasks include specifying and implementing user roles, case creation and testing workflows and reporting of statistics.

Leaderboard

The TG-Symptom MMVB 2.2 toy AI leader board for the performance on the Official FG AI4H Meeting I Benchmarking test data set (71e9c086-2d8f-4cf9-bbe4-106117c95c5c).

Phase: Dev Phase, Split: Train Split Private

ıe submission * - Private submission Sort by best

Participant team	M1 (↑)	M3 (↑)	M10 (↑)	Triage accuracy (↑)	Triage similarity (↑)	Soft triage distance (↑)	Last submission at	Meta Attributes
Host_33953_Team (Ada Sampler) B	0.77	0.89	0.96	0.88	0.94	E	1 month ago	View
Host_33953_Team (Healthily Toy AI) B	0.74	0.87	0.92	0.74	0.86	E	1 month ago	View
Host_33953_Team (Random Sampler) B	0.04	0.08	0.11	0.18	0.42	E	1 month ago	View

Figure 40 – Screenshot of the first benchmarking results in the audit benchmarking system

In the time since meeting N we also continued the cooperation with the audit trial initiative and the corresponding TG-Symptom audit group. As part of this work, we could see first benchmarking results for the challenge we setup inside the audit benchmarking platform. Figure 40 shows the results with the expected performance scores already measured by the MMVB 2.0 benchmarking system previously developed by the topic group.

The cooperation with the audit group also covered the finalization of the audit questionnaire for TG-Symptom systems. Until early June the questionnaire will be implemented in the audit platform. In parallel we now investigate the scores and metrics for both, the qualitative results from the questionnaire and the quantitative results from the benchmarking. In contrast to the other topic groups participating in the audit trial we do not plan to publish any paper before the actual benchmarking mid 2023.

From 7.4.2022 to 8.4.2022 the topic group held its fourth workshop. The focus was here to plan out the remaining time until the final document submission deadline set by the focus group to mid 2023. As output of the workshop, we created a roadmap outlining all relevant foreseeable tasks. The main points are shown in Figure 41.

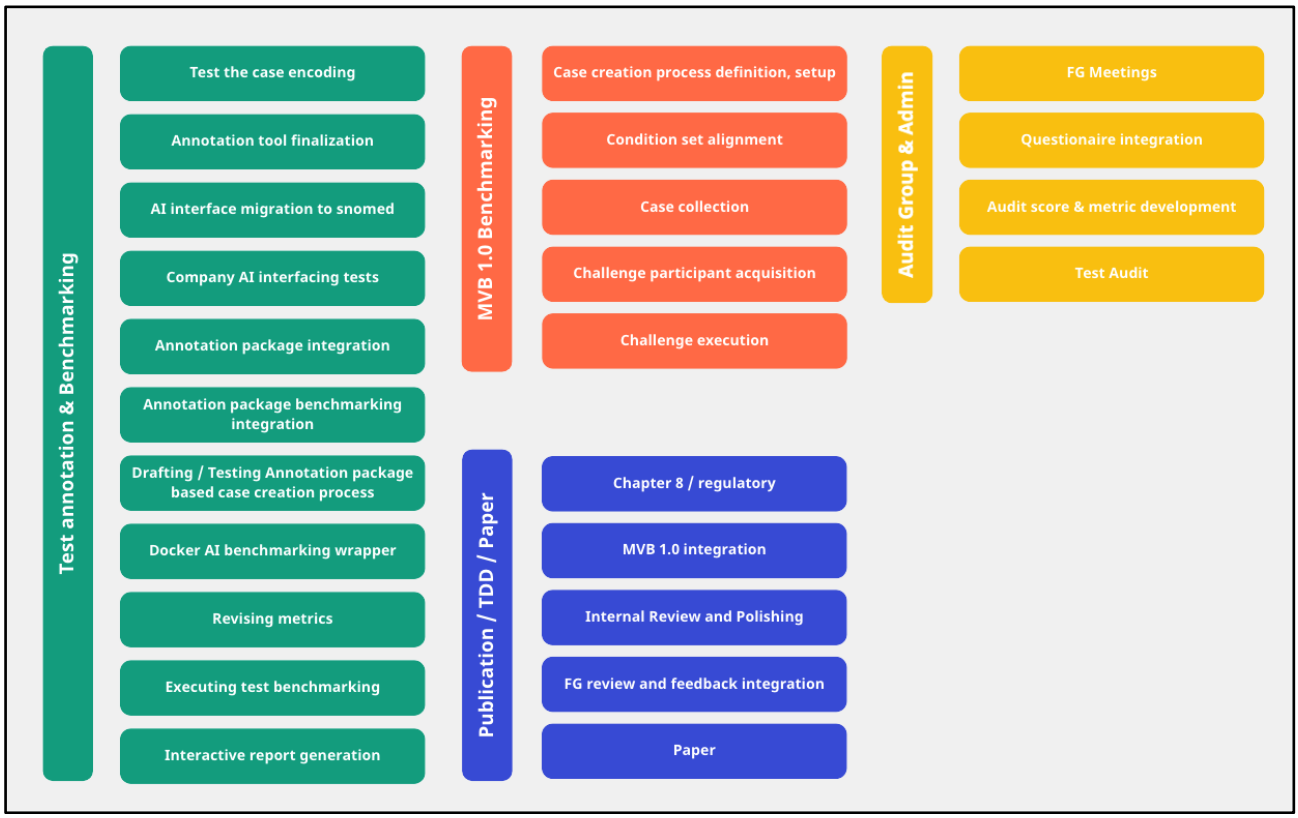


Figure 41 – Main roadmap items for the remaining time of the topic group.

The tickets identified during the workshop have been added to a new Jira instance setup for this purpose. With this transition we will stop using the github issue tracking that did not meet all requirements. According to the plan, in meeting #69 we also started reviewing the AI benchmarking interface. Different than expected we decided to investigate FHIR as format for encoding the benchmarking cases to send to the participating AIs which will increase the interoperability of the benchmarking cases.

During workshop #4 the topic group also agreed to nominate Martin Cansdale from Healthily (former Your.MD) as co-driver of the topic group to facilitate the implementation of the outlined roadmap. All the work in the topic group was organized online. The following list shows all the online meetings since the since meeting N:

- 18.02.2022 – Meeting #66 – Telco Minutes
- 01.04.2022 – Meeting #67 – Telco Minutes
- 07.04.2022 – 08.04.2022 – Meeting #68 – Workshop #4 Minutes
- 20.05.2022 – Meeting #69 – Telco Minutes

All the meeting notes can also be found in the official TG-Symptom SharePoint folder:

https://extranet.itu.int/sites/itu-t/focusgroups/ai4h/tg/SitePages/TG-Symptom.aspx

In addition to the regular bi-weekly TG-Symptom meetings there have been weekly informal developer stand-ups to sync on technical details as well as informal management syncs. TG-Symptom also participated in several meetings of the TG-Symptom audit group. We also introduced a weekly management call.

Currently, our topic group has the following 22 companies and 7 independent contributors:

- 1Doc3 (Lina Porras and Maria Gonzalez)
- Ada Health (Henry Hoffmann, Dr Shubhanan Upadhyay, Ivan Lebovka, Milan Jovanovic)
- Alejandro Orsonio (Independent Contributor)
- Babylon Health (Saurabh Johri, Adam Baker)
- Baidu (Yanwu XU)
- Barkibu (Ernesto Hernandez)
- Buoy (Sarah Hassonjee)
- Deepcare.io (Hanh Nguyen)
- Flo (Anna Klepchukova, Saddif Ahmed)
- Healthily (Jonathon Carr-Brown, Martin Cansdale, Dr Audrey Menezes)
- Infermedica (Piotr Orzechowski, Dr Irv Loh, Jakub Winter, Michal Kurtys)
- Inspired Ideas (Megan Allen, Ally Salim Jnr)
- Isabel Healthcare (Jason Maude)
- Kahun (Michal Tzuchman Katz)
- Dr Reza Jarral (Independent contributor)
- Mfine (Dr Srinivas Gunda)
- Muhammad Murhaba (Independent Contributor)
- MyDoctor (Harsha Jayakody)
- Nivi (David Tresner-Kirsch)
- PnP (Opeoluwa Ashimi)
- Pritesh Mistry (Independent Contributor)
- Dr Salman Razzaki (Independent Contributor)
- Symptify (Dr Jalil Thurber)
- Thomas Neumark (Independent Contributor)
- Visiba Care (Anastacia Simonchik)
- XUND (Lukas Seper, Tamás Petrovics, Sophie Pingitzer)
- Yura Perov (Babylon)

The topic group email reflector fgai4htgsymptom@lists.itu.int altogether has currently 121 (+4) (duplicates not counted) subscribers. The latest meeting O version of this Topic Description Document lists 37 contributors.

Status update for meeting P (Helsinki)

According to the plan outlined in the previous report, the work after meeting O continued with implementing the necessary annotation tool changes requested by the topic group doctors, followed by an additional round of re-testing.

Based on the Annotation Tool the doctors started to revise and extend the step-by-step annotation guidelines on how to use the tool to create cases. The work showed how crucial the quality of this document will be for the quality of the data sets for the benchmarking. Accordingly, we will give it more priority.

In parallel to writing the guidelines the doctors involved in the annotation tool development reached out to other doctors among the topic group members to ask for additional feedback on the usability of the annotation tool and in preparation to the upcoming first round of benchmarking test case creation. As part of this we had several meetings where we introduced doctors to the status of the topic group as well as the details on testing the annotation tool and creating benchmarking cases.

