# **Technology Used**

- Flutter with Dart Language for Mobile App
- Laravel 10 for Admin Panel & API
- MySQL for Database

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

(We have provided the steps to set up using Visual Studio Code Editor. You can use other editors also. Steps may vary based on your editor.)

## 1. Setup Prerequisite (If not available)

- a. Install Visual Studio Code (VSCode) from this link
- b. Install Composer from this link with the latest version.
- c. Install PHP(xampp) from this link with the latest version.
- d. Install Laravel from this link (Minimum version 9 -10)
- e. Install and set up Flutter from this link
- f. Install MySQL from this link

(You can choose the MySQL edition based on your needs)

- g. Install MySQL Workbench from this link (This is optional)
- 2. Setup the Database (Technology MySQL)
  - a. Execute the SQL script from the database.sql file in MYSQL (It will create the database and add the master data)
    - i. Command to execute script using MySQL command prompt

#### mysql -u username -p database\_name < file.sql

- ii. Step to execute script in MySQL Workbench
  - 1. Open MySQL Workbench and create your live server connection

- 2. Open that connection and click on **Data Import** from the **Server Menu**
- 3. Check the **Import from Self-Contained File** radio button and select SQL file
- 4. Click on the **new** button which is in **Default Schema to be Imported to** section. And give the database name "**astrologic**". If you want to change the database name then you need to change the database name in the script.
- 5. Click on the **Start Import** button
- b. Once the script is executed successfully, you should able to see the database with the name "astrologic"

#### NOTE:

- This common error often occurs when executing scripts due to a mismatch in character sets, such as 'utf8' and 'utf8\_general\_ci.' To resolve these errors, it may be necessary to replace character sets, switching from 'utf8\_general\_ci' to 'utf8' and vice versa.
- If you need to change the database name, database user name, or database password, you can change it but then you will need to change them respectively in the API in the .env file.
- The default user is root and If you need to create a new MySQL user, it's essential to grant the appropriate permissions; otherwise, queries may fail with an exception like 'Access denied for user 'xxxx'@'xxxx' to database astrologic.' To address this issue, please refer to the following links for guidance and solutions.
- <u>https://stackoverflow.com/questions/1559955/host-xxx-xx-xxx-is-not-allowed</u> -to-connect-to-this-mysql-server

# 3. Setup the API & Admin Panel (Technology Laravel)

- a. Open VSCode
- b. Open the AdminPanel folder into the VSCode from the ZIP file.
- c. Please execute the following command in the VSCode terminal to update the packages as per the package.json file.

#### composer update

Please make sure the installation gets completed successfully without any errors.

d. Setup database credentials in API & Admin Panel.



- i. Locate the .env file in your Laravel project.
- ii. If the .env file does not exist, please rename the **.env.example** to **.env** and update that file.
- iii. Please replace "YOUR\_DB\_CONNECTION" with the database name you set up in Step 2.
- iv. Please replace "YOUR\_DB\_HOST" with your MySQL hostname.
- v. Please replace "DB\_PORT" with your port address.
- vi. Please replace "YOUR\_DB\_NAME" with your MySQL database name.
- vii. Please replace "YOUR\_DB\_USERNAME" with your MySQL username. (You can also create a separate username for your database. For that please follow this link)

- viii. Please replace "YOUR\_DB\_PASSWORD" with your MySQL password for the above username.
- e. To run an API please execute the following command

#### php artisan serve

#### NOTE:

• To run an API with your IP Address and Port number use the below command

### php artisan serve --host=<YOUR\_IP\_ADDRESS>:8000

f. To set the API & Admin Panel on the Live Server

You can host on whichever hosting server is available to you, here we have provided steps for two hosting servers.

#### i. Setup Cpanel

- 1. Log in to your shared hosting cPanel account.
- 2. Open the "File Manager" in cPanel.
- 3. Create a folder "YOUR\_PROJECT\_NAME".
- 4. Extract your project zip in this folder(Make zip except for public folder)

P File Manager		Search	All Your Files 🗸 for	Go 🗘 Settings		
+ File + Folder @Copy + Move . ✓Compress	File 🕂 Folder 🗘 Copy 🕂 Move 🛓 Upload 🛓 Download 🗙 Delete 🖱 Restore 📲 Rename 🥒 Edit 🖄 HTML Editor 🎾 Permissions 🐵 View 🖍 Extract					
😭 laravel Go	☆ Home J Up One Level ← Back → Forward	C Reload 🛛 🗹 S	Select All Unselect All	Trash Empty Trash		
Collapse All	Name	Size Las	st Modified	Type Permissions		
<pre>- W (home2/homeath) + etc laravel + logs lscache + mail + public_ftp + public_ftp + ssl + tmp + var</pre>	Iaravel.zip     Sownload     Ovew     Move     Copy     Rename     Change Permissions     X Delete           Compress	12.62 MB Tod	lay, 11:28 AM	package/x-generic 0644		

- 5. Upload the public folder inside the 'Public-Html' folder
- 6. Open index.php



- 7. Write your project name to every require\_once \_\_DIR\_\_\_
- 8. Next, go to MySQL and create a new database

Databa	ases						÷.
	phpMyAdmin		MySQL® Dat	abases	2	MySQL® Database Wizard	
-	Remote MySQL®		PostgreSQL [	Databases	2	PostgreSQL Database Wizard	
:: 0)))	<u>phpPgAdmin</u>						
MyS	QL® Databases						
Manage large a and online sho	mounts of information over the web easily oping carts. For more information, read th	y. MySQL datab e <u>documentati</u>					anagement systems,
						1	Jump to MySQL Users
Create	New Database						
brayosit_	laravel						
Create Dat	tabase						

9. Access your domain in a web browser to check if your app is running.

#### ii. Setup Plesk Panel

- 1. Log in to your shared hosting Plesk account.
- 2. Open Plesk and navigate to the 'Add Domain' button. From the drop-down menu that appears, select 'Laravel site' to begin establishing a new domain.

Native Software

Cancel



7

# Adding New Domain

#### Choose a way of creating a website

	Blank website A starter page for your HTML or PHP site (the previous default)		Upload files From a local machine
	Deploy using Git Pull the files from a Git repository	Ŵ	WordPress site A website running on the latest WordPress version
	Import website From another hosting server	<b>B</b>	Laravel site Create a new website by installing a Laravel application
	Node.js application A website running on Node.js		

# 3. If you don't have a domain name for your website yet, you can use a temporary domain name.



4. Click on the temporary domain name and you will get a domain name as shown

 $\times$ 

Select your domain name				
Registered domain name	Temporary domain name			
l already have a registered domain name	I don't have a registered domain name yet			
Your temporary domain name brave-lewin.103-150-136-240.plesk.page				
Show more				
Assign this domain to a customer Webspace *				
Create a new subscription	•			
<ul> <li>Webspace settings</li> </ul>				

5. Click Add Domain and you will see the below screenshot



6. After the website has been created, Plesk will prompt you to select between installing a default skeleton Laravel application or pulling the application from a remote Git repository. Your selection will be saved automatically. Installing the skeleton is the best way to get acquainted with the Laravel Toolkit's capabilities for the time being.



7. Wait for a few seconds.

D	eploying project	_
<b>~</b>	Creating a Git repository	Done
•	Creating Laravel skeleton	
•	Committing to the Git repository	Queued
•	Installing Composer dependencies	Queued
•	Configuring Laravel application	Queued
•	Configuring hosting settings	Queued
•	Configuring PHP environment	Queued
•	Preparing Node.js environment	Queued

## 8. And it is ready

Laravel application for brave-lewin.103-150-136-240.plesk.page				
perA Texastand engagementer Texastand engagem	Dashboard Artisan Compo Application Info	oser Nodejs Deployment		
	URL Repository	brave-lewin.103-150-136-240.plesk.page 0° https://brave-l_0tzxe0hhj97c@brave-lewin.103-150-136-240.plesk.page/plesk.git/laravel_e4b34a		
	Last commit	Date: Fri Dec 30 09:33:33 2022 40530		
S Manage domain		Laravel skeleton		
E Logs	Settings Environment variables (.env) Maintenance mode	Edit		
	Maintenance Hidde			

- 9. Your very first Laravel application has been published on the World Wide Web!
- 10. Navigate to the Laravel dashboard labeled "Information," then click the "Manage domain" button. Following this link will bring you to the "Hosting" dashboard.
- 11. After arriving at that location, select "Databases," and then after that, "Add Database".

