

**Thank you for your purchase! To ensure seamless after-sales support, please activate your product using the link below:**

[Activate Now](#)

Your activation is essential for us to provide you with the best assistance. We appreciate your cooperation.

## Technology Used

- **ReactJS** with Typescript for Admin Panel
- **NodeJS** with Express Framework for API
- **MySQL** for Database

Please follow the below steps to set up the project on your server.

### 1. Setup the Database (Technology MySQL)

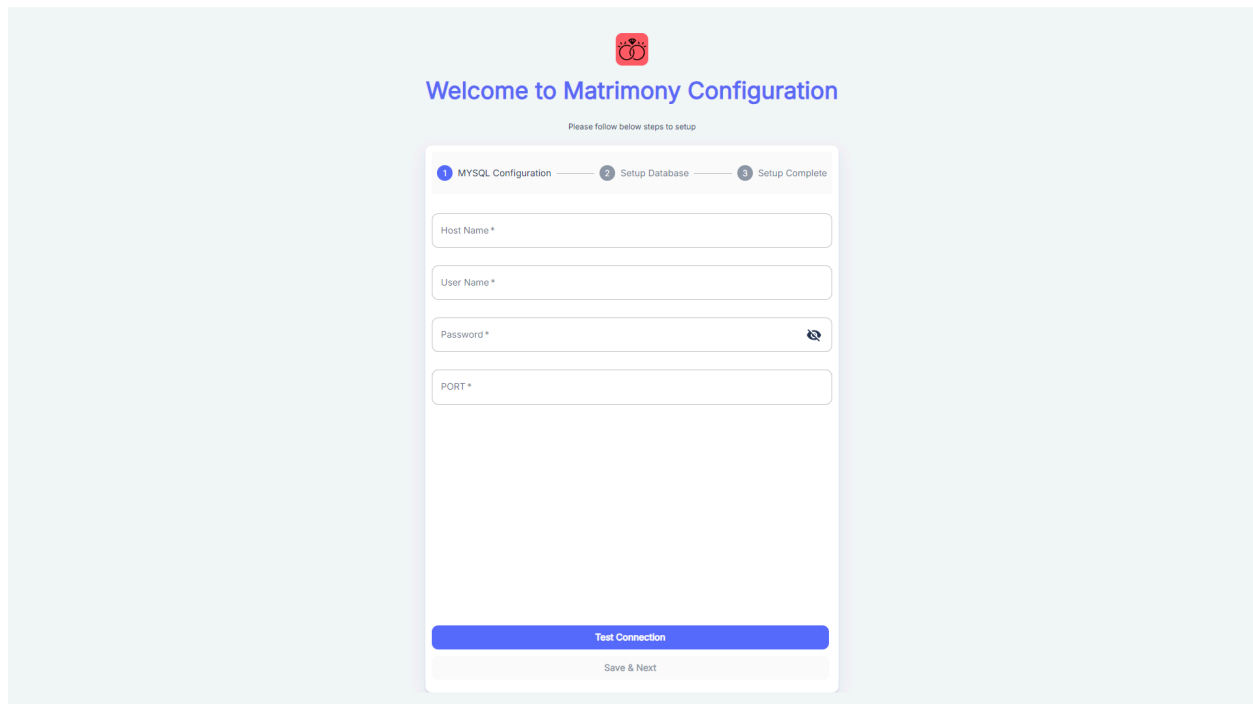
Database configuration could be set up through the admin panel.

#### Step 1: Connect MYSQL Server

For Accessing Admin use url

`https://<YOUR_DOMAIN_NAME>/admin`

- Add Database Host Name, Database User Name, Database Password, Database Port to connect with database.



