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

- Flutter with Dart Language for Mobile App
- DB Browser for SQLite for Database

Please follow the below steps to set up the App.

(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 and set up Flutter from this link
- b. Install DB Browser for SQLite from this link
- 2. Setup Mobile 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. Changes for About screen

You have to change your about us screen URLs, for that go to the file located at **lib\screens\aboutScreen.dart**

- 1. Search <YOUR EMAIL> and replace it with your email
- 2. Search <YOUR WEBSITE> and replace it with your website
- Search <YOUR FB PAGE URL> and replace it with your facebook page URL
- Search <YOUR INSTAGRAM PAGE URL> and replace it with your instagram page URL
- Search <YOUR LINKEDIN PAGE URL> and replace it with your linkedin page URL
- Search <YOUR TWITTER PAGE URL> and replace it with your twitter page URL
- 7. Search <YOUR WEB SITE URL> and replace it with your website URL
- c. Changes for App store and Play store

You have to change your app store and play store URLS, for that go to the file located at **lib\utils\global.dart**

- 1. Search <PLAY STORE URL> and replace it with your play store URL
- 2. Search <APP STORE URL> and replace it with your app store URL
- d. Changes for Privacy policy

You have to change your privacy policy URL, for that go to the file located at **lib\screens\privacyPolicyScreen.dart**

- 1. Search <**PRIVACY POLICY URL>** and replace it with your privacy policy URL
- e. 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

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools"
package="com.demo.accountingapp">
```

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



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

android\app\src\main\java\<mark>com\demo\accountingapp</mark>

Ex. If your package name is com.app.accountingapp

android\app\src\main\java\com\app\accountingapp\

 Change Package Name in file which is located at android\app\src\main\java\com\demo\accountingapp\MainActivity.j ava



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

```
<key>CFBundleExecutable</key>
<string>$(EXECUTABLE_NAME)</string>
<key>CFBundleIdentifier</key>
<string>$(PRODUCT_BUNDLE_IDENTIFIER)</string>
<key>CFBundleInfoDictionaryVersion</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. Go to identity

	General Signing & Ca	pabilities Resour	ce Tags Info	Build Settings	Build Phases	Buil
PROJECT	+ Capability All	Debug Release	Profile			
🛃 Runner	✓ Signing					
			Automatically	manage signing	as ann Da and	
TARGETS			certificates.	ate and update prom	es, app iDs, and	
A Runner	J	Team	Native Software)	٥	
		Bundle Identifier	com.demo.acco	untingapp		
	✓ iOS					

f. Change Bundle Identifier

- f. Create and set Keystore file for Android
 - i. Create a keystore.jks file, if not exist, use 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 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 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
- g. Create Firebase Account & Project

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

- i. Push Notification
- ii. Phone Authentication
- iii. Firestore Database
- iv. Firebase Analytics

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 you 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



×	Create a project (Step 2 of 3)		
	Google Analytics is a free and unlimited anal reporting, and more in Firebase Crashlytics, C Config, A/B Testing, and Cloud Functions.	ytics solution that enables targeting, Cloud Messaging, In-App Messaging, Remote	
	Google Analytics enables:		
	👗 A/B testing 💿	🕵 Crash-free users ⊘	
	User segmentation & targeting across ⑦	Event-based Cloud Functions triggers ③	
	Firebase products	Free unlimited reporting ③	
	Enable Google Analytics for this project Recommended	ot	

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.

vi. Create project

- h. 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)

× Ad	dd Firebase to your Android app
1	Register app
	Android package name 🕐
	com.company.appname
	App nickname (optional) ③
	My 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. 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\keys	store.jks"	-alias
android	releas	sek	ey -storep	ass yo	ur store	password	-keypass	s your
key pase	sword	d						

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.

ertificate fingerprint				
00:00:00:00:00:00	0:00:00:00:00:00:00	:00:00:00:00:00:00:00	00:00:00:0 SHA1	SHA256
			Cancel	Save

- viii. Download the google-services.json file from Firebase project settings and paste it at **android/app** location.
- ix. Setup Authentication
 - Enable Sign in methods:
 - a. In the Firebase console's **Authentication** section, open the <u>Sign in method</u> page.
 - b. From the **Sign in method** page, enable the methods which are shown in the image.