Home Subscriptions brave-lewin.103-150-136-240.plesk.page Websites & Domains brave-lewin.103-150-136-240.plesk.page aravel Toolkit						
Laravel application fo	Laravel application for brave-lewin.103-150-136-240.plesk.page					
pirsk 👤	Dashboard Artisan Comp	poser Node.js Deployment				
The is a default in design encounter in baser works and 10 is the designed agent in Pires - Parational agent and an and an and 1 and Pharagent and an and an and 1 and Pharagent and an an an and an	Application Info					
ligen frei meine verbraute gener Bener frei werte eingen	URL	brave-lewin. 🗧 💦 plesk.page D'				
Value Field And analogy account of a participants of a rate matter matching on any account of a participant of table matching on any account of table of table on any account of table of table on a matching on any account of table of table on any account of table on a matching on any account of table of table on any account of table on a matching on any account of table of table on a set of	Repository	https://brave-l_0tzxe0hhj97c@brave-lewin.	đ			
	Last commit					
		Author: admin = Date: Fri Dec 30 09:33:33 2022 +0530				
N and another to the three strengtheness on an and and a strengtheness of strength		Lanavel skeleton				
🖗 Manage domain						
⊞ Logs	Settings					
	Environment variables (.env)	Edit				

	Databases for brave-lewin.	.plesk.page 👃			
	Information: The database admin_ was created.				음 User Management
	Here you can create new or manage existing databases.				Database Servers
	🕂 Add Database			Z	🔄 Backup Manager
	Admin_ Related to brave-lewin.	plesk.page Change		phpMyAdmin	
	Host: localhost:3306 (MariaDB) Users: localhost Tables: 0	O Size: 0 B			
		▲ SHOW LESS			
<	🎒 phpMyAdmin	Connection Info	📑 Сору		
	Second Dump	1 Import Dump	Oheck and Repair		
	Move to Subscription	X Remove Database			

12. When you are finished creating the database, you will need to modify the configuration file for the Laravel application so that it contains information about how to connect to the newly created database.

- 13. Click the button labeled "Manage Laravel Application" located on the "Hosting" dashboard. Clicking on this link will take you to the Laravel card.
- 14. After arriving at that location, open the.env file and add the database credentials.

Laravel application fo	or brave-lewin	. <mark>En stand</mark> .plesk.page
pjesk 🚬	Dashboard Artisan Compo	oser Node.js Deployment
The based on the programmer processing of the second secon	Application Info	brave-lewin
March Nat And an experimental second	Repository	https://brave-l_0tzxe0hhj97c@brave-lewin.
	Last commit	commit Author: admin - Date: Fri Dec 30 09:33:33 2022 +0530
Manual again and the second and the second again and the second again ag		Laravel skeleton
Manage domain		
≣ Logs	Settings	
	Environment variables (.env)	Edit
	Maintenance mode	Disabled



- 15. Execute the 'artisan migrate' command to create a database as the final step. And yes, the 'artisan' utility can be executed directly from within the Plesk user interface. You will need to select the migrate' command from the list of preloaded commands after you have navigated to the 'Artisan' tab.
- 16. However, there is more to come! In particular, let's look at some additional features that save you time when hosting Laravel applications.
- 17. It is important to point out that you can rapidly invoke the 'composer' and 'npm' commands by simply selecting them from the list of

preloaded \* commands, which I discovered to be an indispensable resource:



LUI	aver appreador	
	1 1	
		Dashboard Artisan Composer Node.js Deployment
	89	composer D
	CONTRACTOR      CONTRACTOR	command composer is not delined.
	Personal         Personal           Bit State         Personal           Table Table         Personal	# composer update
<		INFO Discovering packages.
		laravel/sail DONE
		lanavel/sanctum
S M	anage domain	nesbot/carbon DONE
		nunomaduro/collision
:= L0	gs	nunomaduro/termwind DONE
		spatie/laravel-ignition DONE
		1000 No publishable resources for tag [laravel-assets].
		*

18. If your Laravel application makes use of Laravel Task Scheduling, then in addition to the typical 'artisan schedule: list' command, you can quickly review all scheduled jobs by switching to a different tab.

# 4. Setup Customer App (Technology Flutter)

- a. Initial steps to set up and run mobile app
  - i. Open the App folder in the VSCode
  - ii. Run the following commands in the VSCode Terminal

## flutter clean flutter pub get

- iii. Additional steps to set up for iOS (You can skip these steps if you don't want to set up for iOS)
  - 1. In the VSCode terminal, go to the ios directory

(using the command **cd ios**)

2. Run the following command to install pods

#### pod install

iv. Connect your Android or iOS device with your machine

(To run on an Apple device, you must have an Apple computer)

v. Run the following command to run on an Android or iOS device

#### flutter run

vi. To upgrade the Flutter version run the following command in the VSCode Terminal

(Only if your Flutter version is lower than mentioned in this document)

#### flutter upgrade

b. Change API base URL

After the setup of your API and Admin panel, you have to change your API base URL, for that go to the file located at **lib\utils\global.dart** 



- c. Change App Name
  - i. Change the app name in the Android App
    - 1. Change the app name in the file located at android/app/src/main/AndoidManifest.xml



- ii. Change the app name in the iOS App
  - 1. In VSCode
    - a. Go to ios/Runner/info.plist
    - b. Change string of key **CFBundleDisplayName**

<string>\$(DEVELOPMENT_LANGUAGE)</string>	
<key>CFBundleDisplayName</key>	
<string>AstroLogic<th></th></string>	
<key>CFBundleExecutable</key>	
<pre>/stmings\$/EVECUITABLE_NAME}//stmings</pre>	

- 2. In XCode
  - a. Right-click on the **iOS** folder and Choose Open in Xcode Option
  - b. Click on the folder icon left side of the XCode window
  - c. Select Runner.
  - d. Select Target runner
  - e. From the General Tab Go to identity
  - f. Change Display Name

∨ Identity		
App Category	None	0
Display Name	AstroLogic	+
Bundle Identifier	com.example.astrologic	→
Version	1.0.0	+
Build	1	+

d. Change Package Name/Bundle ID

An app's package name is a unique identifier that is automatically created when you create an app. The term used for iOS apps is "**bundle ID**" and for Android apps, it is "**package name**"

- i. Set Package Name for Android App
  - 1. Change the package name in the file located at android/app/src/main/AndoidManifest.xml

android	> app > src > main > 🔉 AndroidManifest.xml
1	<pre>kmanifest xmlns:android="http://schemas.android.com/apk/res/android"</pre>
	<pre>package="com.example.astrologic"}</pre>
	<pre><uses-permission android:name="android.permission.INTERNET"></uses-permission></pre>

2. Change the package name in the file located at android/app/src/debug/AndoidManifest.xml

android	> app > src > debug > 🔊 AndroidManifest.xml
1	manifest xmlns:android="http://schemas.android.com/apk/res/android"
	package="com.example.astrologic"
	<pre><!-- The INTERNET permission is required for development. Specifically,</pre--></pre>

3. Change Package Name in file which is located at android/app/src/Profile/AndoidManifest.xml

android	> app > src > profile > 🔉 AndroidManifest.xml
1	manifest xm <u>lns:android="http://sche</u> mas.android.com/apk/res/android"
	package="com.example.astrologic"
	The INTERNET permission is required for development. Specifically,</td
	the Flutter tool needs it to communicate with the running application
	to allow setting breakpoints, to provide hot reload, etc.