Beside finishing the annotation tool, the technical work since the last meeting focused on continuing the evaluation of FHIR as format for storing benchmarking cases. In several workshops we analysed the different structures provided by the FHIR specification and agreed on a first version of a FHIR encoded benchmarking case using SNOMED as the reference ontology. To make sure that the specified FHIR documents are valid, we implemented test code for generating cases by using the FHIR HAPI via Kotlin and the Python FHIR resources project. The current fully self-contained FHIR case use a top-level bundle containing observation resources for all the symptoms as well as the expected conditions and expected triage result. The format specifies means to represent present, non-present and skipped symptoms, attribute expressions and the representation of age and sex (which will explicitly not use a patient resource). We also planned out the next technical steps which will involve:

- Finalisation of the FHIR python script generating a case according to the agreed upon schema (done)
- Test-wise automated conversion of existing MMVB 2.2 cases to FHIR
- Implementation of AI endpoints accepting FHIR cases by TG members
- Update and test of the audit benchmark script to use the FHIR endpoints and FHIR cases
- Update AIs to use FHIR as output
- Update and test the audit benchmark to use the FHIR outputs
- Update the annotation tool to directly read/write FHIR cases

In this past reporting period, the topic group also continued the cooperation with the TG-Symptom audit group. The work focused here on further refining the audit questionnaire and test wise filling of it by Ada and Healthily. The filling was conducted in documents shared only between each company and the audit-group so that for instance the criticality of certain IP details can be discussed without already sharing them outside the audit-group. As next steps the questionnaires will now be evaluated as soon as the audio group agreed on the scoring metrics. Filling the questionnaires already showed that some of the questions likely need to be rephrased and some of the answers that might be necessary to interpret the benchmarking results are difficult to share because of business constraints and IP considerations.

All the work in the topic group was organized online. For the past reporting period we switched from having bi-weekly meetings to shorter but weekly calls to coordinate the work in the different workstreams. The notes of the meetings of the corresponding 8 online-meetings notes have been published as one single document:

- 07.06.2022 – 06.09.2022 – Telco Minutes

All the meeting notes can be found in the official TG-Symptom SharePoint folder:

https://extranet.itu.int/sites/itu-t/focusgroups/ai4h/tg/SitePages/TG-Symptom.aspx

TG-Symptom also participated in several meetings of the TG-Symptom audit group.

Currently, our topic group has the following 22 companies and 7 independent contributors (new contributors inside the companies are marked bold):

- 1Doc3 (Lina Porras, Maria Gonzalez, **Jhon Lince**)
- Ada Health (Henry Hoffmann, Dr Shubhanan Upadhyay, Ivan Lebovka, Milan Jovanovic)
- Alejandro Orsonio (Independent Contributor)
- Babylon Health (Saurabh Johri, Adam Baker)
- Baidu (Yanwu XU)
- Barkibu (Ernesto Hernandez)
- Buoy (Sarah Hassonjee)
- Deepcare.io (Hanh Nguyen)
- Flo (Anna Klepchukova, Saddif Ahmed)
- Healthily (Jonathon Carr-Brown, Martin Cansdale, Dr Audrey Menezes)
- Infermedica (Piotr Orzechowski, Dr Irv Loh, Jakub Winter, Michal Kurtys, **Mateusz Palczewski, Jakub Jaszczak**)
- Inspired Ideas (Megan Allen, Ally Salim Jnr)
- Isabel Healthcare (Jason Maude)
- Kahun (Michal Tzuchman Katz)
- Dr Reza Jarral (Independent contributor)
- Mfine (Dr Srinivas Gunda)
- Muhammad Murhaba (Independent Contributor)
- MyDoctor (Harsha Jayakody)
- Nivi (David Tresner-Kirsch)
- PnP (Opeoluwa Ashimi)
- Pritesh Mistry (Independent Contributor)
- Dr Salman Razzaki (Independent Contributor)
- Symptify (Dr Jalil Thurber)
- Thomas Neumark (Independent Contributor)
- Visiba Care (Anastacia Simonchik)
- XUND (Lukas Seper, Tamás Petrovics, Sophie Pingitzer)
- Yura Perov (Babylon)

The topic group email reflector fgai4htgsymptom@lists.itu.int altogether has currently 126 (+5) (duplicates not counted) subscribers. The latest meeting P version of this Topic Description Document lists 37 contributors.

Status update for meeting Q (Douala)

The work since Meeting P focused on three work streams. In the first, technical workstream we continue the transition from our own MMVB 2.2 format for describing benchmarking cases to a FHIR-based case descriptions using SNOMED for encoding symptoms. Based on the agreed upon draft of a TG-Symptom FHIR case we testwise extended one of the Toy-AIs by an API endpoint that accepts cases in the new FHIR format. The updated AI was also deployed to a server where it is now available for the audit benchmarking system. The audit-script was also updated to translate MMVB 2.2 benchmarking cases into FHIR cases and to send AIs the benchmarking cases based on a flag either in the old format or in FHIR. The benchmarking results for the updates AI are still not identical to the MMVB 2.2 version due to details with the handling of attributes specifying the details of the symptom expressions (e.g.,intensity, location, etc.). The next technical steps around the FHIR encoding include:

- Finalize the attribute handling of the FHIR AI endpoints
- Decide on the implementation of the time-since-onset
- Update AIs to use FHIR as output format
- Update and test the audit benchmarking to use the FHIR outputs
- Update the annotation tool to directly read/write FHIR cases

The technical work also included the continued exchange with the open code initiative about the integration of our annotation tool with the annotation package. The topic group's case annotation tool has here been repackaged to facilitate deployment on new infrastructure. Next steps involve:

- Deployment of the annotation tool on ITU infrastructure (we also consider deploying an own annotation package instance to a TG-server as intermediate solution)
- Integration of this new deployment with the OCI annotation package
- Creation of test tasks in the annotation package and verifying that these can be carried out in the topic group annotation tool

Following the initial refinement of the Annotation Tool completed before the last meeting, in the second workstream the doctors refined the case set of 35 cases with adaptations for how the tool works, based on their experience of encoding cases into the tool. They also completed the first version of the step-by-step annotation guidelines on how to encode cases into the Annotation Tool. Subsequently, the guidelines, the Tool and a randomised selection of cases from the case set was sent to other doctors in the topic group who volunteered to be ‘first testers’ of the Tool. Through the tool, they were able to note down their feedback regarding the guidelines and their use of the Tool. The feedback was then reviewed by the doctors in the topic group and action points were created from these regarding improvement to the guidelines and the Tool. These included improvements to instructions for encoding, technical improvements to the tool and recommendations for increased clarity of several features of the tool.

In this reporting period we also continued the cooperation with the TG-Symptom audit group. After providing the group experts with the testwise filled benchmarking participant questionnaires, they started the analysis in parallel to designing the corresponding evaluation criteria. Due to time restrictions this work was mostly performed asynchronously and the next synchronization meetings

are expected in December 2022.

All the work in the topic group was organized online. In the past reporting period, we continued with the weekly calls to coordinate the work in the different workstreams. The notes of the meetings of the corresponding 9 online-meetings notes have been published as one single document:

- 13.09.2022 – 29.11.2022 – Telco Minutes

All the meeting notes can be found in the official TG-Symptom SharePoint folder:

https://extranet.itu.int/sites/itu-t/focusgroups/ai4h/tg/SitePages/TG-Symptom.aspx

TG-Symptom also participated in several meetings of the TG-Symptom audit group.

Currently, our topic group has the following 22 companies and 7 independent contributors:

- 1Doc3 (Lina Porras, Maria Gonzalez, Jhon Lince)
- Ada Health (Henry Hoffmann, Dr Shubhanan Upadhyay, Milan Jovanovic)
- Alejandro Orsonio (Independent Contributor)
- Babylon Health (Saurabh Johri, Adam Baker)
- Baidu (Yanwu XU)
- Barkibu (Ernesto Hernandez)
- Buoy (Sarah Hassonjee)
- Deepcare.io (Hanh Nguyen)
- Flo (Anna Klepchukova, Saddif Ahmed)
- Healthily (Jonathon Carr-Brown, Martin Cansdale, Dr Audrey Menezes)
- Infermedica (Piotr Orzechowski, Dr Irv Loh, Michal Kurtys, Mateusz Palczewski, Jakub Jaszczak)
- Inspired Ideas (Megan Allen, Ally Salim Jnr)
- Isabel Healthcare (Jason Maude)
- Kahun (Michal Tzuchman Katz)
- Dr Reza Jarral (Independent contributor)
- Mfine (Dr Srinivas Gunda)
- Muhammad Murhaba (Independent Contributor)
- MyDoctor (Harsha Jayakody)
- Nivi (David Tresner-Kirsch)
- PnP (Opeoluwa Ashimi)
- Pritesh Mistry (Independent Contributor)
- Dr Salman Razzaki (Independent Contributor)
- Symptify (Dr Jalil Thurber)
- Thomas Neumark (Independent Contributor)

- Visiba Care (Anastacia Simonchik)
- XUND (Lukas Seper, Tamás Petrovics, Sophie Pingitzer)
- Yura Perov (Babylon)

The topic group email reflector fgai4htgsymptom@lists.itu.int altogether has currently 129 (+3) (duplicates not counted) subscribers. The latest meeting Q version of this Topic Description Document lists 37 contributors.

Status update for meeting R (Cambridge)

Shortly after Meeting R the topic group was joined by several new doctors, which allowed us to test the updated annotation guidelines and the annotation tool by letting them encode some of the test case vignettes prepared by other topic group doctors before.

The feedback from this work included both technical aspects and feedback on the annotation guidelines. In response to the technical feedback on the annotation tool we updated the following details:

- For encoding "time since onset" of symptoms and findings (e.g.,for encoding "headache for 3 days") we added two explicit fields "time since onset value" and "time since onset unit" (days, weeks, months, years) and update the case DB format accordingly. The previous approach of encoding it via the "clinical course" attribute and states like "subacute onset" have been considered too ambiguous.
- The sticky input fields for comment and description have been moved from the top to the right to make the transcription of case vignettes easier.
- We also introduced a case title in the case editor and case list which can now be used to better identify cases encoded for a specific purpose – especially if the same case was encoded by the same person several times for different use cases.

For the test the annotation tool was also deployed to a TG internal AWS instance.

The test encoding also showed that the annotation guidelines could be better structured. We started therefore to split the document into the following three distinct ones:

- An introduction providing background and context on the AI4H focus group, the standardized benchmarking idea, the topic group on AI-based symptom assessment as well as the importance and purpose of encoding benchmarking cases.
- A "handbook" on the annotation tool describing all technical features and how to use them.
- A "how to encode a case" tutorial explaining the practicalities around the encoding a case – either from an existing case vignette or from scratch.