Sign-in providers		
		Add new provider
Provider	Status	
C Phone	C Enabled	
G Google	Enabled	
F Facebook	Enabled	
Apple	Senabled	

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
C	Register app
	Apple bundle ID 💿
	com.company.appname
	App nickname (optional) ⑦ My Apple app
	App Store ID (optional) ③
	Register app
2	Download config file
3	Add Firebase SDK
4	Add initialisation code
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. Go to the **ios\Runner\AppDelegate.m** file and replace "YOUR-API-KEY" with your Api key.

```
- (BOOL)application:(UIApplication *)application
didEinishLaunchingWithOptions:(NSDictionary *)launchOptions {
 [GMSServices provideAPIKey:@"YOUR-API-KEY"];
 [GeneratedPluginRegistrant registerWithRegistry:self];
 // Override point for customization after application launch.
```

ix. Replace "YOUR-REVERSED-CLIENT-ID" with your reversed client ID. You can find this Id from the GoogleService-info.plist file you added from step **5.g.vii**.

```
<string>Editor</string>
<key>CFBundleURLName</key>
<string>$(REVERSED_CLIENT_ID)</string>
<key>CFBundleURLSchemes</key>
<array>
</string>YOUR-REVERSED-CLIENT-ID</string>
</array>
</dict>
</dict>
```

x. 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</u> <u>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. Select the Runner target in the Xcode navigator to view your app's settings.
- 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
 - i. 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.)

	General	Signing & Capabilities	Resour	ce Tags	Info	Build Settings	Build Phases	Build Rules
PROJECT	+ Capa	ability All Debug	Release	Profile				
🛃 Runner	~ s	igning						
TARGETS				Autom Xcode certifie	atically will crea cates.	manage signing te and update profil	es, app IDs, and	
A Runner			Team	Native S	oftware		\$	
		Bundle	e Identifier	com.der	no.acco	untingapp		
		√ 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:

ROJECT	v Supp	orted Destinations				
Runner		Destination		SD	<	
		iPhone		iOS		
ARGETS						
Runner		T				
	✓ Minim	um Deployments				
			iOS	12.1	(• +
	√ Identi	ty				
		-	App Category	None		٥
			Display Name	Accounting App		+
			Bundle Identifier	com.demo.accountinga	qq	\rightarrow
			Version	1		+
			Build	1		+
	√ Deplo	yment Info				
			iPhone Orientation	✓ Portrait		
				Upside Down		
				✓ Landscape Left		
				 Landscape Right 		
			Status Bar Style	Default Hide during applicati 	on launch	•

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

v2.1

xi. Upload your APNs authentication key

If you don't already have an APNs authentication key, make sure to create one.

1. Go to https://developer.apple.com and click Account



2. Log in with your Apple Developer account

Sign in to Apple Developer	
Remember me	
Forgot Apple ID or password? a Don't have an Apple ID? Create yours now. a	

3. Click on Certificates, IDs & Profiles

Cortificatos Identifiars 8	
Cartificatos Idantifiara 8	
Certificates, identifiers a	Additional resources
Profiles	Download beta software, and view and
Manage the certificates, identifiers, profiles,	manage your usage of developer services.
and devices required to develop, test, and distribute apps.	Software Downloads
	Feedback Assistant
	A Xcode Cloud
1 Identifiers	CloudKit
Devices	① MapKit JS
Profiles	💣 WeatherKit
	Profiles Manage the certificates, identifiers, profiles, and devices required to develop, test, and distribute apps. Certificates Identifiers Devices Profiles Keys

4. Click on Keys and add a new key by clicking on the blue plus (+) icon next to the title Keys.

Certificates	Keys 😨	Q
Identifiers Devices		
Profiles	Getting Started with Keys	
Keys	Creating a key allows you to configure, authenticate, and use one or more Apple services for that key. Unlike certificates, keys do not expire and can be modified to access more services after their creation. For more information on creating and using keys, refer to Developer Account	
More	Help.	
	Create a key	

5. On the next page, enter 'APNS Key' in the Key Name field and click the checkbox to enable Apple Push Notifications service (APNs), Device Check and Sign in with Apple. Also configure the Sign in with Apple.