4. Change the Package Name in a file that is located at android/app/build.gradle



5. Change the folder structure for the below path as per your package name.

```
android\app\src\main\java\<mark>com\example\astrologic</mark>\
```

 Change Package Name in the file which is located at android\app\src\main\java\com\example\astrologic\MainActivity. java



- ii. Set Bundle ID for iOS App
  - 1. In VSCode
    - a. Go to ios/Runner/info.plist
    - b. Change the string of key CFBundleldentifier



- 2. In XCode
  - a. Right-click on the **iOS** folder and Choose Open in Xcode Option
  - b. Click on the folder icon left side of the XCode window
  - c. Select Runner.
  - d. Select Target runner
  - e. In general, Tab Go to identify

f. Change Bundle Identifier

∨ Identity				
	App Category	None ≎	Ð	
	Display Name	AstroLogic	+	
	Bundle Identifier	com.example.astrologic	→	
	Version	1.0.0	+	
	Build	1	+	

- g. In Signing & Capabilities Go to Signing
- h. Change Bundle Identifier

- e. Create and set Keystore file for Android
  - i. Create a Keystore.jks file if not exist using the below command in the terminal

keytool -genkey -v -keystore "path\keystore.jks" -storetype JKS -keyalg RSA -keysize 2048 -validity 10000 -alias keystore

- ii. Fill in all the details asked while executing the above command
- iii. Recommended. After creating your keystore.jks file, please put it in the **android/app** folder

iv. Create a key.properties file in the **Android** folder and add the details in the file as per the below screenshot.



#### NOTE:

- If you have changed any default value for any of these keys (storePassword, keyPassword, keyAlias, storeFile) while creating the keystore.jks file, then please also change them to the key.properties file.
- If you place your keystore.jks file somewhere else in the project than mentioned in step 5.c.iii then please change storeFile key value accordingly.
- For more details please refer to this link
- f. Create Firebase Account & Project

In this project, we are using the following Firebase services.

- i. Push Notification
- ii. Phone Authentication
- iii. Firebase Analytics
- iv. Firebase Firestore
- v. Firebase Dynamic Link

For this, you need a Firebase account and a project set up in the Firebase. Please follow the below steps for this,

- i. Go to the Firebase console
- ii. Sign up if you don't have a Google Account or want to create a new account for your project. Otherwise, sign in with your Google Account.
- iii. Click on Add Project
- iv. Enter your project name



v. Select Default Account for Firebase

(or you can create a new account)

×	Create a project (Step 3 of 3)
	Configure Google Analytics
	Choose or create a Google Analytics account 💿
	, Default Account for Firebase
	Automatically create a new property in this account 💉
	Upon project creation, a new Google Analytics property will be created in your chosen Google Analytics account and linked to your Firebase project. This link will enable data flow between the products. Data exported from your Google Analytics property into Firebase is subject to the Firebase terms of service, while Firebase data imported into Google Analytics is subject to the Google Analytics terms of service. Learn more Pd

- vi. Create project
- g. Set up Android App in Firebase Project
  - i. Go to the Firebase console
  - ii. Select the project you created in step 5.d.vi
  - iii. Go to Project Setting
  - iv. In the **General** Tab click on the **Add App** button
  - v. Select Android
  - vi. Fill out the form and click on the **Register App Button**

(Please check the below screenshot for reference)

× Ac	dd Firebase to your Android app
1	Register app
	Android package name 🕥
	com.company.appname
	App nickname (optional) 💿
	My Android App
	Debug signing certificate SHA-1 (optional) ②
	00:00:00:00:00:00:00:00:00:00:00:00:00:
	Required for Dynamic Links, and Google Sign-In or phone number support in Auth. Edit SHA-1s in Settings.
	Register app
2	Download and then add config file
3	Add Firebase SDK
4	Next steps

- vii. You need SHA keys (SHA-1 and SHA-256) to add once you create the Android App in the above steps.
  - 1. To Generate debug SHA use the below command

keytool -list -v -keystore "Your directory path\debug.jks" -alias androiddebugkey -storepass android -keypass android

2. To Generate release SHA use the below command

keytool -list -v -keystore "your directory path\keystore.jks" -alias androidreleasekey -storepass your store password -keypass your key password

After generating the debug and release SHA, you have to add them in the Firebase Console where you have created the Android app.

Please check the screenshot below for the reference.

Certificate fingerp	int
0:00:00:00:00	:00:00:00:00:00:00:00:00:00:00:00:00:00
	Cancel Save

- viii. Download the google-services ison file from Firebase project settings and paste it at the **android/app** location.
- ix. Add Firebase SDK Add the plugin as a build script dependency to your project-level build.gradle file:

```
buildscript {
  repositories {
    // Make sure that you have the following two repositories
    google() // Google's Maven repository
   mavenCentral() // Maven Central repository
  }
  dependencies {
    . . .
   // Add the dependency for the Google services Gradle plugin
   classpath 'com.google.gms:google-services:4.3.15'
  }
}
allprojects {
  . . .
  repositories {
   // Make sure that you have the following two repositories
   google() // Google's Maven repository
   mavenCentral() // Maven Central repository
  }
}
```

x. Then, in your module (app-level) build.gradle file, add both the google-services plugin and any Firebase SDKs that you want to use in your app:

## h. To Setup Dynamic Links

i. In the Firebase console click on "Get Started" or "Set Up" to start the process of setting up Dynamic Links for your project.

Dynamic Links		
Send users to the right place in your app whether or not it is already installed		
By continuing below, you understand that your Android/Apple app info (e.g. package name, bundle ID, and App ID Prefix) will be publicly visible in asset/association files.		
Add URL prefix		
1 Create 2 Configure 3 Verify 3	4 Finish	
Brand your links completely with a URL prefix made up of a custom Dy yourdomain.com) and optional path (ex: yourdomain.com/mypath). C professional and contextual, resulting in more clicks on links.	namic Links domain for: ustomized domain fook more	
Domain ③	_	
	Cancel Continue	0
		Watch I

ii. After you create Dynamic Links, you need to set up your app codes **lib\utils\global.dart** 



## In Xcode

• Add associated domain capabilities and set the domain

Ge	eral Signing & Capabilities Resource Tags Info Build Settings Build Phases Build Rules						
PROJECT	+ Capability All Debug Release Profile						
🛃 Runner	> Signing						
TARGETS	V   Associated Domains						
Runner	Domains https://astrologic.page.link						
	+ -						

## i. Setup Firebase iOS App

- i. Go to the Firebase console
- ii. Select the project you created in step 5.d.vi
- iii. Go to Project Setting
- iv. In the General Tab click on the Add App button
- v. Select iOS
- vi. Fill out the form and click on the **Register App** Button

(Please check the below screenshot for reference)

× Ad	dd Firebase to your Apple app
1	Register app
	Apple bundle ID 💿
	com.company.appname
	App nickname (optional) 🕥
	My Apple app
	App Store ID (optional) ③
	123456789
	Register app
2	Download config file
3	Add Firebase SDK
4	Add initialisation code
5	Next steps
5	Here orepo

- vii. Download the GoogleService-info.plist file from Firebase project settings and paste it at the **ios/Runner** location in the app
- viii. XCode Project Setting

This step covers reviewing the most important settings in the XCode workspace. For detailed procedures and descriptions, see <u>Prepare for</u> <u>App Distribution</u>

- 1. Navigate to your target's settings in XCode:
  - a. Open the default Xcode workspace in your project by running the below command in a terminal window from your Flutter project directory.

#### open ios/Runner.xcworkspace

- b. To view your app's settings, select the Runner target in the Xcode navigator.
- 2. Verify the most important settings
  - a. In the Identity section of the General tab
    - i. **Display Name** (The display name of your app.)
    - ii. **Bundle Identifier** (The App ID you registered on App Store Connect.)
  - b. In the Signing & Capabilities tab
    - Automatically manage signing (Xcode should automatically manage app signing and provisioning. This is set true by default, which should be sufficient for most apps. For more complex scenarios, see the <u>Code Signing Guide</u>)
    - ii. **Team** (Select the team associated with your registered Apple Developer account. If required, select Add Account..., then update this setting.)