Since meeting Q we also continued the exchange with the annotation package work stream of the open code initiative. Here we meanwhile got access to the AWS cluster so that we can start the actual integration of our annotation tool as editor for TG-symptom annotation tasks.

The updated medical and technical next steps list includes:

- Updating the annotation guidelines
- Completing the parallel encoding tests
- Integration with the OCI annotation package
- Creation of test tasks in the annotation package and verifying that these can be carried out in the topic group annotation tool

- Finalize the attribute handling of the FHIR AI endpoints
- Decide on the implementation of the time-since-onset
- Update AIs to use FHIR as output format
- Update and test the audit benchmarking to use the FHIR outputs
- Update the annotation tool to directly read/write FHIR cases

All the work in the topic group was organized online. In the past reporting period, we continued with the weekly calls to coordinate the work in the different workstreams. The notes of the meetings of the corresponding 10 online-meetings notes have been published as one single document:

- 13.12.2022 – 14.03.2023 – Telco Minutes

All the meeting notes can be found in the official TG-Symptom SharePoint folder:

https://extranet.itu.int/sites/itu-t/focusgroups/ai4h/tg/SitePages/TG-Symptom.aspx

TG-Symptom also participated in several meetings of the TG-Symptom audit group.

Currently, our topic group has the following 22 companies and 9 (+2) independent contributors:

- 1Doc3 (Lina Porras, Maria Gonzalez, Jhon Lince)
- Ada Health (Henry Hoffmann, Dr Shubhanan Upadhyay, Milan Jovanovic)
- Alejandro Orsonio (Independent Contributor)
- **Dr Audrey Menezes (Independent Contributor)**
- Babylon Health (Saurabh Johri, Adam Baker)
- Baidu (Yanwu XU)
- Barkibu (Ernesto Hernandez)
- Buoy (Sarah Hassonjee)
- Deepcare.io (Hanh Nguyen)
- **Dejan Hajduković (Independent Contributor)**
- Flo (Anna Klepchukova, Saddif Ahmed)
- Healthily (Jonathon Carr-Brown, Martin Cansdale, **Aleem Qureshi andras Meczner)**
- Infermedica (Piotr Orzechowski, Dr Irv Loh, Mateusz Palczewski, **Mateusz Glod**)
- Inspired Ideas (Megan Allen, Ally Salim Jnr)
- Isabel Healthcare (Jason Maude)
- Kahun (Michal Tzuchman Katz)
- Dr Reza Jarral (Independent contributor)
- Mfine (Dr Srinivas Gunda)
- Muhammad Murhaba (Independent Contributor)
- MyDoctor (Harsha Jayakody)
- Nivi (David Tresner-Kirsch)
- PnP (Opeoluwa Ashimi)

- Pritesh Mistry (Independent Contributor)
- Dr Salman Razzaki (Independent Contributor)
- Symptify (Dr Jalil Thurber)
- Thomas Neumark (Independent Contributor)
- Visiba Care (Anastacia Simonchik)
- XUND (Lukas Seper, Tamás Petrovics, Sophie Pingitzer)
- Yura Perov (Babylon Health)

Since the last meeting Dr Menezes moved from Healthily to an independent contributor role while Aleem Qureshi andras Meczner joined as new doctors from their side. Dejan Hajduković joined as new an independent contributor.

The topic group email reflector fgai4htgsymptom@lists.itu.int altogether has currently 131 (+2) (duplicates not counted) subscribers. The latest meeting R version of this Topic Description Document lists 37 contributors.

Annex D
MMVB 3.x case encoding tool manual

ITU/WHO Focus Group AI4H (FG-AI4H)
Topic Group Symptom Assessment (TG-Symptom)

MMVB 3.x Case Encoding Tool Manual

2023-06-25 V1.0

The following is a description of the Case Encoding Tool and its features. This tool for encoding of clinical vignettes is developed by the engineering team of TG Symptom. While explaining features of the Clinical Case Encoding tool this part of the document will only touch upon the case encoding process itself. Detailed, step-by-step instructions for how to encode a case will be provided in the section following this one.

1. To start encoding a new case click on the "Add Case" button of the Case Encoding tool (see Figure D.1). This opens the "Create New Case" page, which consists of several sections.

    a. Right-hand side of this page contains the "Case Encoding Comment" box and the "Case Description" box (see Figure D.2). To help with the case encoding process this section is locked to the right-hand side of the page and visible at any point of the case encoding process.

    b. First section on the left-hand side is dedicated for the title of the clinical vignette to be added (see Figure D.3).

    c. Second section on the left-hand side contains five editing boxes (see Figure D.3). Only the first box, 'Author,' requires information about the user/healthcare professional encoding the case (name and surname of the case encoder), while the other boxes require information about the case itself (the case condition/disease, triage level and the age and sex of the patient in the case).

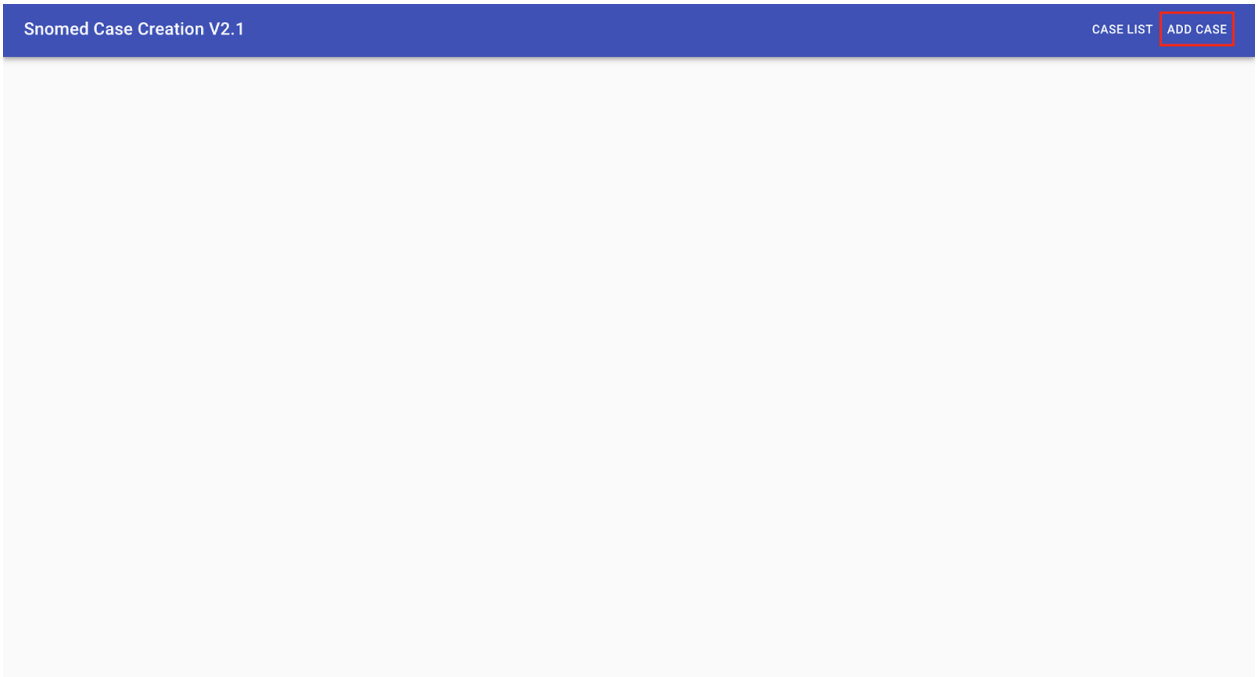


Figure D.1 – Case Encoding Tool will ultimately be used by clinicians with the goal of creating new clinical case vignettes to be used for the independent benchmarking of symptom assessment tools globally.

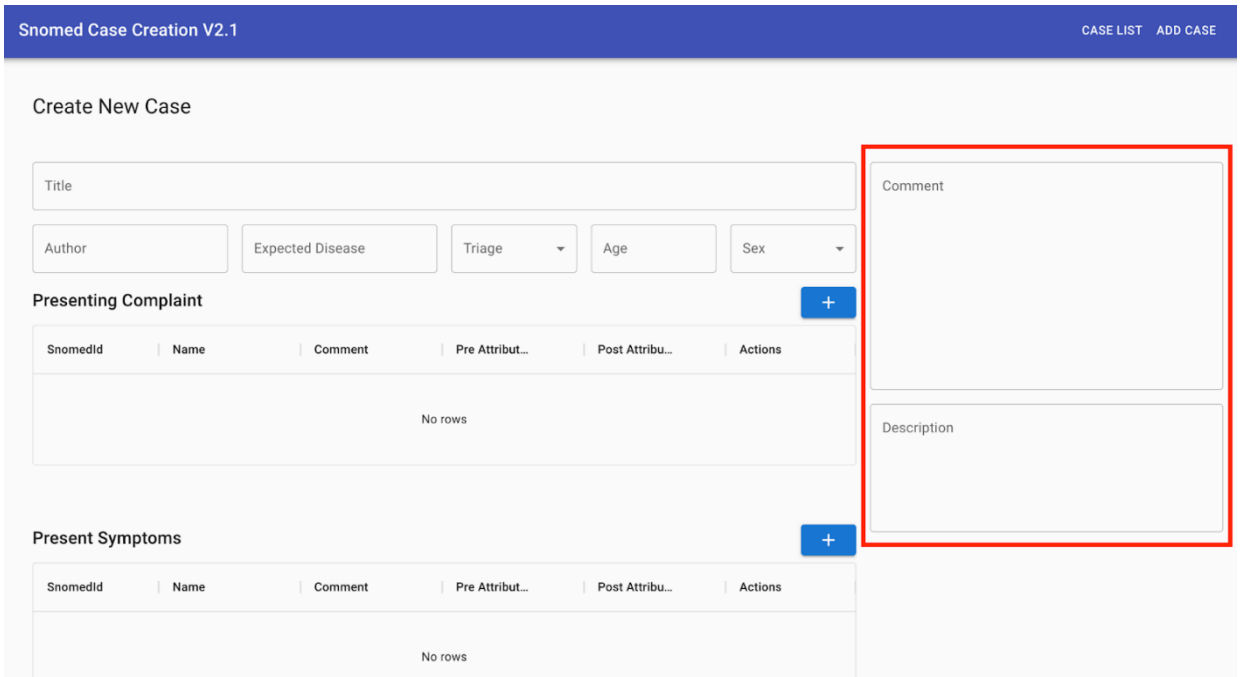


Figure D.2 – The "Comment" box is there to allow a user (case annotator) to report any issues observed during the case creation, e.g., issues with using the Case Encoding Tool, describing the case etc., while the "Case description" box serves the purpose of storing all evidence relevant for the case as reference while encoding the case.