Welcome to Matrimony Configuration

Please follow below steps to setup

1 MYSQL Configuration — 2 Setup Database — 3 Setup Complete

Host Name \*

User Name \*

Password \*

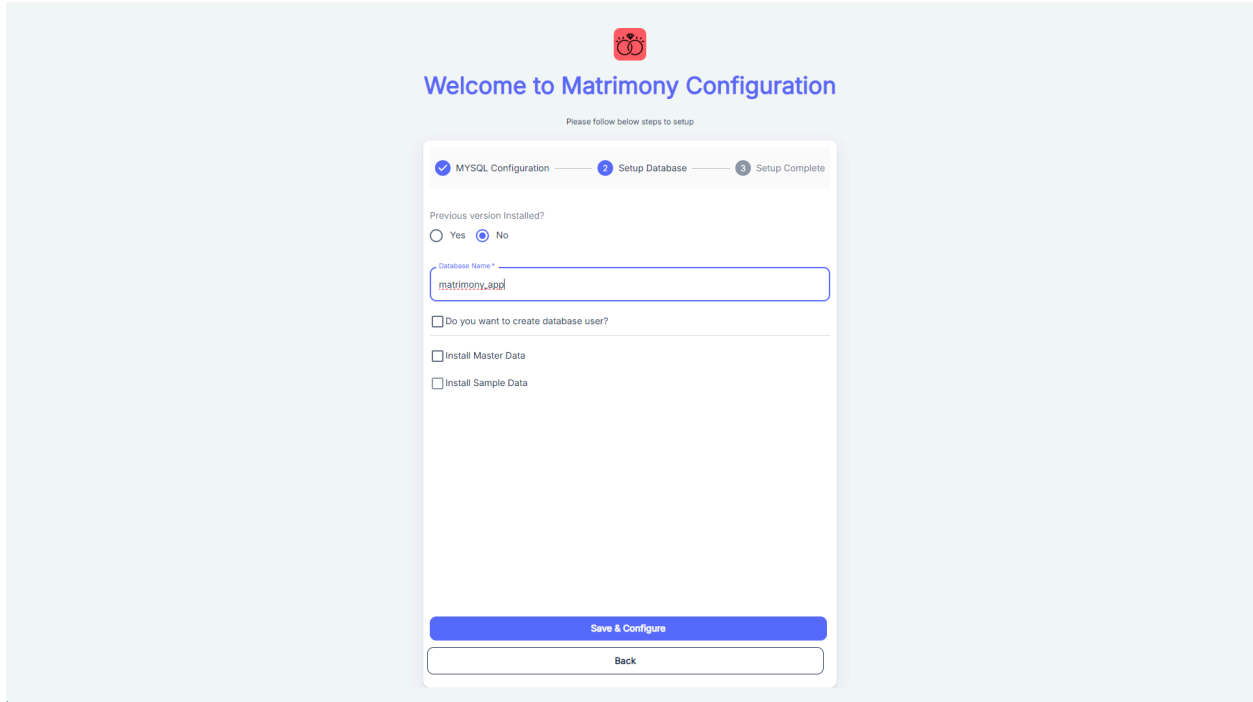
PORT \*

Test Connection

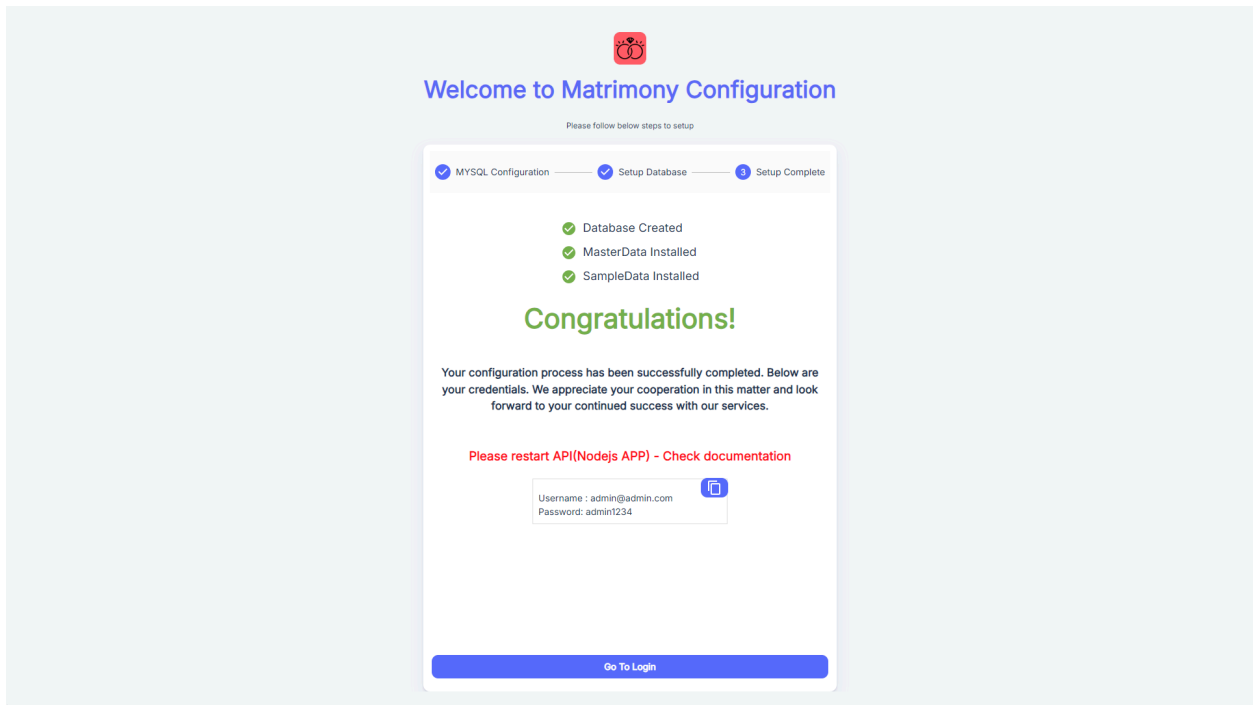
Save & Next

## Step 2: Setup Database

- Add Database Name or select existing database if you already have the previous version.
- Select the checkbox if you wish to create a new database user..
- Select the checkbox if you wish to install masterData.
- Select the checkbox if you wish to install sampleData.



### 3. Success



**Note** :- Please ensure to restart the API server after completing the database configuration step in the admin panel before proceeding with the login process.

## 2. Setup the API, Admin Panel, Website (Technology NodeJS,ReactJS)

Update MatrimonyCMS/matrimony-firebase-adminsdk.json file

```
{
  "type": "<type>",
  "project_id": "<project_id>",
  "private_key_id": "<private_key_id>",
  "private_key": "<private_key>",
  "client_email": "<client_email>",
  "client_id": "<client_id>",
  "auth_uri": "<auth_uri>",
  "token_uri": "<token_uri>",
  "auth_provider_x509_cert_url": "<auth_provider_x509_cert_url>",
  "client_x509_cert_url": "<client_x509_cert_url>",
  "universe_domain": "<universe_domain>"
}
```

- Open the folder and upload it on your hosting server.
- Execute npm i command in server Terminal.
- For Accessing API use url