Gen	ieral	Signing & Ca	apabilities	Resour	ce Tags	Info	Build Settings	Build Phases
PROJECT	+ Ca	pability	All Debug	Relea	se Pro	file		
🛃 Runner	~	Signing						
					✓ Autom	atically 1	manage signing	as ann IDs and
TARGETS					certific	ates.	te and update prom	es, app iDs, and
Runner				Team	None			۵
			Bundle Id	entifier	com.exa	mple.ast	trologic	
		∨ ios						

- c. In the deployment section of the build settings tab:
  - i. iOS Deployment Target
    - 1. The minimum iOS version that the app supports is 11.0.
    - 2. The General tab of your project settings should resemble the following:

	General	Signing & Capabilities	Resource Tags	Info Build Settings Build Phases	Build Rules
PROJECT	`	Supported Destinations			
🛃 Runner		Destination		SDK	
TARGETS		iPhone + -		iOS	
Runner					
		Minimum Deployments			
			iOS	11.0	• +
	~	dentity			
			App Category	None	\$
			Display Name	AstroLogic	+
			Bundle Identifier	com.example.astrologic	$\rightarrow$
			Version	1.0.0	+
			Bulla	1	+
	-	Deployment Info			
		1	Phone Orientation	<ul> <li>Portrait</li> <li>Upside Down</li> </ul>	
				<ul> <li>✓ Landscape Left</li> <li>✓ Landscape Right</li> </ul>	
			Status Bar Style	Default	٥
				Hide during application launch	

3. For a detailed overview of app signing, see <u>Create</u>, <u>export</u>, and <u>Delete signing certificates</u>.

## j. Change App Icon

i. For Android

Replace the icons in the **android\app\src\main\res** folder as shown in the below image.



- ii. For iOS
  - 1. Replace the icons in the below folder as shown in the below image

#### ios\Runner\Assets.xcassets\Applcon.appiconset



- 2. Change icons using XCode
  - a. Right-click on the iOS folder Choose Open in Xcode Option
  - b. Click on the folder icon on the left side of the XCode window



- c. Select Runner.
- d. Select Target runner
- e. Go to App Icons And Launch Images
- f. Click the right arrow button of the app icon source

🖮 🛛 🔍 🛆 🔗 👸 🗖 🗐	Runner	C Assets
> 🙆 Pods	PROJECT Gener	al Signing & Capabilities Resource Tags Info Build Settings Build Phases Build Rules
	🖾 Runner	App Icons and Launch Screen
	TARGETS	App Icon Source Include all app icon assets Launch Screen File
		<ul> <li>Supported Intents</li> <li>Class Name</li> <li>Authentication</li> </ul>

g. Replace all the icons according to their size



## NOTE:

• If you want to generate the App icon bundle from any image you have, you can generate it from publicly available websites like

https://www.appicon.co/

- k. Build Release for Android
  - i. Open Project in VS Code
  - ii. In Terminal Execute the below commands

flutter clean flutter pub get flutter build apk --release

iii. After making the release, to generate the release bundle Execute the below command

#### flutter build appbundle --release

iv. Get the APK from the below path

#### build\app\outputs\flutter-apk\app-release.apk

- I. Build Release for iOS
  - i. Open Project in XCode
  - ii. Select Archive from the Product Menu



- iii. After successfully archiving select the **Organizer** option from the **Windows menu**
- iv. After clicking on it opens one popup for Archive, Click on the **Distribute App** Button



- v. After successfully done, you can upload this app to your Apple developer account in the TestFlight
- vi. To publish your app from TestFlight please follow this link

# m. Other Options for the Advanced User

#### i. Paths to the images used in the app

Images	Path	Screen Path
Splash screen	assets/images/splash.png	lib\views\splashScreen.dart
Home screen	assets/images/confidential.jpg assets/images/verified-account.j pg assets/images/payment.jpg assets/images/blog.jpg assets/images/youtube.png assets/images/translation.png	lib\views\homeScreen.dart
Search Astrologer screen	assets/images/wallet.png assets/images/customer-service. png assets/images/user.png assets/images/shopping_bag.pn g assets/images/default-user.png	lib\views\searchAstrologerScreen.dart
Astrologer Profile screen	assets/images/chat-bg2.jpg assets/images/palmistry.jpeg	lib\views\astrologerProfile\astrologerProfileScreen.d art
Report Type Screen	assets/images/storeLogo.png	lib\views\reportTypeScreen.dart
History screen	assets/images/whatsapp.png assets/images/nodata.png assets/images/noDataFound.pn g	lib\views\historyScreen.dart
Kundli matching result screen	assets/images/sky.jpg	lib\views\kundli matching\kundliMatchingScreen.dart
Create kundli Screen	assets\images\masterCardLogo. png assets/images/maleGender.png assets/images/femaleGender.pn g assets/images/otherGender.png	lib\views\kudali\createNewKundliScreen.dart
Call screen	assets/images/right.jpg	lib\views\callScreen.dart
Login screen	assets/images/arrow_right.png	lib\views\loginScreen.dart

# ii. Fonts used in the app. If you want to change, you can make the changes in the **pubspec.yaml** file and the **Assets** folder.

	Poppins	asset: assets/fonts/Poppins-Black.ttf
--	---------	---------------------------------------

iii. Colors used in the app. If you want to change the colors you can make the changes in the file **lib/Theme/nativeTheme.dart** 

#	Color code
Primary color	#DB4627
primaryColorLight	#DB4627
primarylconTheme - color	black
Text button - background color	black
Text button - foreground color	#DB4627
Divider theme - color	gray
Card - color	white
TextFormfield - border color	#DB4627
Appbar theme - color	#DB4627
Appbar icon theme- color	black
Drawer icon - color	gray
Checkbox - check color	black
Checkbox - fillcolor	#DB4627
scaffoldBackgroundColor	white
elevatedButton -background color	#F5F5F5
elevatedButton -foreground color	#DB4627

iv. Packages used in the app are listed below. You can find them in **pubspec.yaml** file.

Package Name - Version	Description
Font_awesome_flutter :^ 10.5.0	To access icon of font awesome

material_design_icons_flutter :^5.0.6996	To access icon of material design icons			
flutter_rating_bar :^ 4.0.1	For the star Rating bar			
flutter_html - ^3.0.0-alpha.5	For rendering HTML and CSS as Flutter widgets.			
bubble_tab_indicator: ^0.1.6	For tab bar indicator style			
flutter_image_slideshow: ^0.1.4	For image slideshow			
date_format: 2.0.4	For date format			
video_player: ^2.4.7	For play video			
percent_indicator: ^4.2.2	For circular percent indicator			
photo_view: ^0.14.0	For photo preview			
custom_clippers: ^2.0.0	For custom shape			
timelines: ^0.1.0	For timeline UI			
url_launcher: ^6.1.7	For launching a URL			
webview_flutter: ^3.0.4	Used to display web pages			
fluttertoast: ^8.1.1	Use to show a toast message			
agora_rtm: ^1.5.8	For live chat and join live stream message			
agora_rtc_engine: ^6.1.0	For audio calls and live streaming video call			
permission_handler: ^10.2.0	This plugin provides a cross-platform api to request and check permission			
path_provider: ^2.0.11	For finding commonly used locations on the filesystem			
flutter_otp_text_field: ^1.1.1	For otp text field			
device_info_plus: ^8.0.0	To get device info			
connectivity_plus: ^2.1.0	To check the connectivity of the device			
http: ^0.13.5	For consuming HTTP resources			
dio: ^4.0.6	A powerful HTTP client for Dart, which supports Interceptors, Global configuration, FormData, Request Cancellation, File downloading, Timeout, etc.			
geolocator: ^9.0.1	continuous location updates			
geocoding: ^2.0.5	To retrieve the complete address with the help of Latitude and Longitude and vice-versa			
google_place: ^0.4.7	Use for a search place name			
uuid: ^3.0.6	To generate global identifiers that are unique across space and			

