Quick User Guide (v1.0.0+1)

Instructions to setup Firebase for the included projects

The following guide explains the steps on how to update and rebuild the included projects for both client app (ClientApp) and admin app (AdminApp) to host them from your own Firebase account.

Step 1: Go to https://console.firebase.google.com to create your Firebase project.

Step 2: Then, navigate to the newly created Firebase project.

Step 3: Go to Firestore Database and enable Cloud Firestore database.

Step 4: Go to **Firestore Database > Rules** and replace the default rule with the following code:

```
rules_version = '2';
service cloud.firestore {
  match /databases/{database}/documents {
    match /{document=**} {
      allow read, write: if request.auth != null;
    }
  }
}
```

Step 5: Next, Go to **Storage > Rules** and substitute with the following new Rule and publish.

```
rules_version = '2';
service firebase.storage {
  match /b/{bucket}/o {
    match /{allPaths=**} {
      allow read, write: if request.auth != null;
    }
  }
}
```

Step 6: Now, extract the included project folder **'JokesApp.zip'** and navigate to **Apps** folder.

Step 7: Copy and move ClientApp project folder to your Flutter development folder. For example: **C:\src\AndroidStudioProjects\<ClientApp>.**

Step 8: Similarly, copy and move AdminApp project folder to **C:\src\AndroidStudioProjects\<AdminApp>**

Step 9: Now, open command prompt and run the following commands:

cd C:\src\AndroidStudioProjects\ClientApp

flutterfire configure

Follow the instructions and select the Firebase project that was created in the first step when prompted to link and use with your ClientApp project.

Step 10: You will see that **firebase_options.dart** configuration file is generated inside **ClientApp/lib/firebase_options.dart**

Step 11: Next, run the following command to download and install all the package dependencies for the Clientapp project:

flutter pub get

Step 12: Similarly, navigate to AdminApp folder by running the following commands like we did in Steps 9, 10 and 11.

cd C:\src\AndroidStudioProjects\AdminApp

flutterfire configure

flutter pub get

Import included Firestore Database Schema

1. In the extracted **JokesApp** app folder, navigate to **Database > Firestore** and locate **all_namespaces_all_kinds.export_metadata** file.
2. Next, go to your Firebase console and upload the file **all_namespaces_all_kinds.export_metadata** to your **Firebase Storage** account.
3. Then, go to https://console.cloud.google.com/firestore/databases/-default-/ and click **Import/Export** menu on the left panel to select and import the above file from your Storage bucket.
4. After successful import, your Firestore Database will show up on your Firestore Database section.

Compiling and building the app using Android Studio

Renaming the Package

To rename the package, create a new flutter project in Android Studio by following the steps below:

1. **File > New > New Flutter Project** and name the project with your preferred package name: for example, **com.example.clientapp**
2. Copy the files (assets, lib, pubspec.yaml) from included (**Clientapp**) project and replace them in your newly created project.
3. Follow the above same steps (1 and 2) as above to create and rename the package name for your Admin app.

Change the App Logo

To change app logo for Client app and Admin app, follow the steps below:

Change logo for ClientApp

Go to newly created flutter project **ClientApp > android > app > src > main >** and replace **ic_launcher-playstore.png** with logo of your own. Similarly, go to **ClientApp > android > app > src > main > res** and replace launcher images (**ic_launcher.png, ic_launcher_foreground.png, ic_launcher_round.png**) in the **mipmaps** folder.

Change logo for AdminApp

Go to newly created flutter project for **AdminApp > android > app > src > main >** replace **ic_launcher-playstore.png** with logo of your own. Similarly, go to **AdminApp > android > app > src > main > res** and replace launcher images (**ic_launcher.png, ic_launcher**

Building the Client App

Step 1: Open Android Studio and click **File > Open** and select the ClientApp project from **C:\src\AndroidStudioProjects\ClientApp**

Step 2: Go to **Build > Flutter > Build APK to build APK** for distribution**.**

Step 3: Navigate to **ClientApp > build > app > outputs > flutter-apk** and locate **app-release.apk**

Step 3: To build app-bundle, Go to **Build > Flutter** and click **Build App** Bundle in Android Studio.

Step 4: Navigate to **ClientApp > build > app > outputs > bundle** > release and locate **app-release.aab**.