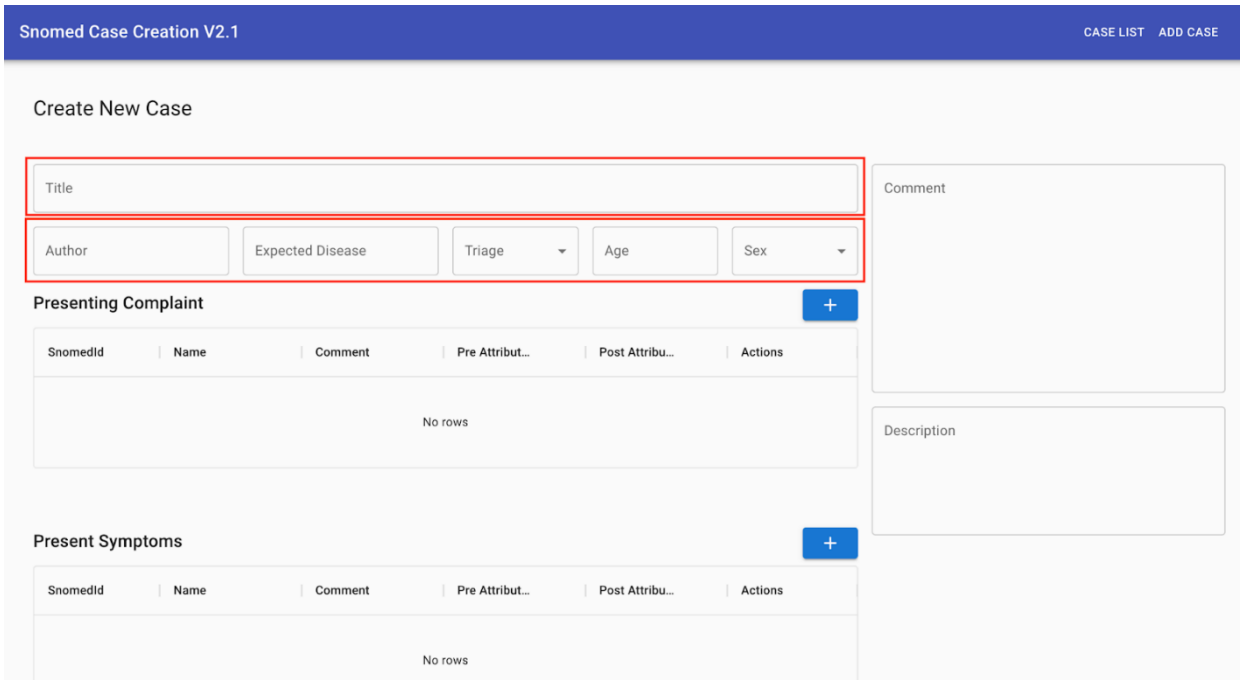


Figure D.3 – The case encoding process starts already at this stage by entering the clinical vignette title, the disease the case is describing, the triage level for the expected disease and the age and sex of the virtual patient.

d. Remaining three sections (see Figure D.4) allow a user to encode:
   i. (A) Presenting complaints,
   ii. (B) Present symptoms and
   iii. (C) Absent symptoms reported in the case vignette.

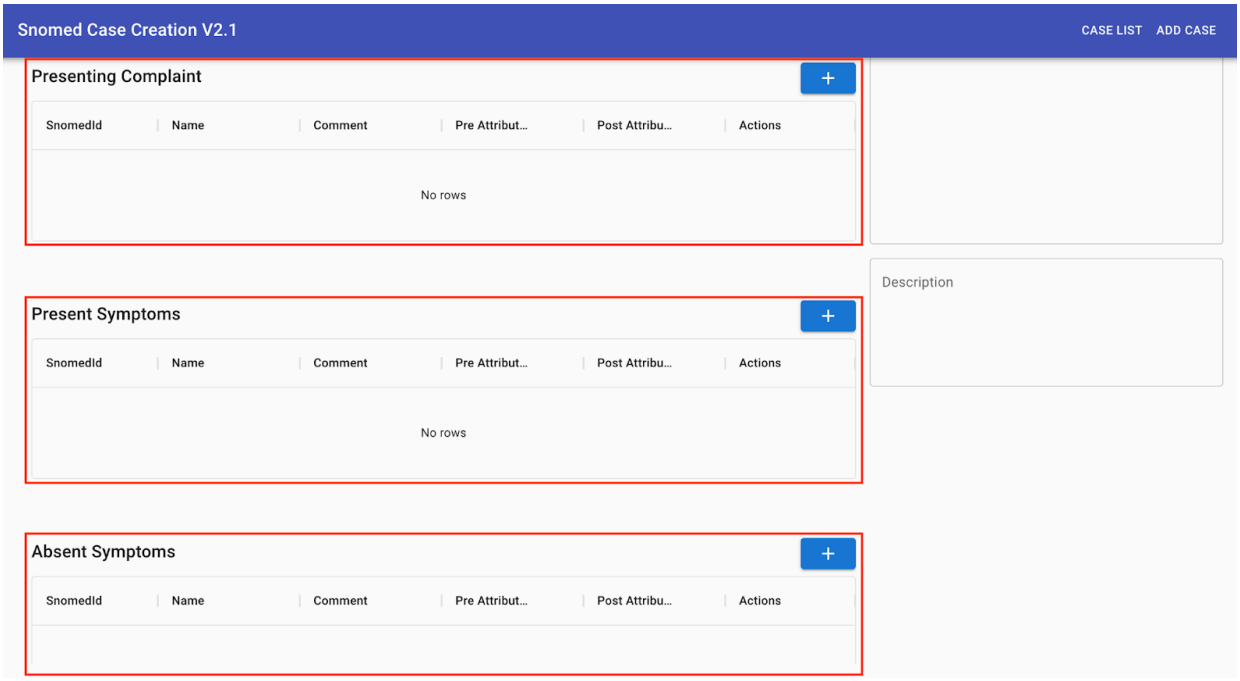


Figure D.4 – At this point, the evidence stored in the "Case description" box is being encoded into corresponding evidence type fields.

e. To encode any of the above-mentioned evidence types a user clicks on the "+" button located in the upper right corner of the respective section (see Figure D.5).

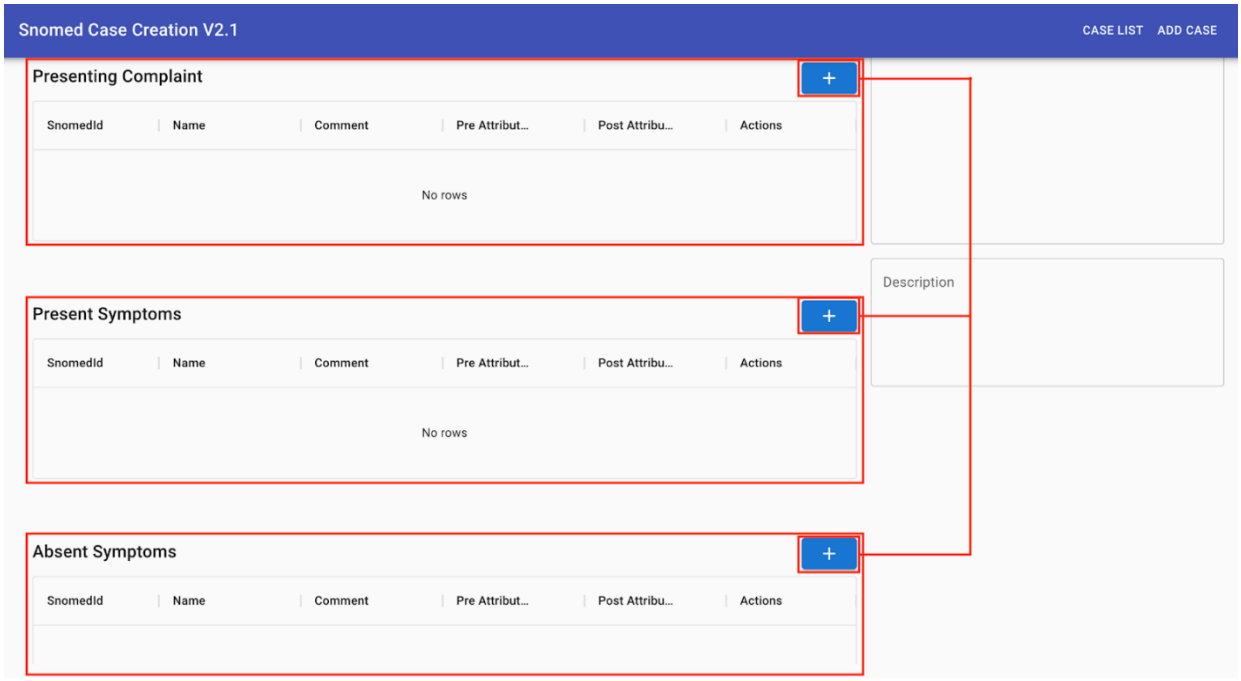


Figure D.5 – The encoding of evidence starts by clicking on the "+" button.

f. Clicking on the "+" button opens the Snomed Concept Browser page (see Figure D.6). This page initially contains:

   i. (A) Symptom search bar and
   ii. (B) Symptom search results box.