	time.		
mapbox_search: ^3.0.1+2	For Search location		
syncfusion_flutter_pdfviewer: ^20.4.38	For pdf view and download		
flutter_mapbox_autocomplete: ^2.0.0	For search location and autocomplete		
store_redirect: ^2.0.1	To redirect users to an app page in the Google Play Store and Apple App Store		
syncfusion_flutter_gauges: ^20.4.44	To change the value at run time(radial gauge UI)		
package_info_plus: ^1.4.3+1	Get the version of the app		
flutter_countdown_timer: ^4.1.0	Shows remaining time		
flutter_holo_date_picker: ^1.1.0	For Date Picker		
flutter_time_picker_spinner: ^2.0.0	For time Picker		
firebase_dynamic_links: ^5.0.9	To send existing and potential users to any location within your iOS or Android app (Firebase Dynamic Links API)		
screenshot: ^2.1.0	Capture any Widget as an image		
get: ^4.6.1	State management		
razorpay_flutter: ^1.3.4	For payment gateway		
firebase_core: ^2.3.0	To use the Firebase Core API, which enables connecting to multiple Firebase apps.		
firebase_auth: ^4.1.1	For Authentication use Firebase authentication options (Firebase Authentication API).		
firebase_messaging: ^14.1.2	To use Firebase Cloud messaging api( Firebase Cloud Messaging API).		
flutter_local_notifications: ^11.0.1	A cross-platform plugin for displaying local notifications.		
cloud_firestore: ^4.1.0	Cloud Firestore, a cloud-hosted, NoSQL database with live synchronization and offline support on Android and iOS		
intl: ^0.17.0	To Provide internationalization and localization facilities, including message translation, plurals and genders, date/number formatting and parsing, and bidirectional text.		
shared_preferences: ^2.0.15	To store something locally		
intl_phone_field: ^3.1.0	For country-code pick		

and IOS

For picking an image

Share messages, links, or files from your Flutter app for Android

flutter\_share: ^2.0.0

image\_picker: ^0.8.6+1

# AstroLogic App Setup Document

image_cropper: ^3.0.0	To crop image on iOS and Android		
cached_network_image: ^3.2.2	To show images from the internet		
flutter_ringtone_player: ^3.2.0	To play ringtone		
marquee: ^2.2.3	Varquee is a flutter widget that scrolls infinitely. It pauses after each round of scroll and has features like duration, and curve and s highly customizable.		
audioplayers: ^1.1.1	Used for pausing or resuming the audio file that is already being played		
gallery_saver: ^2.3.2	Download the image to the gallery		
translator: ^0.1.7	To Translate text		
animated_bottom_navigation_bar: ^1.2.0	For the Bottom Navigation Bar theme		
shimmer: ^3.0.0	For text loading simmer		
dots_indicator: ^3.0.0	For show dots		
readmore: ^2.2.0	To read more text		

# 5. Setup Partner App (Technology Flutter)

- a. Initial steps to set up and run mobile app
  - i. Open the App folder in the VSCode
  - ii. Run the following commands in the VSCode Terminal

## flutter clean flutter pub get

- iii. Additional steps to set up for iOS (You can skip these steps if you don't want to set up for iOS)
  - 1. In the VSCode terminal, go to the ios directory

(using the command **cd ios**)

2. Run the following command to install pods

#### pod install

iv. Connect your Android or iOS device with your machine

(To run on an Apple device, you must have an Apple computer)

v. Run the following command to run on an Android or iOS device

#### flutter run

vi. To upgrade the Flutter version run the following command in the VSCode Terminal

(Only if your Flutter version is lower than mentioned in this document)

#### flutter upgrade

b. Change API base URL

After the setup of your API and Admin panel, you have to change your API base URL for that, go to the file located at **lib\utils\global.dart** 



- c. Change App Name
  - i. Change the app name in the Android App
    - 1. Change the app name in the file located at android/app/src/main/AndoidManifest.xml



- ii. Change the app name in the iOS App
  - 1. In VSCode
    - a. Go to ios/Runner/info.plist
    - b. Change string of key CFBundleDisplayName

<key>CFBundleDisplayName</key>
<pre><string>AstroLogic Partner</string></pre>
<key>CFBundleExecutable</key>

- 2. In XCode
  - a. Right-click on the iOS folder and Choose Open in Xcode Option
  - b. Click on the folder icon left side of the XCode window
  - c. Select Runner.
  - d. Select Target runner
  - e. From the General Tab Go to the identity
  - f. Change Display Name

√ Identity	App Category	None	\$	1
	Display Name	AstroLogic Partner	+	
	Bundle Identifier	com.example.astrologicpartner	÷	
	Version	1.0.0	+	
	Build	1	+	

d. Change Package Name/Bundle ID

An app's package name is a unique identifier that is automatically created when you create an app. The term used for iOS apps is "**bundle ID**" and for Android apps, it is "**package name**".

- i. Set Package Name for Android App
  - 1. Change the package name in the file located at android/app/src/main/AndoidManifest.xml

2. Change the package name in the file located at android/app/src/debug/AndoidManifest.xml

```
manifest xmlns:android="http://schemas.android.com/apk/res/android"
package="com.example.astrologicpartner"
```

3. Change Package Name in file which is located at android/app/src/Profile/AndoidManifest.xml

manifest xmlns:android="http://schemas.android.com/apk/res/android"
package="com.example.astrologicpartner"
<!-- The INTERNET permission is required for development. Specifically,</pre>

4. Change the Package Name in the file which is located at android/app/build.gradle

| defaultConfig {   |
|---|
| // TODO: Specify your own unique Application ID (htt            |
| applicationId "com.example.astrologicpartner"                   |
| // You can update the following values to match your            |
| <pre>// For more information, see: https://docs.flutter.d</pre> |
| minSdkVersion 21  |
| targetSdkVersion 31   |
| <pre>versionCode flutterVersionCode.toInteger()</pre>           |

5. Change the folder structure for the below path as per your package name.

android\app\src\main\java\<mark>com\example\astrologicpartner</mark>\

 Change Package Name in file which is located at android\app\src\main\java\com\example\astrologicpartner\Main Activity.java



- ii. Set Bundle ID for iOS App
  - 1. In VSCode
    - a. Go to ios/Runner/info.plist
    - b. Change the string of key CFBundleldentifier



- 2. In XCode
  - a. Right-click on the **iOS** folder and Choose Open in Xcode Option
  - b. Click on the folder icon on the left side of the XCode window
  - c. Select Runner.

- d. Select Target runner
- e. Go to identity
- f. Change Bundle Identifier

| <ul> <li>✓ Identity</li> </ul> |                   |                               |   |  |
|--------------------------------|-------------------|-------------------------------|---|--|
|                                | App Category      | None                          | ٥ |  |
|                                | Display Name      | AstroLogic Partner            | + |  |
|                                | Bundle Identifier | com.example.astrologicpartner | → |  |
|                                | Version           | 1.0.0                         | + |  |
|                                | Build             | 1                             | + |  |

- g. In Signing & Capabilities Go to Signing
- h. Change Bundle Identifier

| Ge       | neral Signing & Capabilities Resource Tags Info Build Settings Build Phases Build Rules |
|----------|---|
| PROJECT  | + Capability All Debug Release Profile  |
| 🛃 Runner | ✓ Signing   |
|          | ✓ Automatically manage signing  |
| TARGETS  | Xcode will create and update profiles, app IDs, and                                     |
| Runner   | Team None C   |
|          | Bundle Identifier com.example.astrologicpartner   |
|          | ✓ i0s   |
|          | Provisioning Profile Xcode Managed Profile  |
|          | Signing Certificate Apple Development   |
|          | Status A. Signing for "Runner" requires a development                                   |
|          | team.   |
|          | Capabilities editor.  |
|          |   |

- e. Create and set Keystore file for Android
  - i. Create a keystore.jks file if it does not exist using the below command in the terminal

keytool -genkey -v -keystore "path\keystore.jks" -storetype JKS -keyalg RSA -keysize 2048 -validity 10000 -alias keystore

ii. Fill in all the details asked while executing the above command

- iii. Recommended. After creating your keystore.jks file, please put it in the **android/app** folder
- iv. Create a key.properties file in the **Android** folder and add the details in the file as per the below screenshot.



