



# BSA WEB SDK WALKTHROUGH

---

Presented by: Wafiuddin Ali, Senior Systems Engineer, FNSValue Malaysia



# Table of CONTENTS

---

01

BSA Technology  
Overview

02

Introduction to  
Web SDK

03

SDK Components  
and Functionalities

04

SDK  
Implementation

05

Testing SDK  
Integration and  
troubleshooting

06

Resources

07

Demo App

08

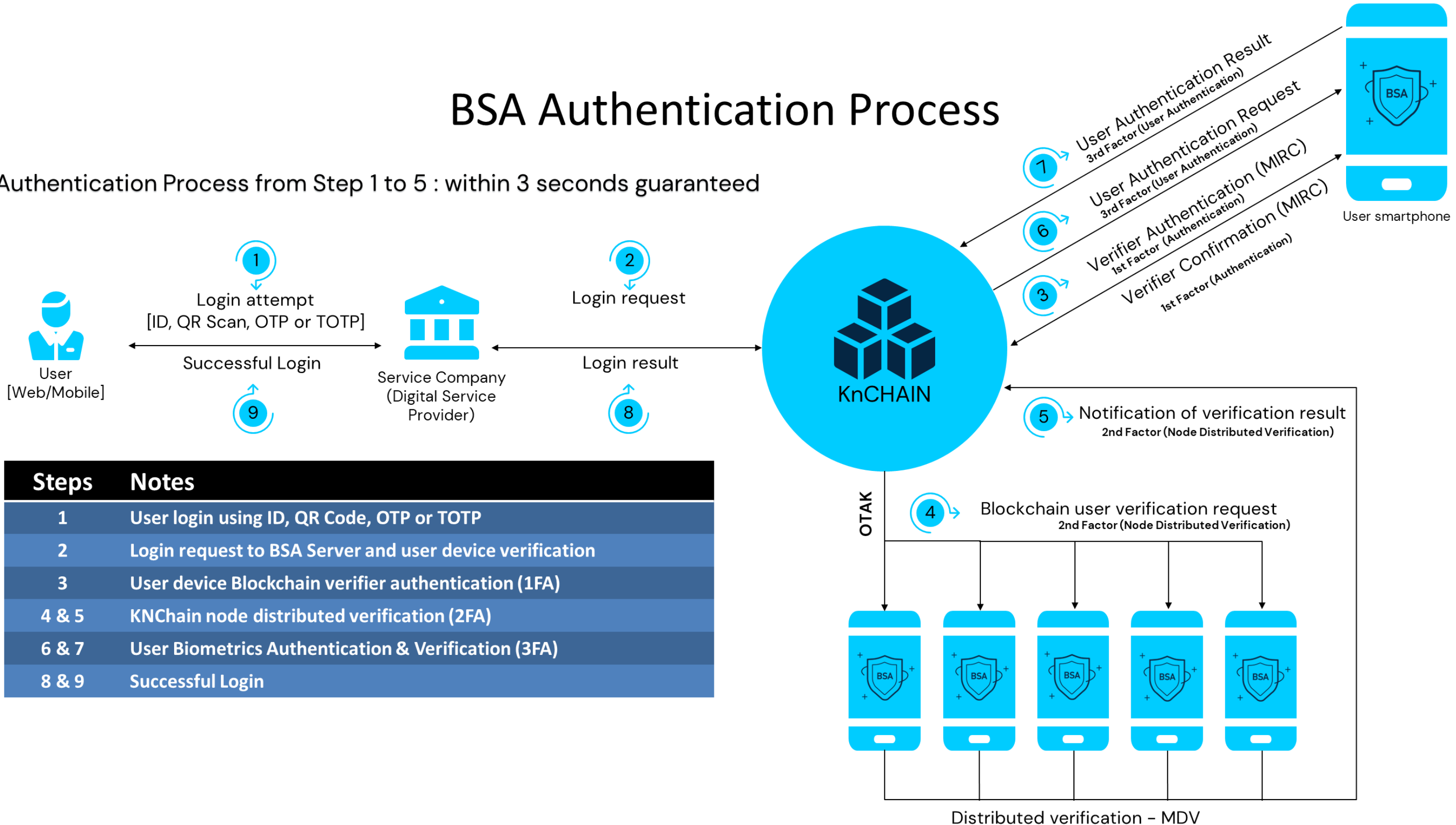
Questions &  
Answer



# BSA Authentication Process

## BSA Authentication Process

Authentication Process from Step 1 to 5 : within 3 seconds guaranteed



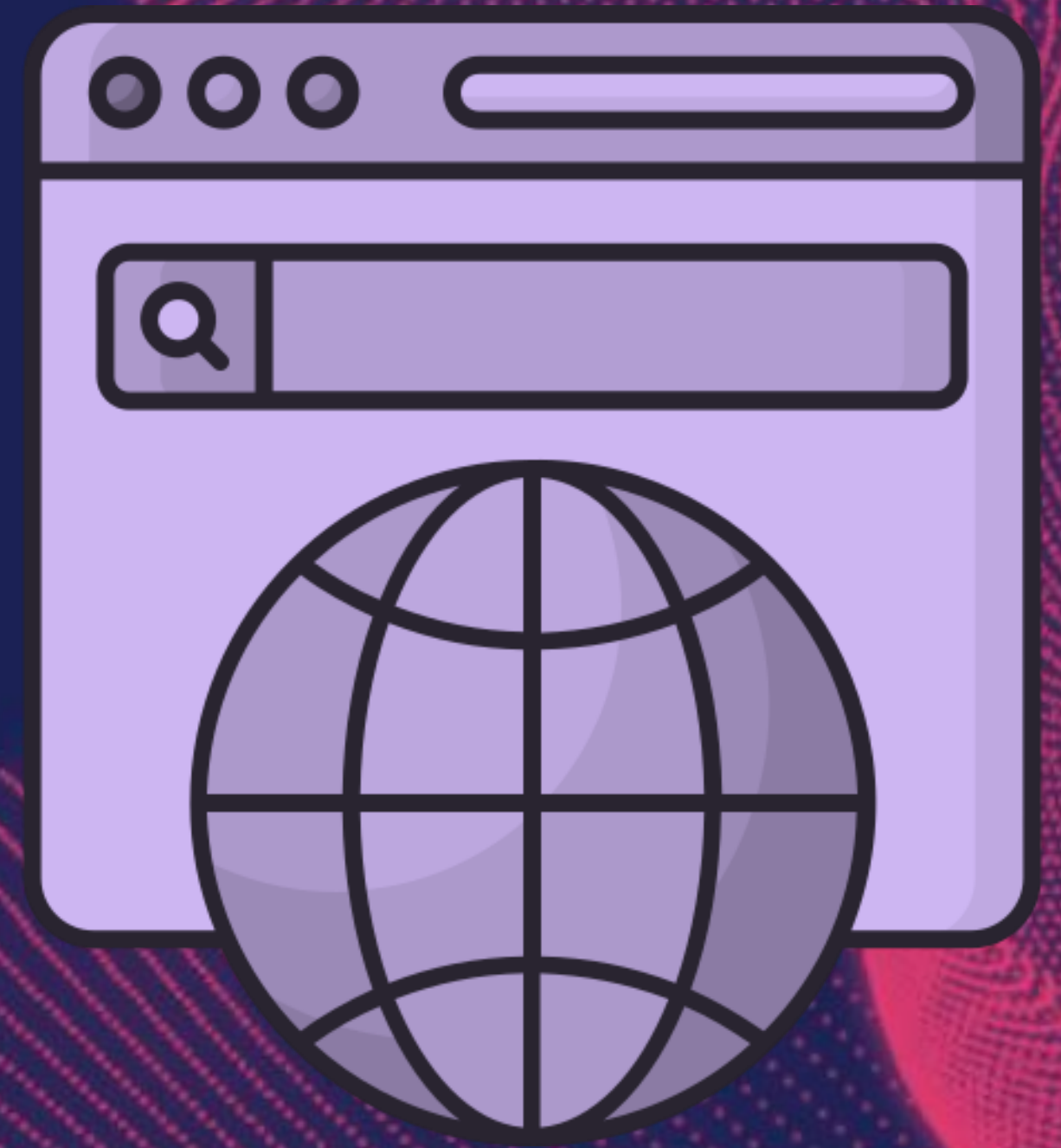


# Introduction

# WEB SDK

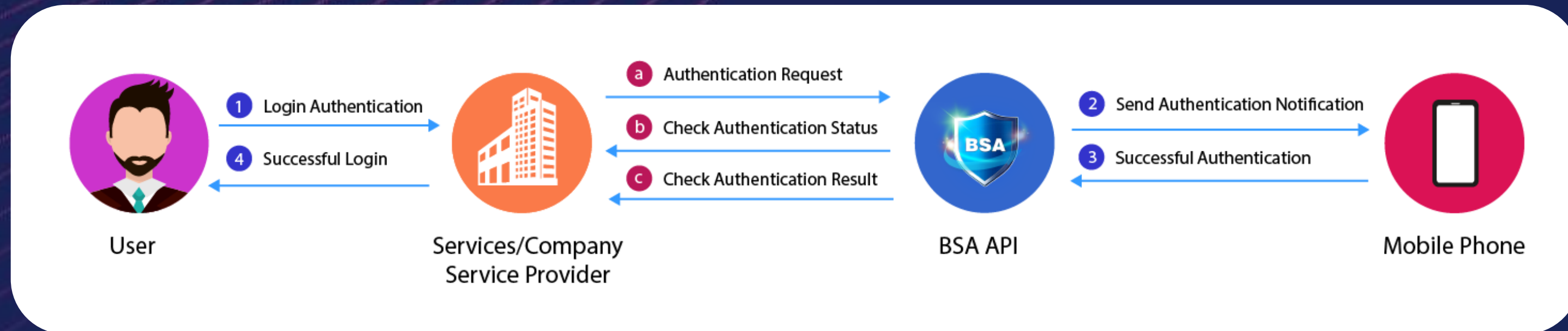
---

- Web SDK (Software Development Kit) is a collection of tools, libraries, and documentation that enables developers to create applications or integrate specific functionalities into existing web applications.
- Web SDKs are typically designed to simplify the development process by providing pre-built components, APIs (Application Programming Interfaces), and resources that developers can leverage to add complex features to their web projects without having to build them from scratch.