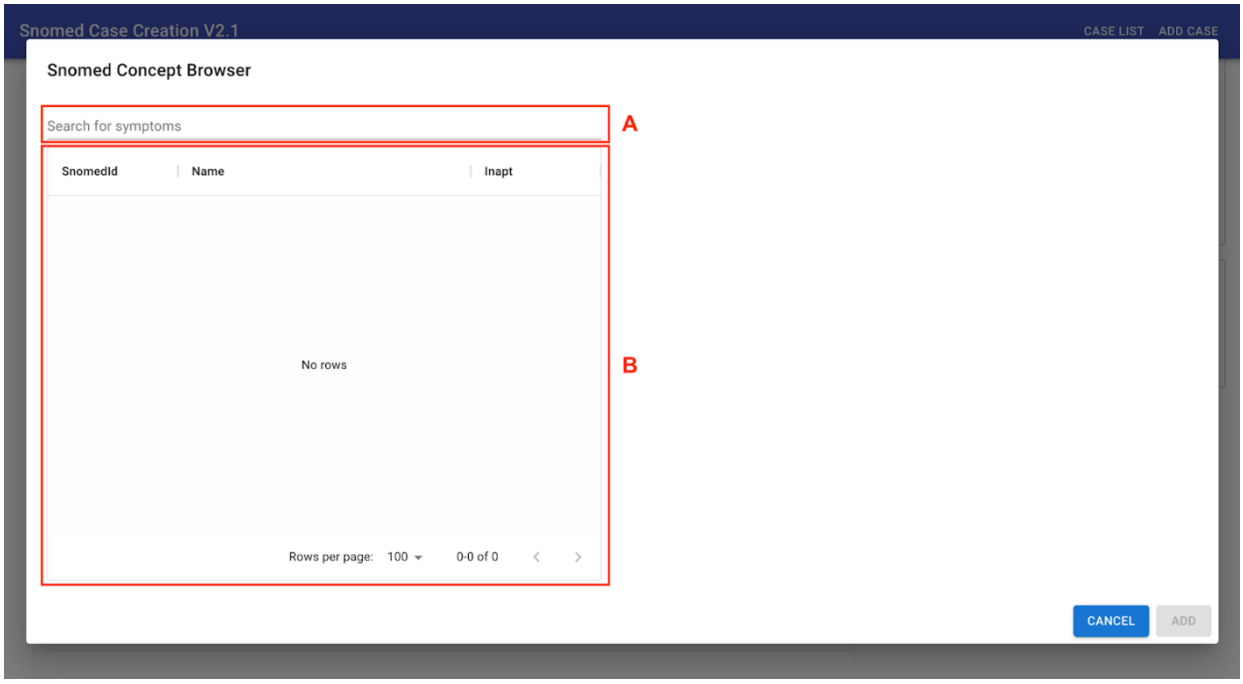


Figure D.6 – Snomed Concept Browser allows a user to quickly find evidence of interest, e.g., fever, abdominal pain, fatigue etc.

g. To start searching for evidence of interest (e.g., headache) a user types in at least three letters of the evidence name (e.g., hea) in the Symptom Search bar. Search results will show up in the Search Results box (see Figure D.7).

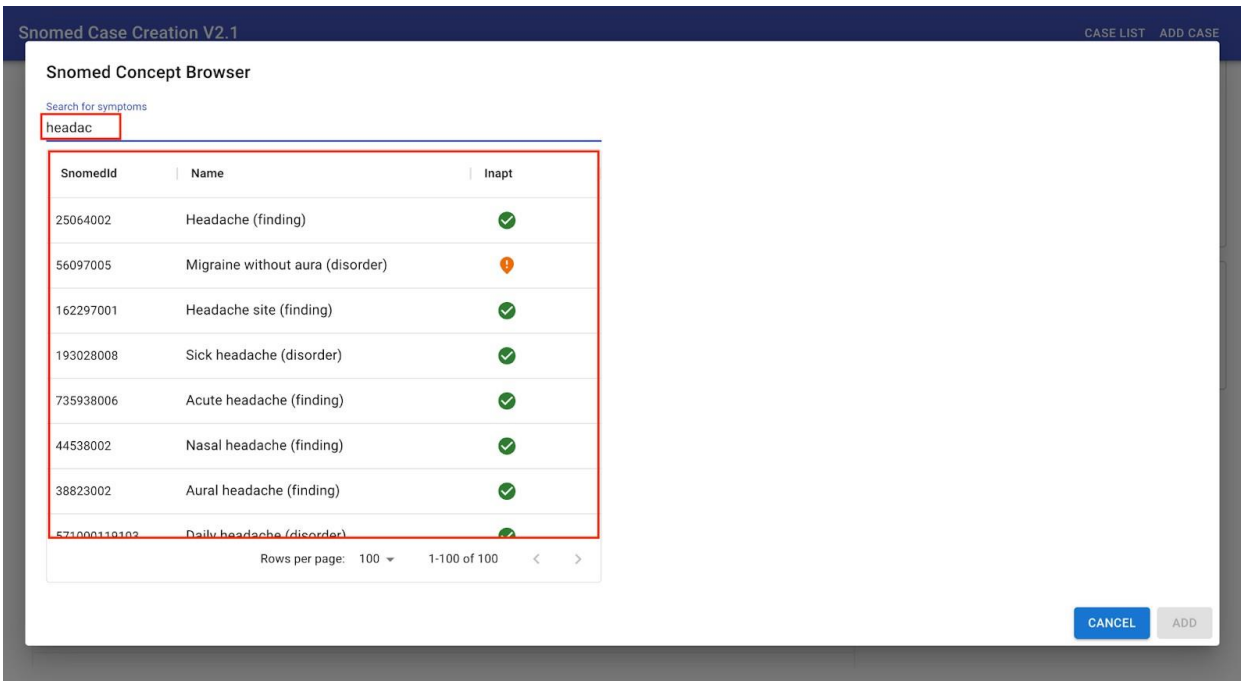


Figure D.7 – SNOMED is a systematically organized computer-processable collection of medical terms used in clinical documentation and reporting.

h. To select the evidence of interest a user clicks on one of the results from the Search Results box. On the right hand side of the SNOMED Concept Browser a user is now able to review the Ancestor and Children concepts of the selected evidence (see Figure D.8).

i. Ancestor or Children concepts that more closely describe the case could be selected directly from the list. If e.g., Acute headache in the list of Children concepts is selected as a better match, the SNOMED Concept Browser will now show the list of Ancestor and Children concepts related to the Acute headache (see Figure D.9).

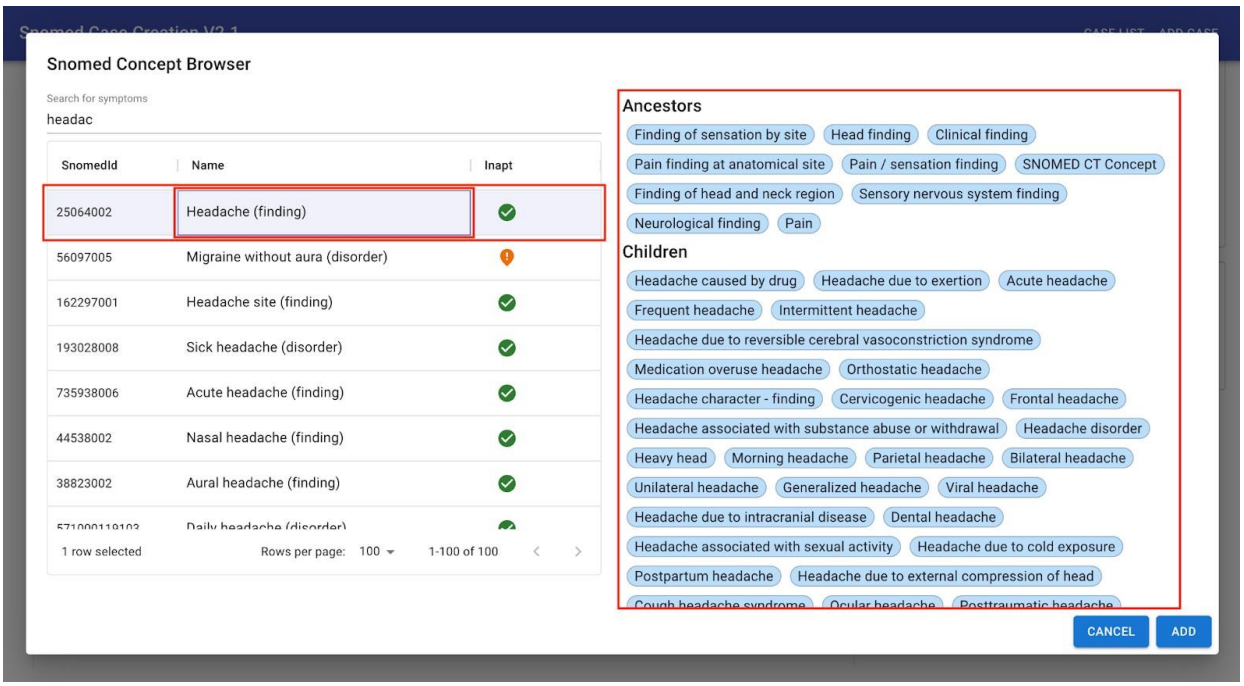


Figure D.8 – User should search for the term that describes the evidence from a case most precisely.