#### NOTE:

- If you have changed any default value for any of these keys (storePassword, keyPassword, keyAlias, storeFile) while creating the keystore.jks file, then please also change them to the key.properties file.
- If you place your keystore.jks file somewhere else in the project than mentioned in step 5.c.iii then please change storeFile key value accordingly.
- For more details please refer to this link
- f. Create Firebase Account & Project

In this project, we are using the following Firebase services.

- i. Push Notification
- ii. Phone Authentication
- iii. Firebase Analytics
- iv. Firebase Firestore
- v. Firebase Dynamic Links

For this, you need a Firebase account and a project set up in the Firebase. Please follow the below steps for this,

- i. Go to the Firebase console
- Sign up if you don't have a Google Account or want to create a new account for your project. Otherwise, sign in with your Google Account.

- iii. Click on Add Project
- iv. Enter your project name



#### v. Select Default Account for Firebase

(or you can create a new account)



- vi. Create project
- g. Set up Android App in Firebase Project
  - i. Go to Firebase console
  - ii. Select the project you created in step 5.d.vi
  - iii. Go to Project Setting
  - iv. In the General Tab click on the Add App button
  - v. Select Android
  - vi. Fill out the form and click on the Register App Button

(Please check the below screenshot for reference)

| × A | dd Firebase to your Android app   |
|-----|---|
| 1   | Register app  |
|     | Android package name 🕐  |
|     | com.company.appname   |
|     | App nickname (ontional)   |
|     | My Android App  |
|     | Wy Android App  |
|     | Debug signing certificate SHA-1 (optional) ⑦  |
|     | 00:00:00:00:00:00:00:00:00:00:00:00:00:   |
|     | Required for Dynamic Links, and Google Sign-In or phone number support in Auth.<br>Edit SHA-1s in Settings. |
|     | Register app  |
| 2   | Download and then add config file   |
| 3   | Add Firebase SDK  |
| 4   | Next steps  |

- vii. You need SHA keys (SHA-1 and SHA-256) to add once you create the Android App in the above steps.
  - 1. To Generate debug SHA use the below command

keytool -list -v -keystore "Your directory path\debug.jks" -alias androiddebugkey -storepass android -keypass android

2. To Generate release SHA use the below command

keytool -list -v -keystore "your directory path\keystore.jks" -alias androidreleasekey -storepass your store password -keypass your key password

After generating the debug and release SHA, you have to add them in the Firebase Console where you have created the Android app.

Please check the screenshot below for the reference.

| Certificate fingerprin | t                                       |
|------------------------|---|
| 80:00:00:00:00:        | 00:00:00:00:00:00:00:00:00:00:00:00:00: |
|                        |   |
|                        | Cancel Save                             |

- viii. Download the google-services.json file from Firebase project settings and paste it at the **android/app** location.
- ix. Add Firebase SDK Add the plugin as a build script dependency to your project-level build.gradle file:

```
buildscript {
  repositories {
    // Make sure that you have the following two repositories
    google() // Google's Maven repository
   mavenCentral() // Maven Central repository
  }
  dependencies {
   // Add the dependency for the Google services Gradle plugin
   classpath 'com.google.gms:google-services:4.3.15'
  }
}
allprojects {
  . . .
  repositories {
   // Make sure that you have the following two repositories
   google() // Google's Maven repository
   mavenCentral() // Maven Central repository
  }
}
```

x. Then, in your module (app-level) build.gradle file, add both the google-services plugin and any Firebase SDKs that you want to use in your app:



## h. To Set Up Dynamic Links

i. In the Firebase console click on "Get Started" or "Set Up" to start the process of setting up Dynamic Links for your project.

| Dynamic Links  |   |                           |  |
|--|---|---------------------------|--|
| Send users to the right place in your app<br>whether or not it is already installed  |   |                           |  |
| By continuing below, you understand that your Android/Apple<br>app info (e.g. package name, bundle ID, and App ID Prefix) will<br>be publicly visible in asset/association files.  |   |                           |  |
| Add URL prefix   |   |                           |  |
|  |   |                           |  |
| 1 Create 2 Configure 3 Verify  | 4 Finish  |                           |  |
| 1 Create       2 Configure       3 Verify         Brand your links completely with a URL prefix made up of a cryourdomain.com) and optional path (ex: yourdomain.com/my professional and contextual, resulting in more clicks on links.         Domain ③ | 4 Finish<br>ustom Dynamic Links don<br>ypath). Customized dor | iajn foi.<br>Is fook more |  |
| 1 Create       2 Configure       3 Verify         Brand your links completely with a URL prefix made up of a cryourdomain.com) and optional path (ex: yourdomain.com/my professional and contextual, resulting in more clicks on links.         Domain ③ | 4 Finish<br>ustom Dynamic Links don<br>ypath). Customized dor | iain foit<br>e look more  |  |

ii. After you create Dynamic Links, you need to set up your app codeslib\utils\global.dart



## In Xcode

• Add associated domain capabilities and set the domain



## i. Setup Firebase iOS App

- i. Go to the Firebase console
- ii. Select the project you created in step 5.d.vi
- iii. Go to Project Setting
- iv. In the General Tab click on the Add App button
- v. Select iOS
- vi. Fill out the form and click on the **Register App** Button

(Please check the below screenshot for reference)

| ×A | dd Firebase to your Apple app |
|----|-------------------------------|
| 1  | Register app                  |
|    | Apple bundle ID 🕥             |
|    | com.company.appname           |
|    | App nickname (optional) 💿     |
|    | My Apple app                  |
|    | App Store ID (optional) ③     |
|    | 123456789                     |
|    | Register app                  |
| 2  | Download config file          |
|    |                               |
| 3  | Add Firebase SDK              |
| 4  | Add initialisation code       |
| 4  |                               |
| 5  | Next steps                    |

- vii. Download the GoogleService-info.plist file from Firebase project settings and paste it at the **ios/Runner** location in the app
- viii. XCode Project Setting

This step covers reviewing the most important settings in the XCode workspace. For detailed procedures and descriptions, see <u>Prepare for App Distribution</u>

- 1. Navigate to your target's settings in XCode:
  - a. Open the default Xcode workspace in your project by running the below command in a terminal window from your Flutter project directory.

#### open ios/Runner.xcworkspace

- b. To view your app's settings, select the Runner target in the Xcode navigator.
- 2. Verify the most important settings
  - a. In the Identity section of the General tab
    - i. **Display Name** (The display name of your app.)
    - ii. **Bundle Identifier** (The App ID you registered on App Store Connect.)
  - b. In the Signing & Capabilities tab
    - iii. Automatically manage signing (Xcode should automatically manage app signing and provisioning. This is set true by default, which should be sufficient for most apps. For more complex scenarios, see the <u>Code Signing Guide</u>)
  - Team (Select the team associated with your registered Apple Developer account. If required, select Add Account..., then update this setting.)

| Gen      | eral Signing & Capabilities Resource lags Info Build Settings Build Phases Build Rules |
|----------|--|
| PROJECT  | + Capability All Debug Release Profile   |
| 🛃 Runner | ✓ Signing  |
|          | ✓ Automatically manage signing   |
| TARGETS  | Xcode will create and update profiles, app IDs, and                                    |
| 🛑 Runner | Team None C  |
|          | Bundle Identifier com.example.astrologicpartner  |