**https://<YOUR\_DOMAIN\_NAME>/api**

- For Accessing Admin use url

**https://<YOUR\_DOMAIN\_NAME>/admin**

- For Accessing API use url

`https://<YOUR_DOMAIN_NAME>`

## A. Setup Web App (Technology Flutter)

1. Open the  
**MatrimonyCMS\build\Website\assets\assets\jsondata\variable.js**  
**on** file in the editor Replace with your own parameters.

```
assets > assets > jsondata > {} variable.json > ...
1  {
2    "appName": "<YOUR-APP-NAME>",
3    "apiBaseURL": "<YOUR-API-BASEURL>",
4    "imageBaseurl": "<YOUR-API-IMAGEBASEURL>",
5    "version": 1.0,
6    "apiKeyForWeb": "Your key",
7    "authDomainForWeb": "*Your Firebase Porject Id*.firebaseapp.com",
8    "projectIdForWeb": "Your Firebase Porject Id",
9    "storageBucketForWeb": "*Your Firebase Porject Id*.appspot.com",
10   "messagingSenderIdForWeb": "Your message Sender Id",
11   "appIdForWeb": "Your Firebase App Id",
12   "measurementIdForWeb": "Your measurementId",
13   "vapidKey": "Your vapidKey"
14 }
```

2. Change Following Firebase keys In Following Path
  - i. PATH : Build/Website/firebase-messaging-sw.js
  - ii. Changeable Keys :

```
firebase.initializeApp({
  apiKey: "Your key",
  authDomain: "*Your Firebase Porject Id*.firebaseapp.com",
  projectId: "Your Firebase Porject Id",
  storageBucket: "*Your Firebase Porject Id*.appspot.com",
  messagingSenderId: "Your message Sender Id",
  appId: "Your Firebase App Id",
  measurementId: "Your measurementId",
});
```

### 3. Change Web Name & Firebase Parameters In Following Path

- i. Path: Build/Website/index.html
- ii. Change Your App Name

```
<!-- iOS meta tags & icons -->
<meta name="apple-mobile-web-app-capable" content="yes">
<meta name="apple-mobile-web-app-status-bar-style" content="black">
<meta name="apple-mobile-web-app-title" content="Matrimony">
<link rel="apple-touch-icon" href="icons/Icon-192.png">

<!-- Favicon -->
<link rel="icon" type="image/png" href="favicon.png" />

<title>Matrimony</title>
<link rel="manifest" href="manifest.json">
```

- iii. Change Your Firebase Parameters

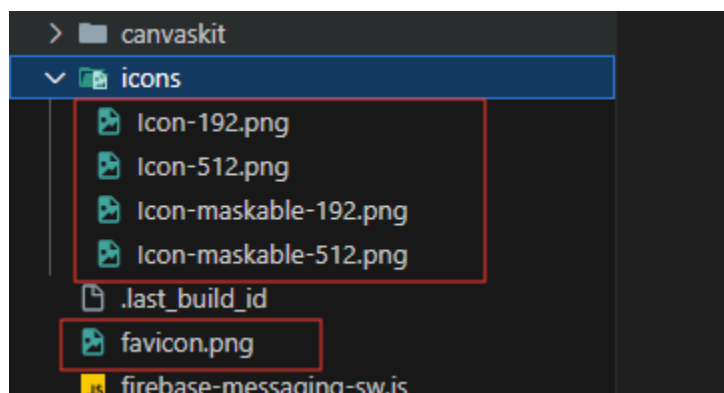
```
<script>
const firebaseConfig = {
  apiKey: "Your key",
  authDomain: "*Your Firebase Porject Id*.firebaseapp.com",
  projectId: "Your Firebase Porject Id",
  storageBucket: "*Your Firebase Porject Id*.appspot.com",
  messagingSenderId:"Your message Sender Id",
  appId: "Your Firebase App Id",
  measurementId: "Your measurementId",
};
// Initialize Firebase
firebase.initializeApp(firebaseConfig);
</script>
```

#### 4. Change Website Icon:

- i. Generate Icon From this URL:

<https://favicon.io/favicon-converter/>

- ii. Replace the icons in the Build\Website\icons folder as shown in the below image.



#### 5. Upload the **Builds/Website** folder to your desired location

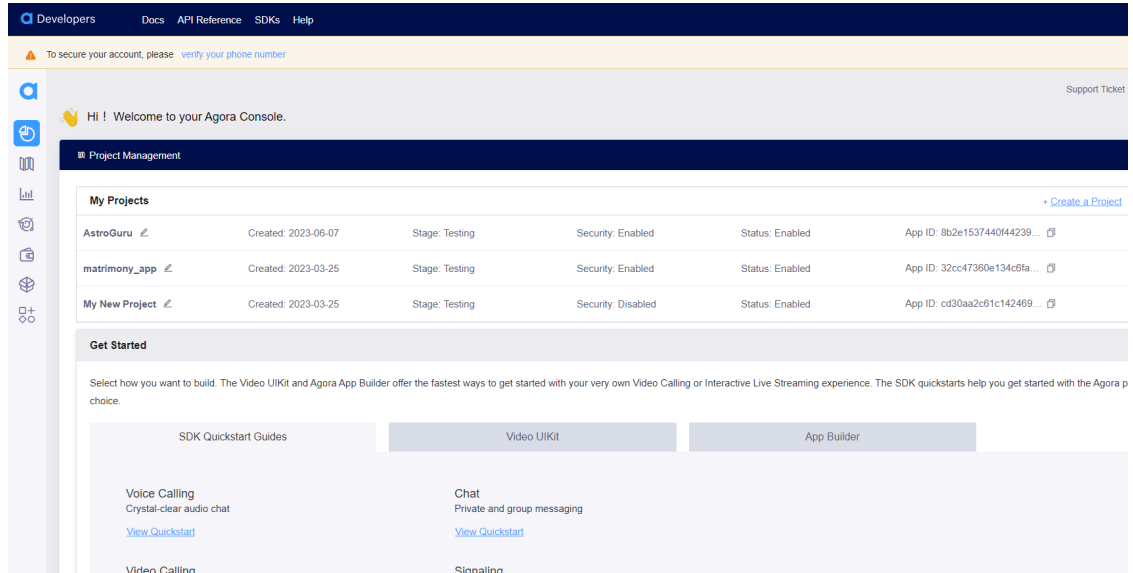
- i.