- Blockchain Secure Authentication (BSA) Software Development Kit (SDK) consists of BSA-JS which enables custom configuration of the login component that can configure default authentication page.





# SDK COMPONENTS & FUNCTIONALITIES



The diagram above illustrates the comprehensive process of authentication components and functionalities required for the login authentication process using the BSA SDK.

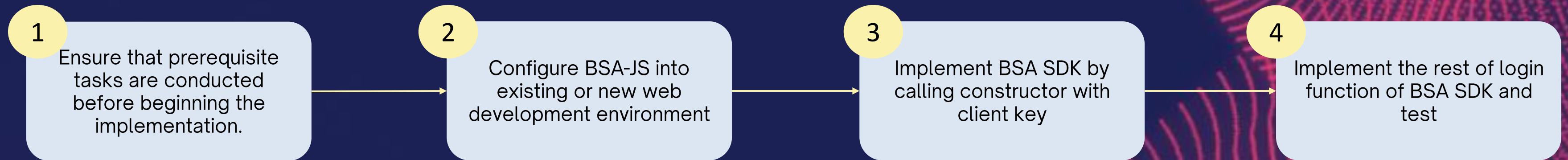
The forthcoming slide will meticulously outline the diverse authentication methods offered within the BSA SDK, elucidating their individual functionalities.

# SDK COMPONENTS & FUNCTIONALITIES

| Name                | Description   |
|---------------------|---|
| constructor         | constructor of the BSA-JS   |
| requestAuth         | Handles the BSA authentication request, and redirect to the setup URL after authentication                              |
| requestAuthCallback | Handles the BSA authentication request, and returns the result to the setup callback function after authentication      |
| onCancel            | Cancels the BSA authentication request  |
| setAuthTimer        | Provides remaining time for authentication  |
| setAuthMessage      | Provides authentication process message   |
| requestQr           | Handles the BSA QR authentication request, and redirect to the setup URL after authentication                           |
| requestQrCallback   | Handles the BSA QR authentication request, and returns the result to the setup callback function after authentication   |
| onQrCancel          | Cancels the BSA QR authentication request   |
| setQrTimer          | Provides remaining time for QR authentication   |
| setQrMessage        | Provides QR authentication process message  |
| requestOtp          | Handles the BSA OTP authentication request, and redirect to the setup URL after authentication                          |
| requestOtpCallback  | Handles the BSA OTP authentication request, and returns the result to the setup callback function after authentication  |
| onOtpCancel         | Cancels the BSA OTP authentication request  |
| setOtpTimer         | Provides remaining time for OTP authentication  |
| setOtpMessage       | Provides OTP authentication process message   |
| requestTotpCallback | Handles the BSA TOTP authentication request, and returns the result to the setup callback function after authentication |