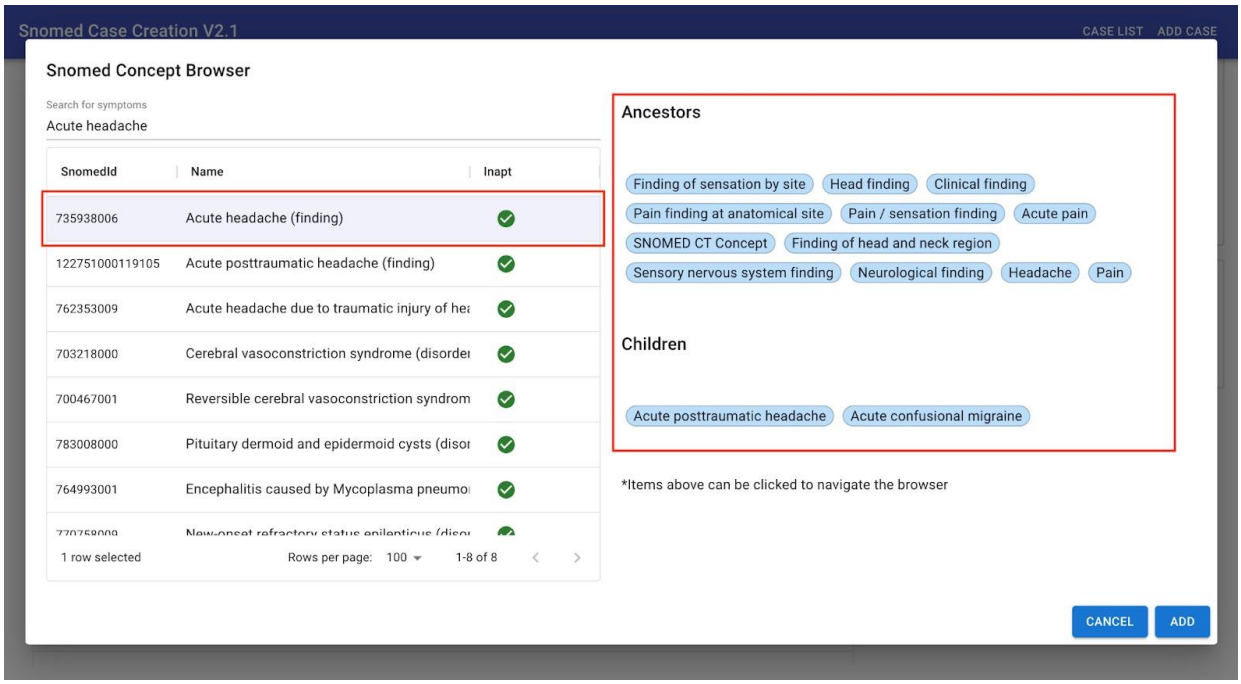


Figure D.9 – In case that the evidence of interest is not precisely defined by the selected search result, a user should try to look for it by reviewing Ancestor and Children concepts related to it.

j. Finally, to add the selected evidence a user clicks the "Add" button located in the lower right corner of the Snomed Concept Browser (see Figure D.10).

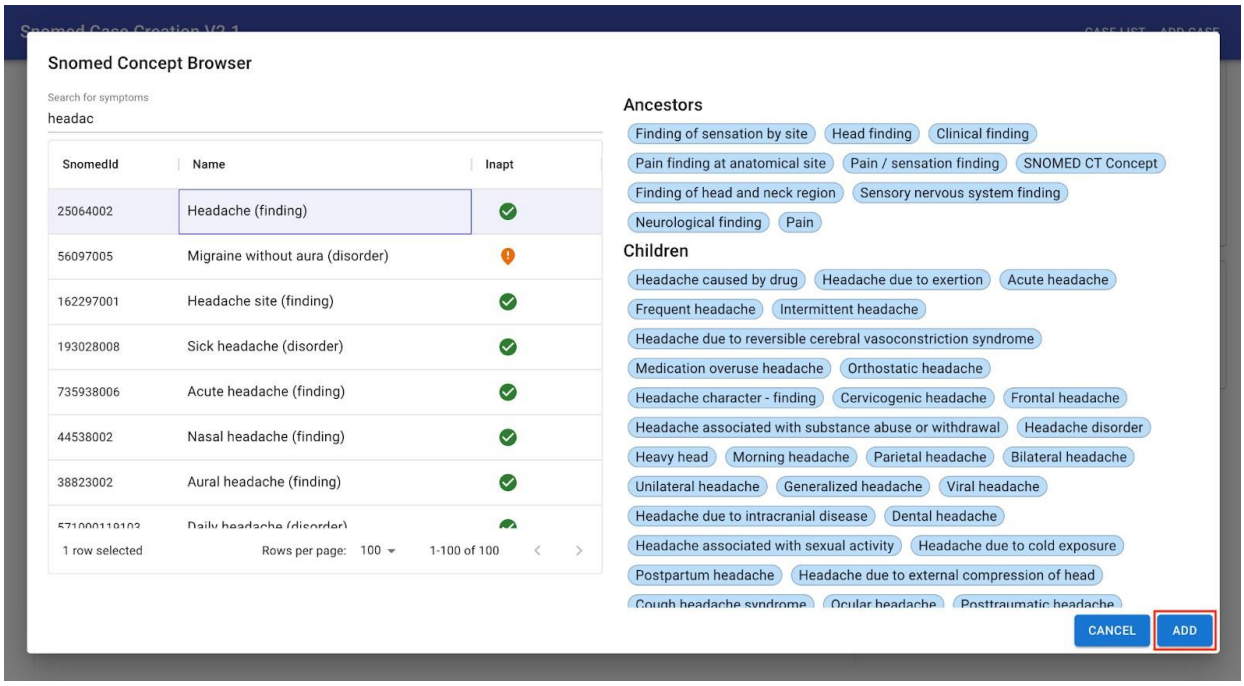


Figure D.10 – Once a user finds an evidence of interest this part of the process should be finalised by clicking on the "Add" button.

k. Selected evidence is now located in the respective evidence type section. To define the evidence in more detail (time since onset, severity, clinical course and finding site) a user clicks the blue pen icon located on the right hand side of the evidence type section (see Figure D.11), which opens the Edit Attributes window (see Figure D.12).

l. User defines the "time since onset" attribute by setting the time since onset unit and time since onset value. To define any of the remaining three attributes (severity, clinical course and finding site) user clicks on the Select State box located on the right hand side of a respective Attribute field (see Figure D.13) and typing in the attribute of interest or simply choosing one from the drop down menu.

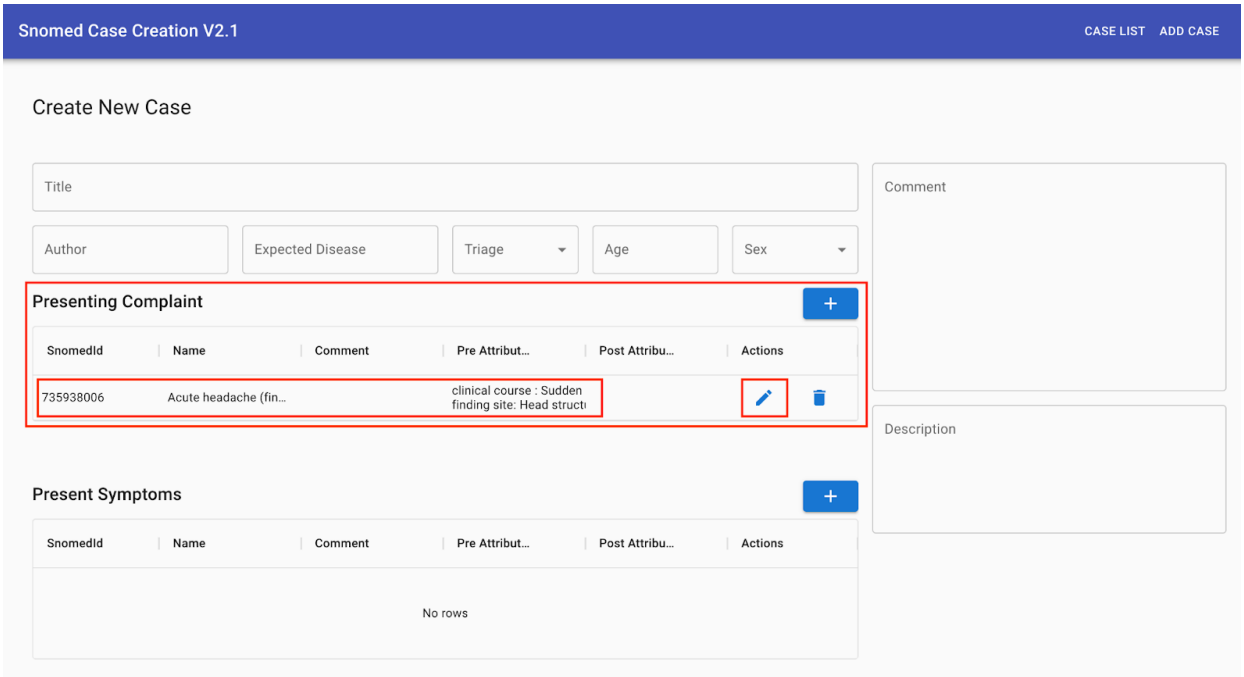


Figure D.11 – To define time since onset, severity, clinical course and/or finding site of an evidence of interest a user clicks the blue pen icon.

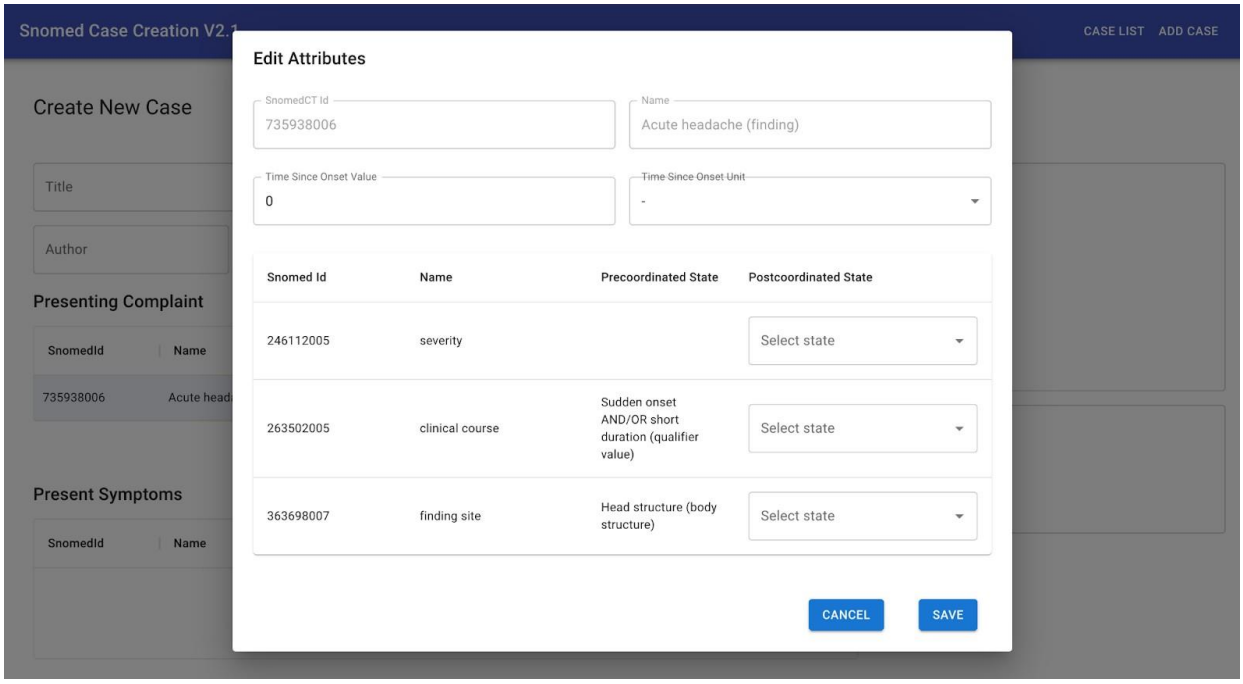


Figure D.12 – Only those fields that will help to define an evidence of interest as precisely as possible should be completed.