## b. Third-Party Integration

### c. Setup Agora

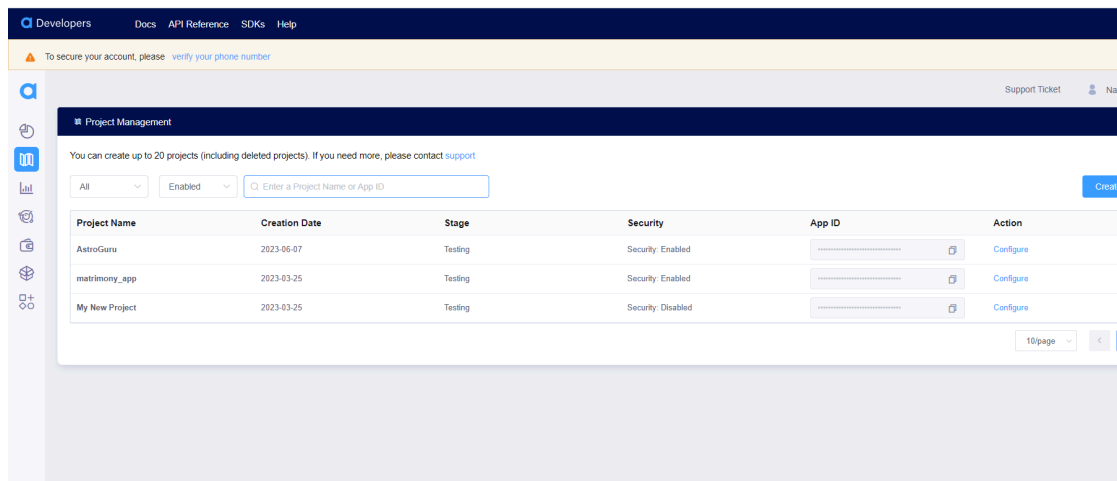
The Agora account is used to make a video call between 2 users in the mobile app.