# SDK Implementation Process Overview



This process flow illustrates the high-level implementation of the BSA SDK. Further detailed implementation will be presented in subsequent slides.



# SDK Implementation

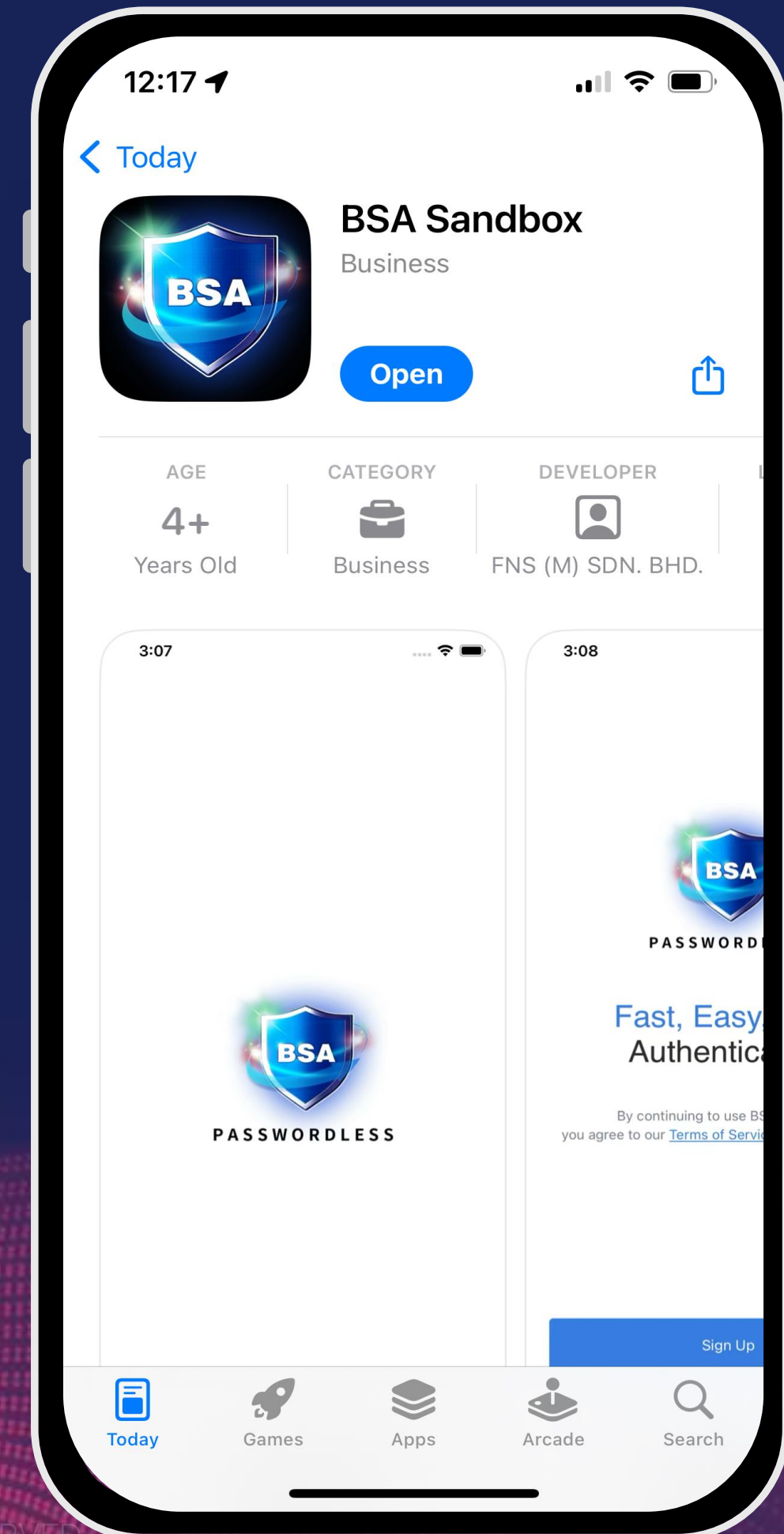
## PRE-REQUISITE

### A. Key Issuance

1. Client Key will be provided and to be configured in the development

### B. Authentication Test: If the actual user wishes to activate the BSA authentication, pre- registration process is required as described below:

1. Download the BSA Sandbox mobile app and sign in
  - i. Search for the 'BSA Sandbox' or 'fnsvalue' at the Play Store/App Store and download the app
  - ii. In case of new user, sign in is required
2. Link to the client website
  - i. Go through Menu > My information > Trusted Website
  - ii. Search for the client websites' name
  - iii. Click the [Link] button to connect with the client site
3. Now the user can authenticate with the BSA information







## SDK Implementation

# LIMITATIONS

---

BSA SDK, designated as BSA-JS, is optimized for JavaScript environments, implying potential constraints outside of this framework.

To ensure seamless functionality and minimize limitations, it's preferable to integrate the SDK within a JavaScript ecosystem.

For optimal performance, leveraging a JavaScript environment is strongly recommended.



# SDK Implementation INSTALLATION

**Option 1:** Latest version of BSA-JS declaration

```
<script type="text/javascript" src="https://resourceaz.fnsbsa.com/bsa-sandbox/resources/bsa-web-sdk/bsa-web-sdk.js"></script>
```

**Option 2:** Specify version of BSA-JS declaration

```
<script type="text/javascript" src="https://resourceaz.fnsbsa.com/bsa-sandbox/resources/bsa-web-sdk/versions/{version_num}/bsa-web-sdk-{version_num}.js"></script>
```

Note: You may choose either option 1 or option 2 to declare BSA-JS. To know which versions are available, visit <https://developers.fnsvalue.co.kr/docs/web/guardianJS/more>