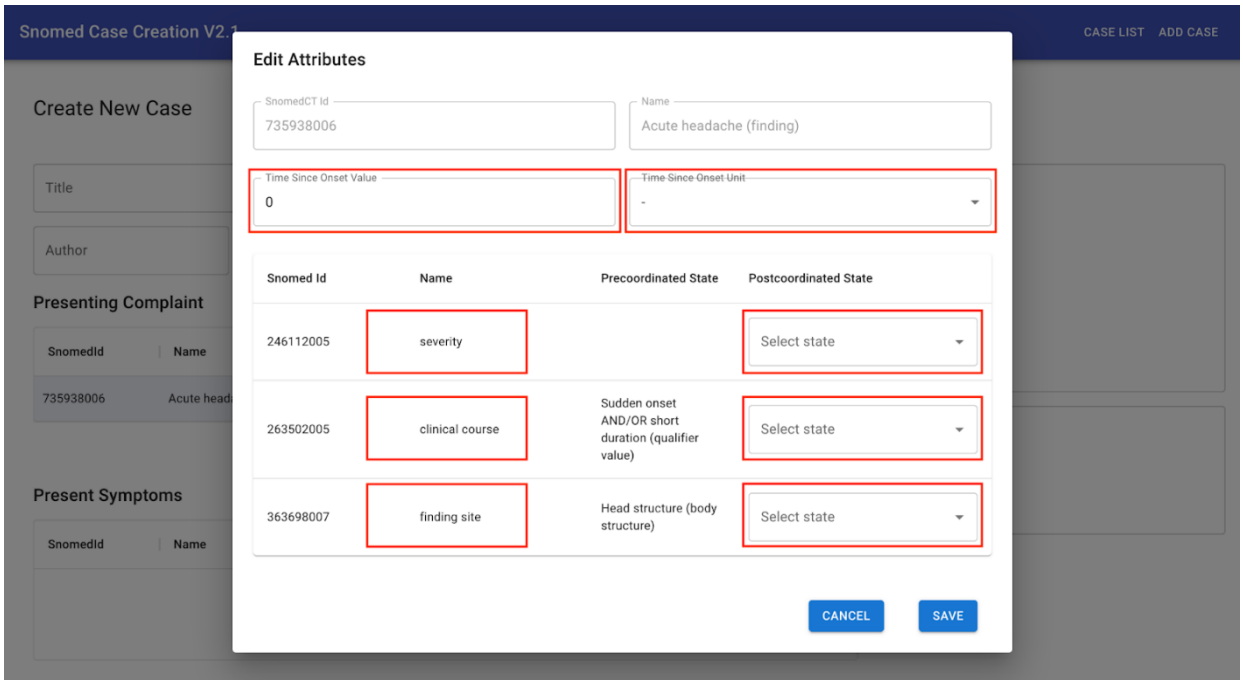


Figure D.13 – User defines attributes by clicking on the Select State box located on the right hand side of a respective Attribute field and by selecting the time since onset unit and time since onset value in case of the "time since onset" attribute.

m. To save the changes a user clicks on the Save button located in the lower right corner of the Edit Attributes window (see Figure D.14). Changes will be shown in the respective evidence type section of the "Create New Case" page (see Figure D.15).

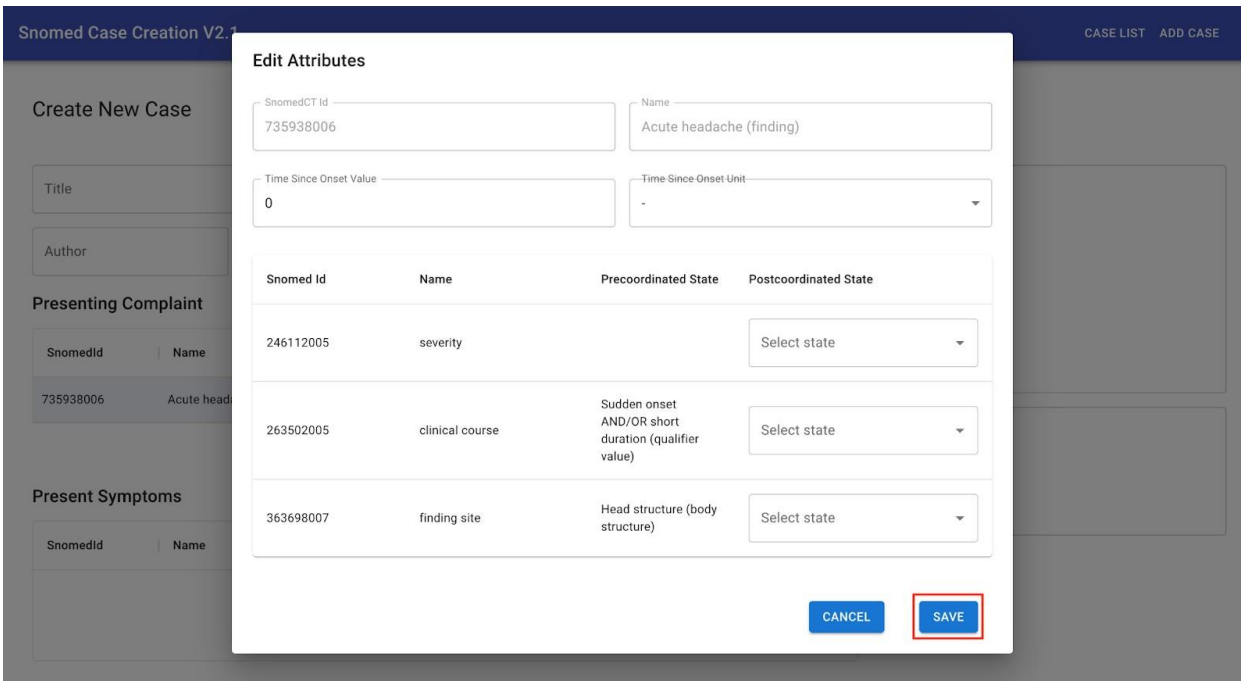


Figure D.14 – Another part of the encoding process could now be finalised by clicking the "Save" button.

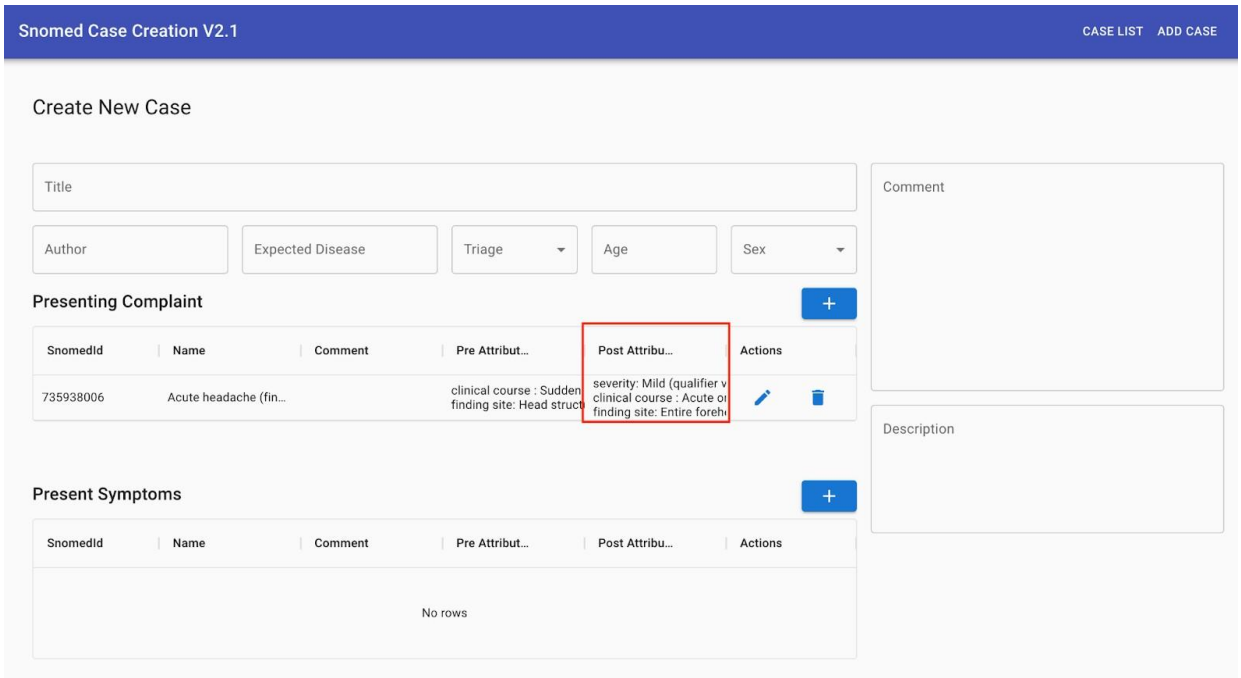


Figure D.15 – A user is now able to see the changes made during the previous step of the process..

n. After entering all relevant information related to the case (e.g., Expected disease, triage, Presenting complaint…) and case creation process (e.g., comments in the Comment section) a user is ready to submit the case by clicking on the "Submit New

Case" button located at the bottom right corner of the "Create New Case" page (see Figure D.16).