Develope	er		Publishing Bui
ertifi	icates, Identifiers &	Profiles	
< All Keys			
Regis	ster a New Key		Continue
Key Name APNS & You canno	e Key ot use special characters such as @, &, *, ', '', -, .		
ENABLE	NAME	DESCRIPTION	
	Apple Push Notifications service (APNs)	Establish connectivity between your notification server and the Apple Push Notification service. One key is used for all of your apps. Learn more	
	DeviceCheck	Access the DeviceCheck and AppAttest APIs to get data that your associated server can use in its business logic to protect your business while maintaining user privacy. Learn more	
	MapKit JS	Use Apple Maps on your websites. Show a map, display search results, provide directions, and more. Learn more ① There are no identifiers available that can be associated with the key	Configure
	Media Services (MusicKit, ShazamKit)	Access the Apple Music catalog and make personalized requests for authorized users, and check audio signatures against the Shazam music catalog. O There are no identifiers available that can be associated with the key	Configure
	Sign in with Apple	Enable your apps to allow users to authenticate in your application with their Apple ID. Configuration is required to enable this feature.	Configure
	orgin in which Apple	① There are no identifiers available that can be associated with the key	

6. Click Register

CAll Keys	ter a New Key		Back
Key Name APNS Key			
ENABLE	NAME	DESCRIPTION	
×	Apple Push Notifications service (APNs)	Establish connectivity between your notification server and the Apple Push Notification service. One key is used for all of your apps.	

7. Click Download. This will download the APNs key that you will upload to Firebase. Please keep this page open to obtain the Key ID and Team ID for Firebase.

NOTE: Once the key has been downloaded, it cannot be retrieved again.

All Keys	
Download Your Key	Download Done
After downloading your key, it cannot be re-downloaded as the server copy is removed. If you are not prepared to download your key at this time, click Done and download it at a later time. Be sure to save a backup of your key in a secure place.	
lame: APNS Key	
ley ID: 36AZ4Y87HH	

8. Now that you have the APNS key downloaded, you will need to upload this to Firebase. Open up a new browser tab or window and navigate to <u>https://console.firebase.google.com/</u>

NOTE: The next few steps require you to have an iOS Firebase Certificate. If you have not done this yet, please check out our <u>iOS Firebase Certificate</u> article before continuing.

🖕 Firebase	Buildfire App 👻 Go to docs 🌲 🌍
♠ Project Overview	Project settings 0
Build Authentication, Firestore Database,	General Cloud Messaging Integrations Service accounts Data privacy Users and permissions App Check (BITA)
Release & Monitor Crashlytics, Performance, Test Lab,	Project credentials Upload APNs auth key × Add server key
Analytics Dashboard, Realtime, Events, Conve	Key APNs auth key ③
Engage Predictions, A/B Testing, Cloud Mes	Server Key AuthKey_UN823KU9WJ.p8 Browse
	365502819897 Key ID (required) ③
	Apple app configura C7X6UZZ8R8
	Apple apps Team ID (required) ⑦ an APNs authentication key or APNs certificate to connect (105+) com.ffeca N4H76AD88C
	Team ID detected for this app. You can edit this in <u>General settings</u> .
	Cancel Upload is recommended as they are the more current ons to Apple devices
	File Key ID Team ID
Extensions	No APNs auth key Upload
Spark Upgrade Free \$0/month	ADNo Costification

9. Click on your App project

🖉 Firebase			
	Recent projects		[]
	+ Add project	AstroLogic astrologic-6ec7c	Matrimonial App matrimonial-app-cdc42
	Explore a demo project	≧ i05+	≝ i05-
	Credit App credit-app-9be53	FreshSell freshsell-india	

10. On the top left, click on the gear icon on the right side of Project Overview and select Project Settings

😕 F	irebase		Matrimonial App 👻	
n P	roject Overview	\$	Project settings	Matrim
Proiect s	shortcuts		Users and permissions	3 apps
 A	uthentication		Usage and billing	
😤 F	irestore Database			
Ø A	pp Check			
s s	torage			Analytics
🛇 н	losting			Daily active us
G №	lessaging			6 -81.3%
What's n	ew			56
🕹 E	xtensions NEW			1
Product	categories			28
Build		~		
Releas	se & Monitor	~		0 Feb 2
Analyt	tics	~		

11. Click on the Cloud Messaging tab and in the Apple app configuration section, click Upload

- 🖊	Firebase		Matrimonia	al App 👻				
A	Project Overview	\$	Proje	ect setting	js			
Proje	ct shortcuts		General	Cloud Messaging	Integrations	Service accounts	Data privacy	Users and permissions
:: #	Authentication	0						
ŝ	Firestore Database							
\bigcirc	App Check					Firebase Cloud	Messaging API	(V1) 🕑 Enabled
	Storage					Recommended for	most use cases. <u>Le</u>	arn more 🛛
\bigcirc	Hosting					Sender ID		Service Acc