- c. In the deployment section of the build settings tab:
  - i. iOS Deployment Target
    - 1. The minimum iOS version that the app supports is 11.0.
    - 2. The General tab of your project settings should resemble the following:

|         | General | Signing & Capabilities | Resource Tags      | Info Build Settings                 | Build Phases Build R | ules          |
|---------|---------|------------------------|--------------------|-------------------------------------|----------------------|---------------|
| PROJECT | ~       | Supported Destination  | s                  |                                     |                      |               |
| Runner  |         |                        |                    |                                     |                      |               |
| Kunner  |         | Destination            |                    | SDH                                 | <                    |               |
|         |         | 📋 iPhone               |                    | iOS                                 |                      |               |
| TARGETS |         | + -                    |                    |                                     |                      |               |
| Runner  |         |                        |                    |                                     |                      |               |
|         | ~       | Minimum Deployments    | 3                  |                                     |                      |               |
|         |         |                        | iOS                | 11.0                                |                      | +             |
|         | ~       | Identity               |                    |                                     |                      |               |
|         |         |                        | App Category       | None                                | 5                    | •             |
|         |         |                        | Display Name       | AstroLogic Partner                  |                      | +             |
|         |         |                        | Bundle Identifier  | com.example.astrologic              | cpartner             | $\rightarrow$ |
|         |         |                        | Version            | 1.0.0                               |                      | +             |
|         |         |                        | Build              | 1                                   |                      | +             |
|         | ~       | Deployment Info        |                    |                                     |                      |               |
|         |         |                        | iPhone Orientation | ✓ Portrait                          |                      |               |
|         |         |                        |                    | Upside Down                         |                      |               |
|         |         |                        |                    | <ul> <li>Landscape Left</li> </ul>  |                      |               |
|         |         |                        |                    | <ul> <li>Landscape Right</li> </ul> |                      |               |
|         |         |                        | Status Bar Style   | Default                             | :                    | •             |
|         |         |                        |                    | Hide during application             | on launch            |               |
|         |         |                        |                    |                                     |                      |               |

3. For a detailed overview of app signing, see <u>Create</u>, <u>export</u>, <u>and Delete signing certificates</u>.

## j. Change App Icon

i. For Android

Replace the icons in the **android\app\src\main\res** folder as shown in the below image.



ii. For iOS

1. Replace the icons in the below folder as shown in the below image

ios\Runner\Assets.xcassets\Applcon.appiconset



- 2. Change icons using XCode
  - a. Right-click on the iOS folder Choose Open in Xcode Option
  - b. Click on the folder icon on the left side of the XCode window



c. Select Runner.

- d. Select Target runner
- e. Go to App Icons And Launch Images
- f. Click the right arrow button of the app icon source

|              | ⊞ I < > ⊠ Runner | C Assets  |
|--------------|------------------|---|
| > 🔝 Runner M | 🐱 Runner         |   |
| > 🔝 Pods     | Gene             | ral Signing & Capabilities Resource Tags Info Build Settings Build Phases Build Rules |
|              | PROJECT          | Hide during application launch  |
|              | 🔝 Runner         | V App Icons and Launch Screen   |
|              | TAROFTO          | App Icon AppIcon +  |
|              | Runner           | App Icons Source 🗌 Include all app icon assets  |
|              |                  | Launch Screen File  |
|              |                  | Supported Intents     Class Name Authentication                                       |

g. Replace all the icons according to their size



#### NOTE:

• If you want to generate the App icon bundle from any image you have, you can generate it from publicly available websites like

https://www.appicon.co/

- k. Build Release for Android
  - i. Open Project in VS Code
  - ii. In Terminal Execute the below commands

| flutter clean            |
|--------------------------|
| flutter pub get          |
| flutter build apkrelease |

iii. After making the release, to generate the release bundle Execute the below command

#### flutter build appbundle --release

iv. Get the APK from the below path

#### build\app\outputs\flutter-apk\app-release.apk

- I. Build Release for iOS
  - i. Open Project in XCode
  - ii. Select Archive from the Product Menu



- iii. After successfully archiving select the **Organizer** option from the **Windows menu**
- iv. After clicking on it opens one popup for Archive, Click on the **Distribute App** Button

| Ś   | Xcode File E                   | dit View Find Navigate   | Editor Product Debug | g Integrate Window | Help              |         | zoom 🗊 ᅙ               |
|-----|--------------------------------|--------------------------|----------------------|--------------------|-------------------|---------|------------------------|
| ۰ ک | •• •                           | •••                      | Archives             |                    |                   |         |                        |
| r 🖻 | $\boxtimes \square \bigcirc 4$ | 🔴 Runner (iOS App, com ᅌ | Name                 | Creation Date      | v Ve              | rsion   |                        |
|     | from '[[Sti<br>'Any'           | Products                 | 🔼 Runner             | 14-Dec-202         | 23 at 3:11 PM 1.0 | 0.0 (1) | Distribute App er      |
| 1   | > 🛆 Expression                 | Archives                 |                      |                    |                   |         | Validate App           |
| 9   | from '[[Sti<br>'Any'           | Reports                  |                      |                    |                   |         |                        |
| C   | > 🛆 Expression                 | A Crashes                |                      |                    |                   |         | Details                |
| ÷   | from '[[Sti<br>'Any'           |                          |                      |                    |                   |         | Version 1.0.0 (1)      |
| ĩ   | > 🛆 Expression                 | Disk writes              |                      |                    |                   |         | Type iOS App Archive   |
| E   | from '[[Sti<br>'Any'           | Energy                   |                      |                    |                   |         | Team Native Software   |
|     | > 🛆 Expression                 | - Feedback               |                      |                    |                   |         | Architectures arm64    |
| Ļ   | from '[[Sti<br>'Any'           | Hangs                    |                      |                    |                   |         | Download Debug Symbols |
|     | > 🛆 Expression                 | Insights                 |                      |                    |                   |         | C C                    |
| 2   | from '[[Sti<br>'Any'           | n Regressions            |                      |                    |                   |         | Description            |
| 2 ~ | AgoralrisRTC_i                 |                          |                      |                    |                   |         | In                     |
|     | 🛆 The iOS dep                  | Metrics                  |                      |                    |                   |         | No Recorded on         |
| 4   | RGET' is set                   | D Battery Usage          |                      |                    |                   |         | No Description         |
| ~   | AgoraRtcEngin                  | 😑 Disk Writes            |                      |                    |                   |         |                        |
| 1   | A The iOS dep                  | Hang Rate                |                      |                    |                   |         |                        |
|     | RGET' is set                   | 🤣 Launch Time            |                      |                    |                   |         |                        |
| ~   | AgoraRtm_iOS                   | C Memory                 |                      |                    |                   |         |                        |

- v. After successfully done, you can upload this app to your Apple developer account in the TestFlight
- vi. To publish your app from TestFlight please follow this link

## m. Other Options for the Advanced User

| Images                              | Path   | Screen Path   |
|-------------------------------------|--|---|
| Splash screen                       | assets/images/splash.png                                 | lib/views/splash/splashScreen.dart  |
| Login Screen                        | assets/images/splash.png<br>assets/images/arrow_left.png | lib/views/Authentication/loginScreen.dart   |
| Success Registration<br>Screen      | assets/images/thank_you_image.<br>png                    | lib/views/Authentication/successRegistrationScre<br>en.dart                               |
| Chat Screen                         | assets/images/chat_background.<br>png                    | lib/views/HomeScreen/chatScreen.dart  |
| Home Screen                         | assets/images/translation.png                            | lib/views/HomeScreen/homeScreen.dart  |
| Search Blog Screen                  | assets/images/2022Image.png                              | lib/views/HomeScreen/searchBlogScreen.dart  |
| Assistant Chat<br>Screen            | assets/images/chat_background.<br>png                    | lib\views\HomeScreen\Assistant\assistantChatSc<br>reen.dart                               |
| Blog Astrology Detail<br>Screen     | assets/images/2022Image.png                              | lib\views\HomeScreen\FloatingButton\Astrologer<br>Blog\blogAstrologyDetilScreen.dart      |
| Blog Astrology<br>Screen            | assets/images/2022Image.png                              | lib\views\HomeScreen\FloatingButton\Astrologer<br>Blog\blogAstrologyScreen.dart           |
| Daily Horoscope<br>Detail Container | assets/images/sky.png                                    | lib\views\HomeScreen\FloatingButton\DailyHoro<br>scope\dailyHoroScopeDetailContainer.dart |