#### i. Log in or sign up for the [Agora Console](#)



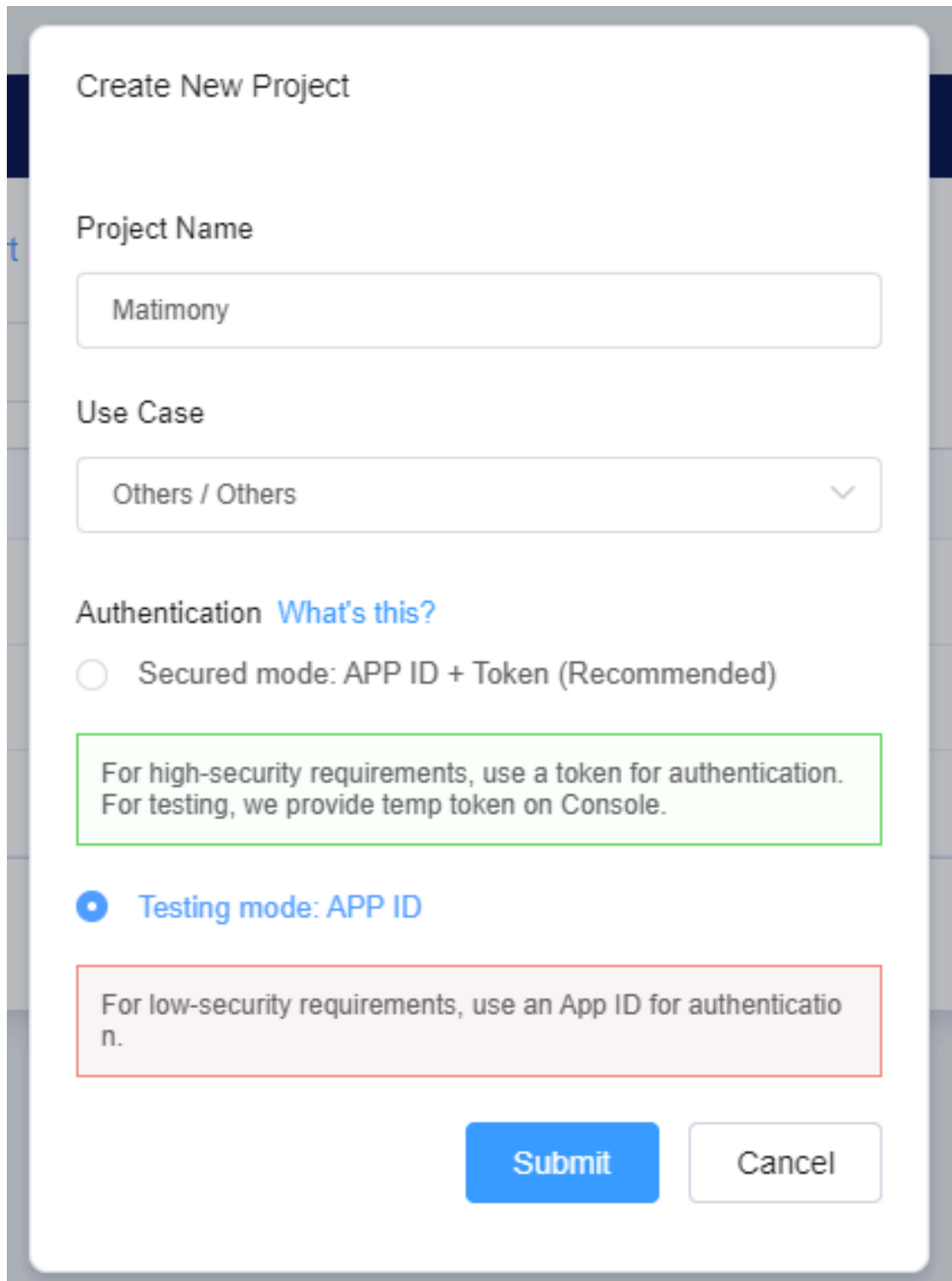
#### ii. Create Agora Project

##### 1. From the side menu, Click on Project Management





## 2. Click on Create a Project Button



Create New Project

Project Name

Matimony

Use Case

Others / Others

Authentication [What's this?](#)

Secured mode: APP ID + Token (Recommended)

For high-security requirements, use a token for authentication. For testing, we provide temp token on Console.

Testing mode: APP ID

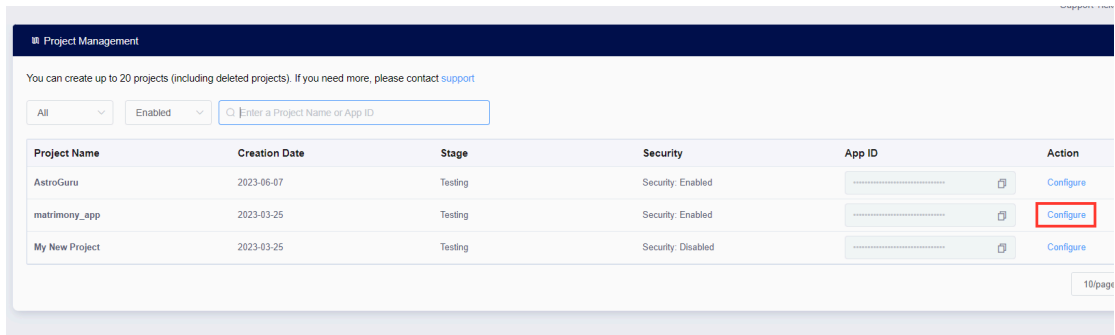
For low-security requirements, use an App ID for authentication.

Submit Cancel

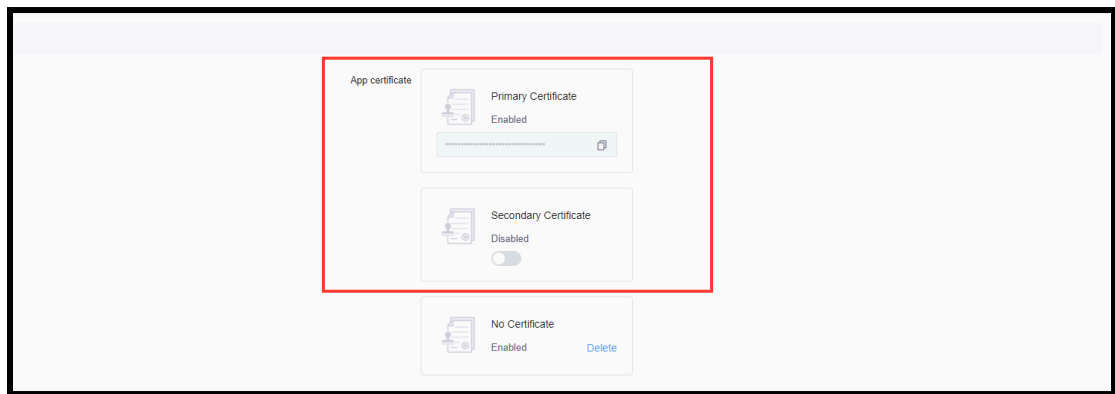
## 3. Give the project name select Other in Use Case and check Authentication in Testing Mode