# SDK Implementation

## DECLARATION - Constructor

### Constructor

A constructor is an **essential element** for the BSA-JS. The client key which is necessary to utilize the BSA-JS, can be confirmed by inquiring the person in charge or directly at the BSA Portal.

```
constructor(clientKey)
```

### Parameter

| Name      | Type   | Description                         |
|-----------|--------|-------------------------------------|
| clientKey | String | Client key generated to utilize BSA |

### Example

```
const bsa = new BSA("{Client Key}");
```



# SDK Implementation

## DECLARATION – User ID Authentication

### Authentication Request

When the authentication is requested, the API call will be made with `requestAuthCallback()`. Push notification will be sent to the app, and the result will be returned through `onSuccess` if successfully authenticated.

```
requestAuthCallback(userKey, successCallback, errorCallback)
```

### Parameter

| Name    | Type   | Description      |
|---------|--------|------------------|
| userKey | String | BSA user account |

### Example

```
const bsa = new BSA("{Client Key}"); bsa.requestAuthCallback(userKey,
(data) => {
  console.log('onSuccess');
  console.log('data : ', data);
}, (errorCode, errorMsg) => {
  console.log('onError');
  console.log('errorCode : ',
errorCode);
  console.log('errorCode : ',
errorMsg);
});
```



# Testing SDK & TROUBLESHOOTING

01

## BSA App Authentication

If the integration is successful, you shall be able to receive authentication request notification for you to approve in your BSA mobile app when testing a login request.

02

## Print to console

You may print to console at every method when you call. This is way to debug and understand behavior of the code.

03

## Return Code

Successful integration will return code '0'. Other than that, will be considered as unsuccessful and error. You may refer to the error code provided.

04

## Network Inspector

Checking network inspector is important to check request header is sending correctly, response header is providing correct metadata with response and check response body, to ensure it contains expected data.