i. Paths to the images used in the app

| Kundli Detail screen             | assets/images/whatsapp.png  | lib\views\HomeScreen\FloatingButton\FreeKundli<br>\kundliDetailsScreen.dart         |
|----------------------------------|---|---|
| Kundli Matching<br>Result Screen | assets/images/night_star.png<br>assets/images/splash.png<br>assets/images/couple_ring_imag<br>e.png<br>assets/images/couple_image.png                   | lib\views\HomeScreen\FloatingButton\KundliMat<br>ching\kundliMatchResultScreen.dart |
| Live Screen                      | assets/images/voice.gif   | lib\views\HomeScreen\live\liveScreen.dart   |
| Report History<br>Screen         | assets/images/2022Image.png   | lib\views\HomeScreen\ReportModule\reportHisto<br>ryListScreen.dart                  |
| Report Request<br>Screen         | assets/images/2022Image.png   | lib\views\HomeScreen\ReportModule\reportRequ<br>estHomeScreen.dart                  |
| Floating Action<br>Button Widget | assets/images/daily_horoscope.p<br>ng<br>assets/images/free_kundli.png<br>assets/images/kundli_matching.p<br>ng<br>assets/images/astrology_blog.pn<br>g | lib\widgets\floatingActionButton.dart   |
| Create New Kundli                | assets/images/maleGender.png<br>assets/images/femaleGender.png<br>assets/images/otherGender.png   | lib\views\HomeScreen\FloatingButton\FreeKundli<br>\newKundliCreate.dart             |

ii. Fonts used in the app. If you want to change, you can make the changes in the **pubspec.yaml** file and the **Assets** folder.

| Poppins | asset: assets/fonts/Poppins-Black.ttf<br>asset: assets/fonts/Poppins-Bold.ttf<br>asset: assets/fonts/Poppins-Medium.ttf<br>asset: assets/fonts/Poppins-Regular.ttf<br>asset: assets/fonts/Poppins-SemiBold.ttf<br>asset: assets/fonts/Poppins-Thin.ttf<br>asset: assets/fonts/Poppins-ExtraBold.ttf<br>asset: assets/fonts/Poppins-ExtraLight.ttf<br>asset: assets/fonts/Poppins-Light.ttf |
|---------|--|
|---------|--|

iii. Colors used in the app. If you want to change the colors you can make the changes in the file **lib/Theme/nativeTheme.dart** 

| #                        | Color code |
|--------------------------|------------|
| Primary color            | DB4627     |
| primaryColorLight        | DB4627     |
| primarylconTheme - color | black      |

| Text button - background color   | black  |
|----------------------------------|--------|
| Text button - foreground color   | DB4627 |
| Divider theme - color            | gray   |
| Card - color                     | white  |
| TextFormfield - border color     | DB4627 |
| Appbar theme - color             | DB4627 |
| Appbar icon theme- color         | black  |
| Drawer icon - color              | gray   |
| Checkbox - check color           | black  |
| Checkbox - fillcolor             | DB4627 |
| scaffoldBackgroundColor          | white  |
| elevatedButton -background color | DB4627 |
| elevatedButton -foreground color | F5F5F5 |
| Dialog -background color         | white  |

iv. Packages used in the app are listed below. You can find them in **pubspec.yaml** file.

| Package Name - Version            | Description   |
|-----------------------------------|---|
| cupertino_icons: ^1.0.2           | Use for icon  |
| flutter_otp_text_field: ^1.1.1    | For otp text field  |
| percent_indicator: ^4.2.3         | For circular percent indicator  |
| Font_awesome_flutter :^ 10.6.0    | To access icon of font awesome  |
| date_format: 2.0.7                | For date format   |
| dotted_border: ^2.1.0             | For dotted border   |
| Flutter_rating_bar :^ 4.0.1       | For the star Rating bar   |
| flutter_document_picker: ^5.2.3   | For pick .pdf, .doc, .docx file   |
| syncfusion_flutter_pdfviewer: any | For pdf view and download   |
| simple_speed_dial: ^0.1.7         | Add an infinity number of menu items with or without label next to an icon  |
| intl: ^0.18.1                     | To Provide internationalization and localization facilities, including message translation, plurals and genders, date/number formatting |

|                                      | and parsing, and bidirectional text.  |
|--------------------------------------|---|
| custom_clippers: ^2.0.0              | For custom shape  |
| flutter_holo_date_picker: ^1.1.2+1   | For Date Picker   |
| flutter_time_picker_spinner: ^2.0.0  | For time Picker   |
| fluttertoast: ^8.2.2                 | Use to show a toast message   |
| flutter_local_notifications: ^16.1.0 | A cross-platform plugin for displaying local notifications.   |
| google_place: ^0.4.7                 | Use for the search place name   |
| flutter_share: ^2.0.0                | Share messages, links, or files from your Flutter app for Android and IOS   |
| geocoding: ^2.1.1                    | To retrieve the complete address with the help of Latitude and Longitude and vice-versa   |
| firebase_dynamic_links: ^5.3.7       | To send existing and potential users to any location within your iOS or Android app   |
| screenshot: ^2.1.0                   | Capture any Widget as an image  |
| dio: ^4.0.6                          | A powerful HTTP client for Dart, which supports Interceptors, Global configuration, FormData, Request Cancellation, File downloading, Timeout, etc. |
| open_filex: ^4.3.4                   | To open file  |
| share_plus: ^7.1.0                   | To share file   |
| webview_flutter: ^3.0.4              | Used to display web pages   |
| connectivity_plus: ^4.0.2            | To check the connectivity of the device   |
| http: any                            | For consuming HTTP resources  |
| get: ^4.6.6                          | State management  |
| get_storage: ^2.1.1                  | For local storage   |
| shared_preferences: ^2.2.2           | To store something locally  |
| firebase_messaging: ^14.6.9          | To use Firebase cloud messaging API   |
| cloud_firestore: ^4.9.3              | Cloud Firestore, a cloud-hosted, NoSQL database with live synchronization and offline support on Android and iOS                                    |
| firebase_core: ^2.17.0               | To use the Firebase Core API, which enables connecting to multiple Firebase apps.   |
| firebase_auth: ^4.10.1               | For Authentication using Firebase authentication options  |
| device_info_plus: ^9.0.3             | To get device info  |
| package_info_plus: ^4.1.0            | Get the version of the app  |

# AstroLogic App Setup Document

| path_provider: ^2.1.11                    | For finding commonly used locations on the filesystem   |
|---|---|
| image_picker: ^1.0.4                      | For picking an image  |
| image_cropper: ^4.0.1                     | To crop image on iOS and Android  |
| cached_network_image: ^3.3.0              | To show images from the internet  |
| flutter_countdown_timer: ^4.1.0           | Shows remaining time  |
| audioplayers: ^5.2.0                      | Used for pausing or resuming the audio file that is already being played  |
| marquee: ^2.2.3                           | Marquee is a flutter widget that scrolls infinitely. It pauses after each round of scroll and has features like duration, and curve and is highly customizable. |
| agora_rtm: ^1.5.8                         | For live chat and join live stream message  |
| agora_rtc_engine: ^6.2.2                  | For audio calls and live streaming video call   |
| permission_handler: ^11.0.1               | This plugin provides a cross-platform api to request and check permission   |
| translator: ^0.1.7                        | To Translate text   |
| animated_bottom_navigation_bar:<br>^1.3.0 | For the Bottom Navigation Bar theme   |
| Pdf: ^3.10.4                              | For PDF   |
| shimmer: ^3.0.0                           | For text loading simmer   |
| video_player: ^2.4.7                      | For play video  |

#### **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
- For more information on iOS refer to this link

This document was last updated on 06 December 2023.