iii. Config Project in Agora

4. Click on Configure button

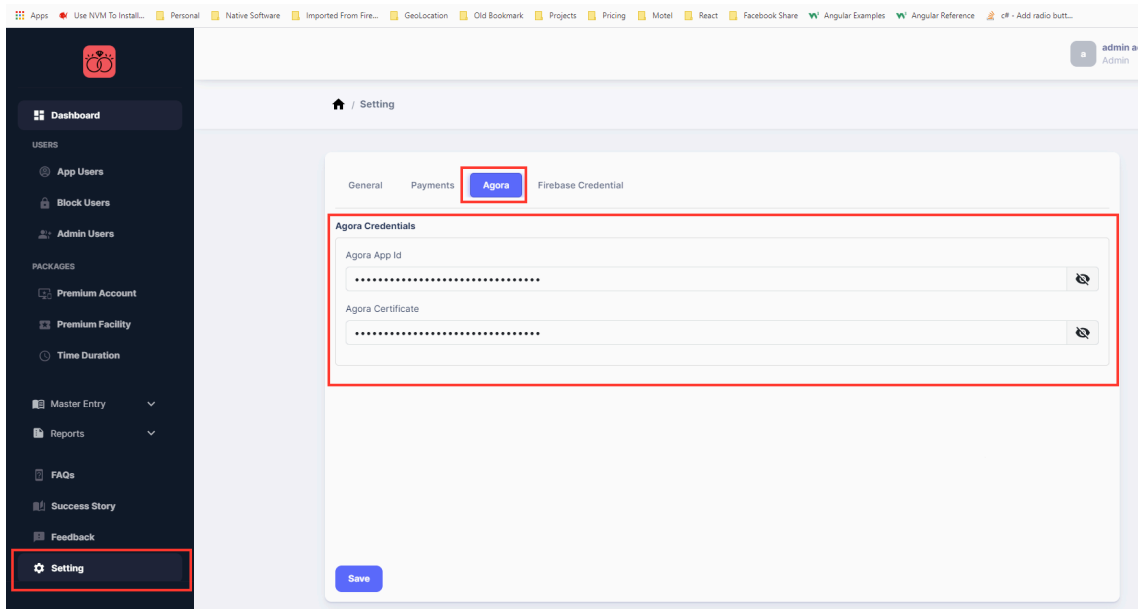


5. Enable primary certificate and Disable Secondary Certificate



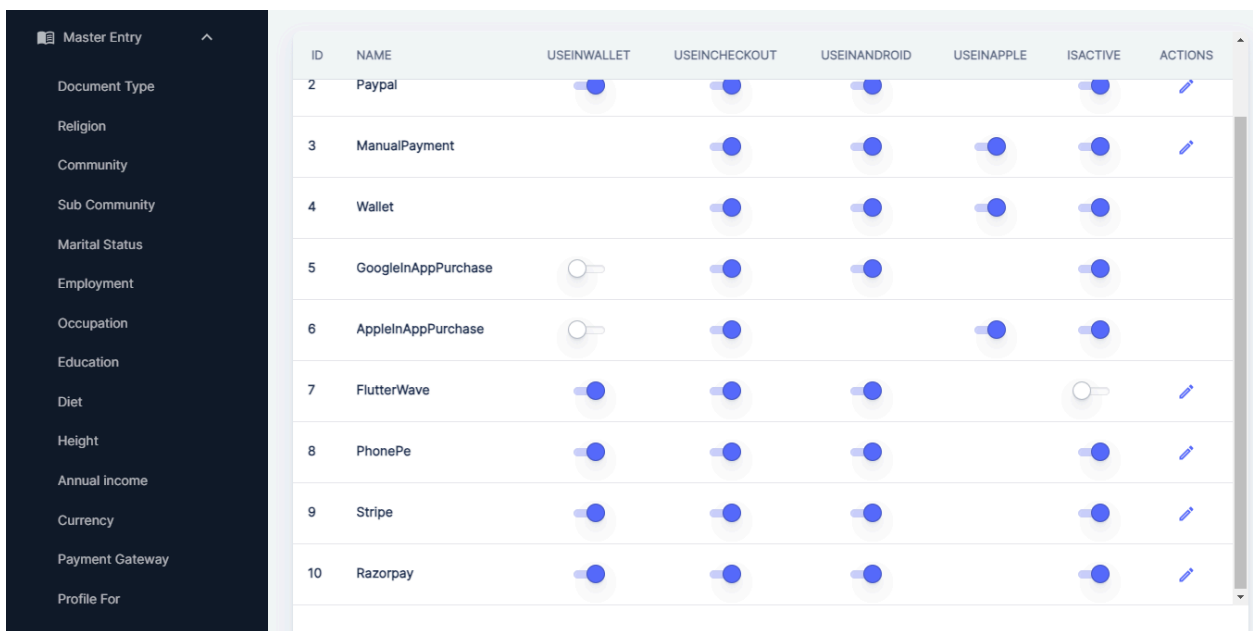
iv. After Successfully Deploying the Admin panel on live (after completing step 4), Login into the Admin Panel and change the Agora key on the live

v. Change your **Agora credentials** from the **Agora** tab on the **Setting** page