Apple apps	Firebase Cloud Messa with APNs	ging can use either an APNs au	thentication key or APNs	certificate to connect
	APNs Authentication Key Configuration with auth keys is recommended as they are the more current method for sending notifications to Apple devices			
	File	Key ID	Team ID	Upload
	No APNs auth key			Upload

- 12. Here you will upload the key file by clicking Browse. Select the file that ends with .p8 that was downloaded in the previous steps. The file name will look like this: AuthKey_UN823KU9WJ.p8
- 13. Now you will have to copy the Key ID and Team ID by going back to your Apple Developer account. The Key ID is located below the key name and the Team ID is located in the top right corner, next to your Apple developer name.

Developer	-	Team ID
ertifica	tes, Identifiers & Profiles	
< All Keys		
Downloa	d Your Key	Download Done
After downl	downloading your key, it cannot be re-downloaded as the server copy is removed. If you are not prep load your key at this time, click Done and download it at a later time. Be sure to save a backup of you	pared to Ir key in a secure
Name: APNs key	/ 	
Services: Apple	Push Notifications service (APNs)	

14. Go back to the Firebase page and copy and paste the Key ID and Team ID. Lastly, click on the Upload button.

	billioneryp
A Project Overview	Project settings @
Build Authentication, Firestore Database,	General Cloud Messaging Integrations Service accounts Data privacy Users and permissions App Check (BETA)
Release & Monitor Crashlytics, Performance, Test Lab,	Project credentials Upload APNs auth key × Add server key
Analytics Dashboard, Realtime, Events, Conve	Key APNs auth key 💮
Engage Predictions, A/B Testing, Cloud Mes	Server key AuthKey_UN823KU9WJ.p8 Browse h6yggGKdFuTsp47-jF50bKDqKzAyaL4Ppwpjfck Sender ID ①
	365502819897 Key ID (required) ⑦
	Apple app configure Apple apps Team ID (required) ③ an APNs authentication key or APNs certificate to connect
	(105- com.ffect Team ID detected for this app. You can edit this in <u>General settings</u> .
	Cancel Upload is recommended as they are the more current ons to Apple devices
	File Key ID Team ID
Stressions	No APNs auth key Upload
Spark Upgrade Free \$0/month	APNs Certificates

j. Configure the Firebase setting to the Project

Go to the lib\firebase_option.dart file

• For Android settings replace your credentials in the android method



• For iOS settings replace your credentials in the ios method

static const FirebaseOptions ios = FirebaseOptions(
 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",
 androidClientId: 'Your Android Client Id',
 iosClientId: 'Your Ios Client Id',
 iosBundleId: 'com.demo.matrimonyapp',
);

- k. 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 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 icons source

> 🔝 Runner M	Kunner Runner			
Pods		General Signing & Capabilities Resource Tags	Info Build Settings Build Phases	Build Rules
	PROJECT	iPhone Orientation Status Bar Style	 ✓ Portrait Upside Down ✓ Landscape Left ✓ Landscape Right Default 	٥
			Hide during application launch	
	A Runner	✓ App Icons and Launch Screen		
		App Icon	Applcon	+
		App Icons Source	Include all app icon assets	
		Launch Screen File		•
		✓ Supported Intents		

Runner	🛛 🛛 Runner 🕽 🚞 Runner 🕽 🖾 Assets) 🖪 Appleon						< 🔺)
> 🔚 Flutter ~ 🔚 Runner Runner	Applcon	Applcon		Δ	A		Δ	App Icon
Assets	м		2x	3x	1x	2x	Зх	
 LaunchScreen Info GeneratedPluginRegistrant GeneratedPluginRegistrant AppDelegate Runner-Bridging-Header Products 	м		iPhone Notificat 20pt	tion		iPhone Settings 29pt		
 Pods Formation 			2x	3х	1	lx 2x		
Pods			iPhone	e Spotlight 40pt		iPhone App iOS 5,6 57pt		
				2x	3x			
				iPh	one App 60pt			
			A	A	A		A	A
		1x	2x	1x	2x		1x	2x
	+ — 🕞 Filter	iPad No	tifications	iPad	Settings		iPad S	potlight

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/

- I. 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

- m. 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
- n. Other Options for the Advanced User
 - i. Paths to the images used in the app