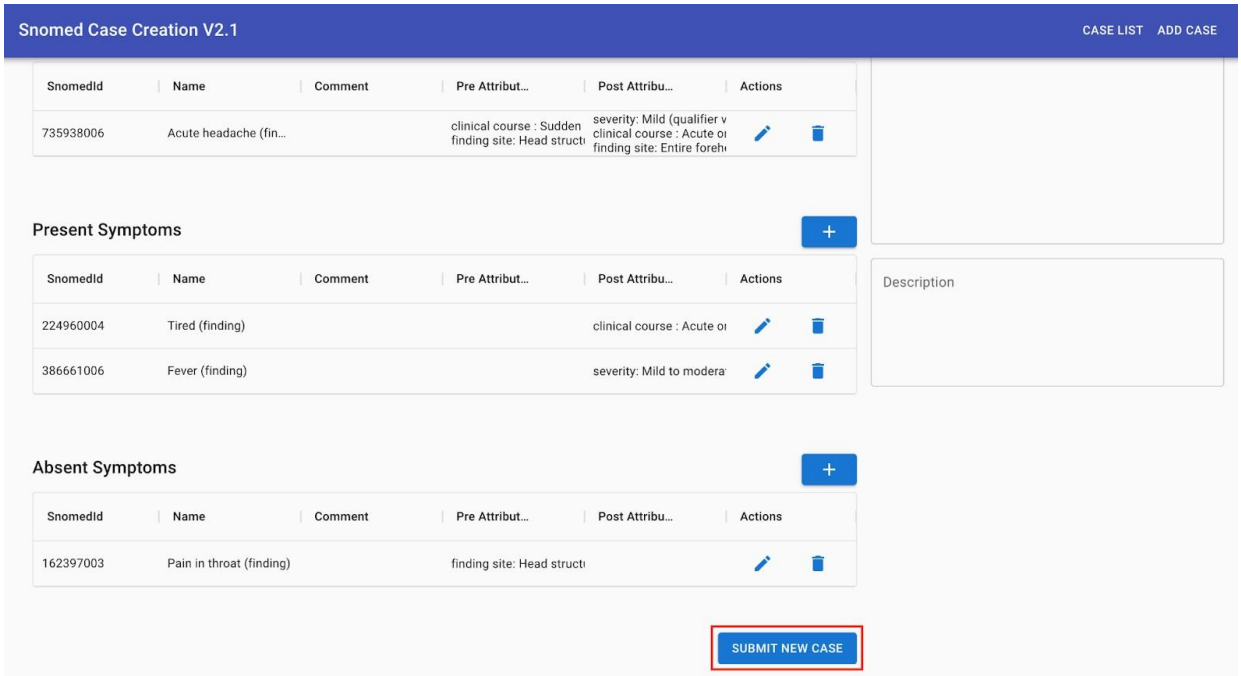


Figure D.16 – Once all the evidence from the "Case description" box is encoded a user is ready to finalise the case creation process by clicking the "Submit new case" button.

Annex E
MMVB 3.x case annotation guideline

ITU/WHO Focus Group AI4H (FG-AI4H)
Topic Group Symptom Assessment (TG-Symptom)

MMVB 3.1 Case Encoding Guideline
2023-06-25 V1.0

Background

This document provides guidance on how to encode test cases for benchmarking of AI-based symptom assessment systems as part of the ITU/WHO AI4H Focus Group, using the MMVB 3.1 Case Encoding Tool.

Prerequisites

For encoding cases you are assumed to have access to the latest version of the *MMVB 3.1 Case Encoding Tool.* The URL to use has been provided to you via email as part of the invite to the case annotation. This annotation guidelines assumes that you have read and understood the *MMVB 3.1 Case Encoding Tool Manual.*

Your annotation task

In the current phase the task is to encode textual case vignettes using the case encoding tool. In the invite mail you have received the ID of clinical vignettes assigned to you to encode. For some annotation sessions clinical vignettes have been already added into the annotation tool so that you only need to open those tasks, rather than creating new tasks (see Figure E.1). They usually contain the name for the annotation session e.g., PCET-2023-02 for the Parallel Case Encoding Test session from February 2023 - followed by the case ID and your names initials.

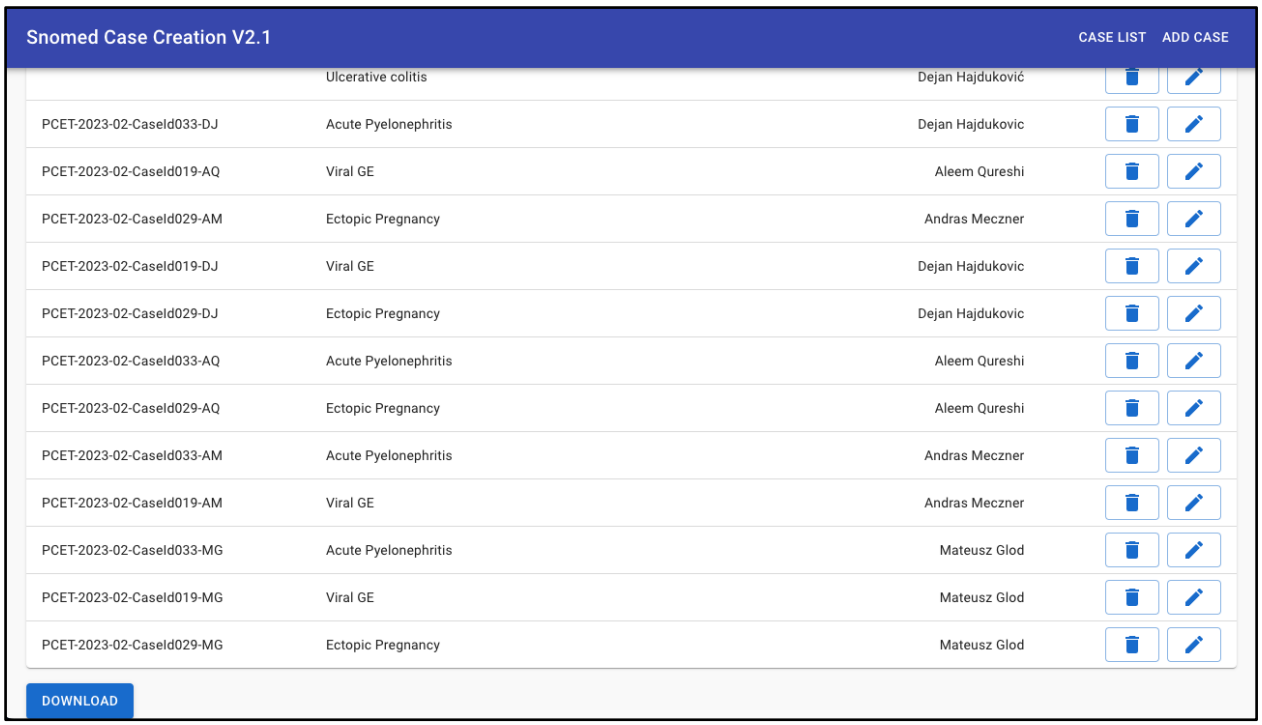


Figure E.1 – Example of a case list showing pre-created annotation task cases

Case Encoding

For each of your case encoding tasks please follow the steps listed below. In case of questions please reach out to the support contact in the invitation mail.

1. To start encoding a case open the Case Encoding Tool and click on the Add Case button.
2. At the beginning of the encoding process add the case description by copying it from a clinical vignette assigned to you and paste it into the "Case Description" input box. This makes it easier to proceed with entering the basic information related to the assigned clinical vignette:
    a. Enter the clinical vignette title;
    b. Enter your name and surname into the ‘Author’ input field;
    c. Enter the condition this case is encoding for into ‘Expected Disease’ input field;
    d. Enter the triage level assigned to the case by adding it into the "Triage" input field;
    e. Enter the age and sex details of the case profile into ‘Age’ and ‘Sex’ input fields.
3. In the ‘Comment’ input box:
    a. Input the time at which you started the process. At the end of the case, you will refer back to it to estimate and note down in the same box the overall duration of the encoding process of a single case.
    b. Start to keep notes of any thoughts or ideas coming up while encoding the case. This includes difficulties/frustrations with the process, symptoms you were unable to add, uncertainties about the process, aspects of the encoding process you found easy or helpful or useful or anything else you feel worth noting.
4. Continue with the encoding process by entering evidence from the assigned clinical vignette: presenting complaint(s), present symptoms and absent symptoms:
    a. To add a presenting complaint click on the plus (+) button at the right hand side of the Presenting Complaint input box;

b. Start typing in the presenting complaint finding/feature in the "Search for symptoms" search bar;

   i. Symptom search results will appear below the search bar;

   ii. Select the finding that best describes the presenting complaint;

   iii. On the right hand side of the symptom search bar are Ancestors/Children concepts related to the symptom search results - use them to navigate the symptom/feature browser in case symptom search results does not contain the finding/feature of interest;

   iv. To add the symptom click on the Add button

   v. At this point a user is able to define the Presenting complaint in more detail (time since onset, severity, clinical course and finding site):

      1. Click the blue pen icon located on the right hand side of the evidence type section (see Figure D.11) to open the Edit Attributes window (see Figure D.12).

      2. To define any of the three attributes (severity, clinical course and finding site) click on the Select State box located on the right hand side of a respective Attribute field (see Figure D.13) and type in the attribute of interest or simply choose one from the drop down menu.

      3. To define the "time since onset" attribute click on the dropdown menu of the Time Since Onset Unit box to select the time since onset unit (days, weeks, months or years) and then enter the value in the Time Since Onset Value box.

      4. Save changes by clicking on the "Save" button located in the lower right corner of the "Edit Attributes" window (see Figure D.14). Changes will be shown in the respective evidence type section of the "Create New Case" page (see Figure D.15).

c. Repeat previous two steps (a and b) for other two evidence types: Present symptoms and Absent symptoms.

d. Return to the "Comment" input box to note down:

   i. The overall duration of the encoding process of a single case - refer to the start time you previously input when you commenced.

   ii. Any further thoughts or ideas coming up while encoding the case. This includes difficulties/frustrations with the process, symptoms you were unable to add, uncertainties about the process, aspects of the encoding process you found easy or helpful or useful or anything else you feel worth noting.

e. Once all evidence has been noted and described in detail a user is ready to submit the case. Click on the "Submit New Case" button located at the bottom right corner of the "Create New Case" page (see Figure D.16) to finalise the case encoding process.