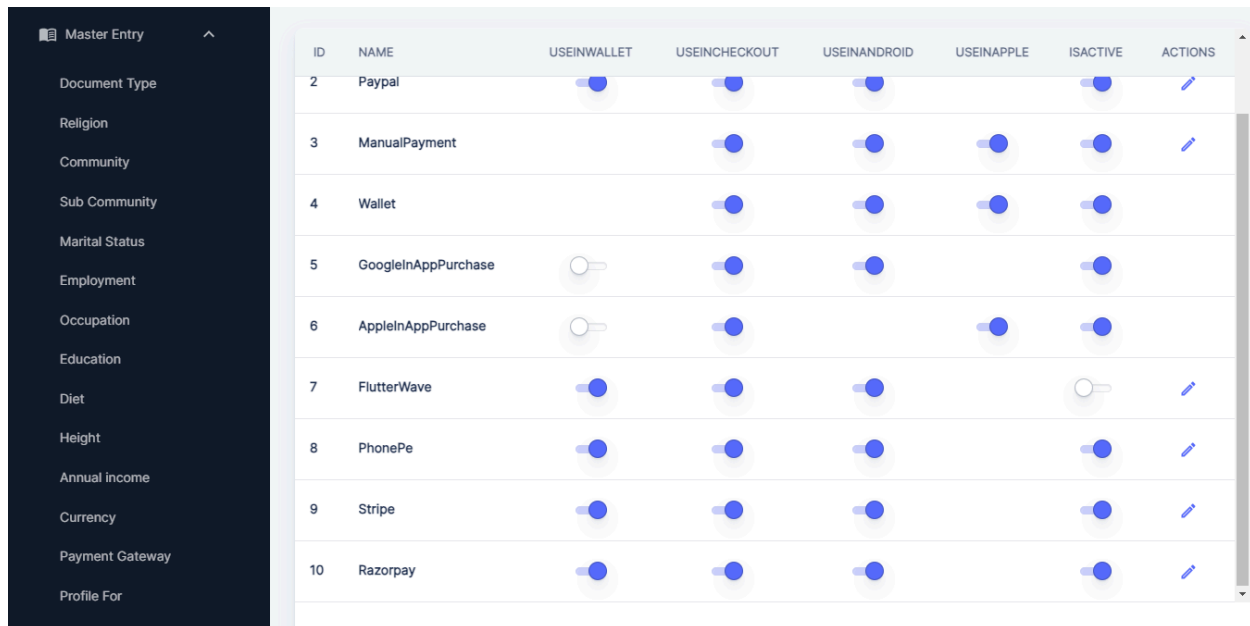
d. Razorpay Setup

- i. Setup RazorPay from [this link](#).
- ii. After generating Razorpay KeyId and Razorpay Secret Key from the link, Setup them in the admin panel.
- iii. In the **Admin Panel** Go to Master Entry from the menu and then click on the Payment Gateways, On this page change your credentials in the Razorpay Section.



## e. Stripe Setup

- i. Setup Stripe from [this link](#).
- ii. After generating the Stripe Secret key from the link, set it up in the admin panel.
- iii. In the **Admin Panel** Go to Master Entry from the menu and then click on the Payment Gateways, On this page change your credentials in the Stripe Section.

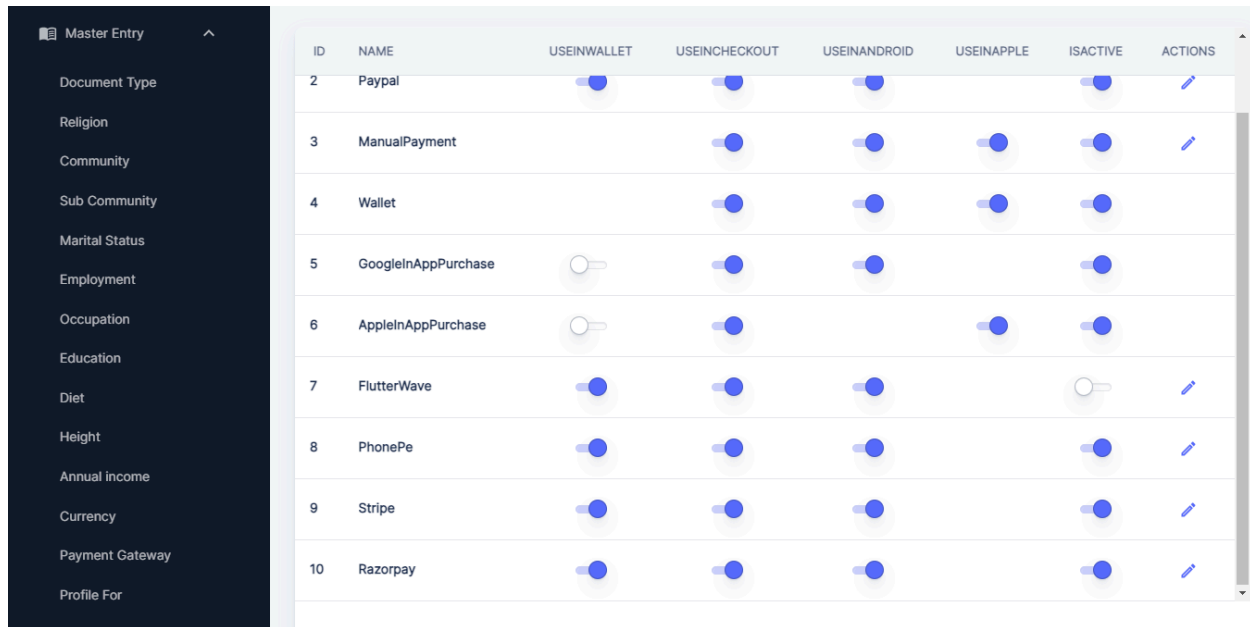


The screenshot shows the 'Master Entry' page in the Admin Panel. On the left is a dark sidebar menu with options: Document Type, Religion, Community, Sub Community, Marital Status, Employment, Occupation, Education, Diet, Height, Annual income, Currency, Payment Gateway, and Profile For. The main content area displays a table of Payment Gateways with columns for ID, NAME, USEINWALLET, USEINCHECKOUT, USEINANDROID, USEINAPPLE, ISACTIVE, and ACTIONS. The table contains 10 rows of data.

ID	NAME	USEINWALLET	USEINCHECKOUT	USEINANDROID	USEINAPPLE	ISACTIVE	ACTIONS
2	Paypal	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<a href="#">✎</a>
3	ManualPayment		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<a href="#">✎</a>
4	Wallet		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
5	GoogleInAppPurchase	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	
6	AppleInAppPurchase	<input type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
7	FlutterWave	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input type="checkbox"/>	<a href="#">✎</a>
8	PhonePe	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<a href="#">✎</a>
9	Stripe	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<a href="#">✎</a>
10	Razorpay	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<a href="#">✎</a>

## f. PhonePe Setup

- i. Setup PhonePe from [this link](#).
- ii. Add PhonePe MerchantId, Redirect Mode, Salt Key, Salt Index, and Environment Value from the link in the admin panel.
- iii. In the **Admin Panel** Go to Master Entry from the menu and then click on the Payment Gateways, On this page change your credentials in the PhonePe Section.