Images	Path	Screen Path		
About Screen	assets/images/logo.png	lib/screen/aboutScreen.da rt		
Splash Screen	assets/images/logo.png	lib/screen/splashScreen.d art		
Backup & Restore Screen	assets/images/googleicon .png	lib/screen/backupAndRest oreScreen.dart		
Lock Screen	assets/images/logo.png	lib/screen/lockScreen.dart		
Other Apps Screen	assets/images/eventdecor 144x144.png	lib/screen/otherAppsScre en.dart		
	assets/images/securewall et144x144.png	lib/screen/otherAppsScre en.dart		
	assets/images/wordpuzzl e128x128.png	lib/screen/otherAppsScre en.dart		
	assets/images/kidsnurser y144x144.png	lib/screen/otherAppsScre en.dart		
	assets/images/yoga4me1 44x144.png	lib/screen/otherAppsScre en.dart		
	assets/images/kidsabcd1 44x144.png	lib/screen/otherAppsScre en.dart		
	assets/images/wishkarts1 50x150.png	lib/screen/otherAppsScre en.dart		
	assets/images/easycount er144x144.png	lib/screen/otherAppsScre en.dart		

ii.

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	0xFF0076bc
Primary Color Dark	0xFF000000
Primary Color Light	0xFF7c7e7d
Primary Icon Theme Color	0xFF0076bc
Canvas Color	white
Scaffold Background Color	white
Card Theme Shadow Color	grey
Appbar theme - color	0xFF0076bc

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

Package Name - Version	Description		
package_info_plus: ^8.0.0	Check <u>here</u>		
webview_flutter: ^4.7.0	Check <u>here</u>		
shared_preferences: ^2.2.3	Check <u>here</u>		
url_launcher: ^6.2.6	Check <u>here</u>		
sqflite_sqlcipher: ^3.0.0	Check <u>here</u>		
material_design_icons_flutter: 7.0.7296	Check <u>here</u>		
http: ^1.2.1	Check <u>here</u>		
email_validator: 2.1.17	Check <u>here</u>		
font_awesome_flutter: ^10.7.0	Check <u>here</u>		
image_cropper: ^6.0.0	Check <u>here</u>		
flutter_mailer: ^2.1.2	Check <u>here</u>		
ribbon_widget: ^1.0.5	Check <u>here</u>		
printing: ^5.12.0	Check <u>here</u>		
page_transition: ^2.1.0	Check <u>here</u>		
number_to_words: 1.0.0	Check <u>here</u>		
mobile_number: 2.1.1	Check <u>here</u>		
local_auth: ^2.2.0	Check <u>here</u>		
open_file: 3.3.2	Check <u>here</u>		
mailer: 6.0.1	Check <u>here</u>		
ulid: 2.0.0	Check <u>here</u>		
badges: 3.1.2	Check <u>here</u>		
encrypt: 5.0.3	Check <u>here</u>		
flutter_local_notifications: ^17.1.2	Check <u>here</u>		
device_info_plus: ^10.1.0	Check <u>here</u>		

fl_chart: ^0.68.0	Check here
flutter_randomcolor: ^1.0.15	Check <u>here</u>
contacts_service: 0.6.3	Check <u>here</u>
multi_image_picker_plus: ^0.0.4	Check <u>here</u>
intro_slider: 4.2.1	Check <u>here</u>
permission_handler: ^11.3.1	Check <u>here</u>
google_sign_in: ^6.2.1	Check <u>here</u>
googleapis: ^13.1.0	Check <u>here</u>
image_picker: ^1.1.1	Check <u>here</u>
share_plus: ^9.0.0	Check <u>here</u>
file_picker: ^8.0.3	Check here
connectivity_plus: ^6.0.3	Check <u>here</u>
intl: ^0.18.0	Check <u>here</u>
path_provider: ^2.1.3	Check <u>here</u>
flutter_inappwebview: ^5.8.0	Check <u>here</u>
background_fetch: ^1.3.4	Check <u>here</u>
in_app_update: ^4.2.2	Check <u>here</u>
archive: ^3.5.1	Check <u>here</u>
pdf: ^3.10.8	Check <u>here</u>
image_picker_for_web: ^3.0.4	Check here
universal_html: 2.2.4	Check <u>here</u>
flutter_html_to_pdf: ^0.7.0	Check here

USEFUL LINKS

• For more information on iOS refer to this link

This document was last updated on 30 May 2024.