# RESOURCES

---



## BSA Manuals

This document contains on how to onboard with BSA mobile app and how to test it



## Web SDK Documentation

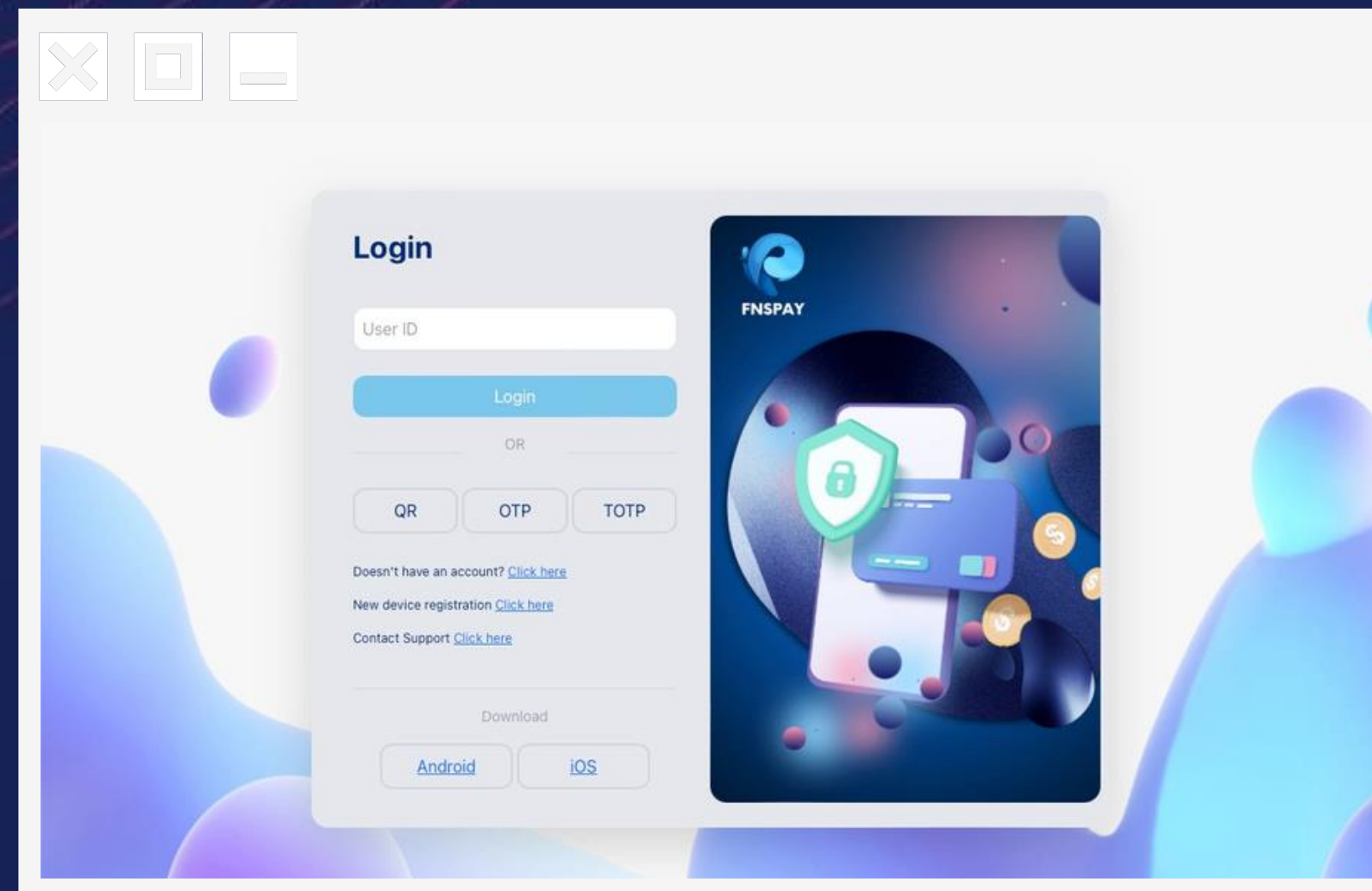
This web SDK document contains overall implementation BSA-JS method with error code



FNSPay Demo



FNSPay is a e-wallet demo portal which uses FNSPay mobile app for login authentication. You may test it and understand how and what method of BSA-JS SDK is suitable to apply for each process







# THANK YOU !

It's game time! Time to unleash your creativity, conquer those coding challenges, and make your dreams a reality!