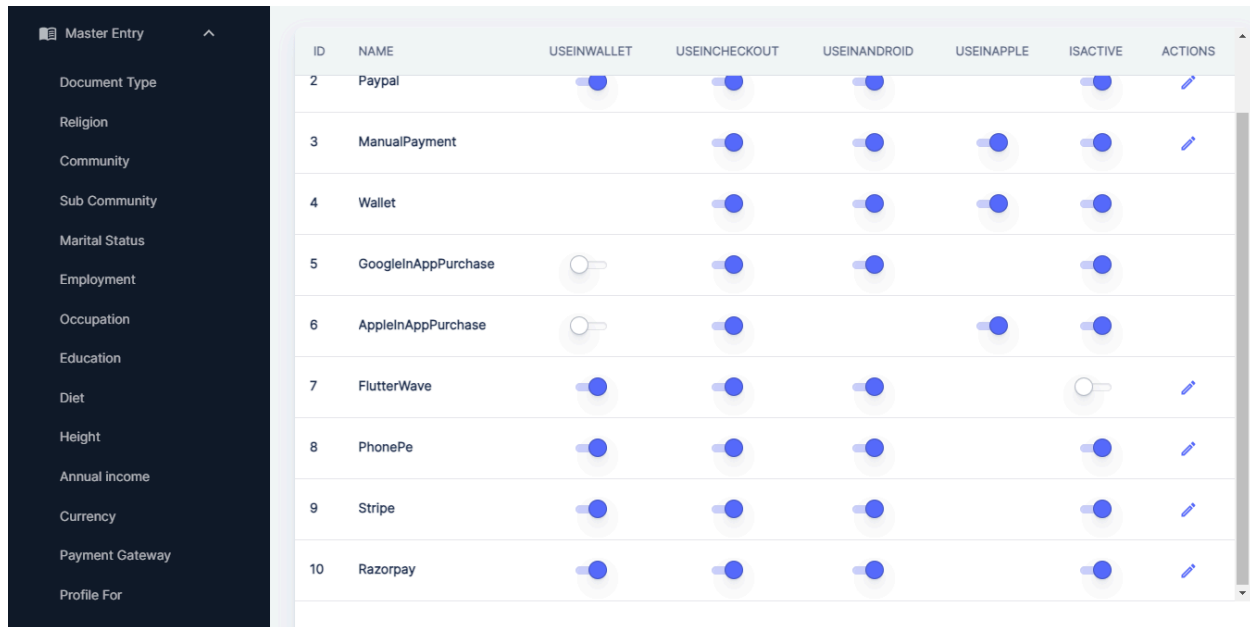


The screenshot shows the 'Master Entry' menu on the left and a table of Payment Gateway settings on the right. The table has columns for ID, NAME, USEINWALLET, USEINCHECKOUT, USEINANDROID, USEINAPPLE, ISACTIVE, and ACTIONS. The rows represent different payment gateways with their respective settings.

ID	NAME	USEINWALLET	USEINCHECKOUT	USEINANDROID	USEINAPPLE	ISACTIVE	ACTIONS
2	Paypal	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<a href="#">✎</a>
3	ManualPayment		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<a href="#">✎</a>
4	Wallet		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
5	GoogleInAppPurchase	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	
6	AppleInAppPurchase	<input type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
7	FlutterWave	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input type="checkbox"/>	<a href="#">✎</a>
8	PhonePe	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<a href="#">✎</a>
9	Stripe	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<a href="#">✎</a>
10	Razorpay	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<a href="#">✎</a>

## g. FlutterWave Setup

- i. Set up FlutterWave from [this link](#).
- ii. Add FlutterWave MerchantId, Publickey, Secretkey, and Encryption key from the link in the admin panel.
- iii. In the **Admin Panel** Go to Master Entry from the menu and then click on the Payment Gateways, On this page change your credentials in the FlutterWave Section.



The screenshot displays a web application interface for configuring payment gateways. On the left is a dark sidebar menu with categories like 'Master Entry', 'Document Type', 'Religion', 'Community', 'Sub Community', 'Marital Status', 'Employment', 'Occupation', 'Education', 'Diet', 'Height', 'Annual income', 'Currency', 'Payment Gateway', and 'Profile For'. The main area shows a table with columns for ID, NAME, USEINWALLET, USEINCHECKOUT, USEINANDROID, USEINAPPLE, ISACTIVE, and ACTIONS. The table lists 10 payment gateways with their respective toggle settings.

ID	NAME	USEINWALLET	USEINCHECKOUT	USEINANDROID	USEINAPPLE	ISACTIVE	ACTIONS
2	Paypal	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<a href="#">✎</a>
3	ManualPayment		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<a href="#">✎</a>
4	Wallet		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
5	GoogleInAppPurchase	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	
6	AppleInAppPurchase	<input type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
7	FlutterWave	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input type="checkbox"/>	<a href="#">✎</a>
8	PhonePe	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<a href="#">✎</a>
9	Stripe	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<a href="#">✎</a>
10	Razorpay	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<a href="#">✎</a>

## USEFUL LINKS

- To set up NodeJS with Typescript from scratch you can use [this link](#)
- To set up MySQL database you can use [this link](#)

This document was last updated on 14 August 2024.