Building the Admin App

The procedure to compile and rebuild AdminApp project follows the steps as the ClientApp from Step 1 to Step 4.

Create Admin User

To be able to login to the Admin panel app, you need to add user to your Firebase project in your **Firebase Console** by going to Authentication option and clicking **Sign-in method** tap. Add both **Email/Password** and **Anonymous** providers. After that, go to **Users** tap and click **Add user** button to add the Admin user (For example, admin@example.com) and set password for the user. This new user will now be able to login as Admin in your Admin panel app.

Change AdMob App ID in Client App

1. Navigate to **ClientApp > android > app > src > main > AndroidManifest.xml**.
2. Open the **AndroidManifest.xml** file and locate the following code.

```
<meta-data
   android:name="com.google.android.gms.ads.APPLICATION_ID"
   android:value="ca-app-pub-3940256099942544~3347511713"/>
```

Replace **ca-app-pub-3940256099942544~3347511713** with your own AdMob App ID from your AdMob dashboard.

Next Steps – Managing Content

The following guide explains the steps for managing the content through the Admin app for your user facing app ClientApp:

Manage Category

1. **Create Category**: To create new category, go to **Categories > Add New Category.**
2. **Edit Category**: To edit existing category, go to **Categories >** Click **Edit** icon to update and click **Save** icon to save the changes.
3. **Delete Category**: To delete existing category, go to **Categories >** Click **Delete** icon and confirm to delete the selected category.
4. **Change Category Image**: Go to **Category >** Click **Image** icon to select the new category image.

Manage Culture

1. **Create Culture**: To create new culture, go to **Cultures > Add New Culture**.
2. **Edit Culture**: To edit existing culture, go to **Cultures >** Click **Edit** icon to update and click **Save** icon to save the changes.
3. **Delete Culture**: To delete existing culture, go to **Cultures >** Click **Delete** icon and confirm to delete the selected culture.
4. **Change Culture Image**: Go to **Cultures >** Click **Image** icon to select the new culture image.

Manage Content

1. **Create Content**: To create new content, go to **Content > Add New Content**.
2. **Edit Content**: To edit existing content, go to **Content >** Click **Edit** icon to update and click **Save** icon to save the changes.
3. **Delete Content**: To delete existing content, go to **Content >** Click **Delete** icon and confirm to delete the selected content.

Manage Background/Theme

1. **Add Background/Theme**: To add new background/theme image, go to **Backgrounds > Add New Background Image**.
2. **Delete Background**: To delete existing background/theme, go to **Backgrounds >** Click **Delete** icon and confirm to delete the selected background.

Manage Notification

1. **Send Content Notification**: To send content as notification, go to Content > click **Send** icon to send the selected content as notification.
2. **Send Promo Notification**: To send promotional message notification, click **Send Promo Notification** option on the home screen. Next, enter **Notification Title** and **Notification Message** in the text fields and click **Send Notification** button.

Manage App Settings

1. **Change App Name**: To change App Name, go to **App Settings >** Enter **App Name** in the app name field and click **Save** button to save the changes.
2. **Change Font Family for App Name**: To change font family, go to **App Settings >** Select the font family from the dropdown menu and click **Save** button to save the changes.
3. **Change Font Size for App Name**: To change font size, go to **App Settings** > Set the font size (for example, 18 or 20 or 24, etc) in the font size field and click **Save** button to save the changes.
4. **Add/Update Privacy Policy Page/Terms of Service URLs**: To add/update **Privacy Policy/Terms of Service URLs**, go to **App Settings** > Enter the Privacy Policy and Terms of Service URLs in the corresponding fields and click **Save** button to save the changes.
5. **Configure Server Key for FCM Notification:** Go your **Firebase Console > Project Overview > Project settings > Cloud Messaging > Cloud Messaging API (Legacy)** and copy the Server key. Next open your Admin app and fill in the Server key field with the copied Server key from the Firebase console.
6. Finally, click **Save** to save the changes.

This allows you to utilize Firebase FCM service to send and receive notification through your app.

Manage In-App Purchase

1. **Create Product ID** for your **In-App Purchase**: To enable and create In-app purchase, you need to first upload and publish your app on your Google Play Console. Once you have published your app, create Product ID from your Google Play Console Developer. You can do that by going to **Google Play Console > Select the App > Monetize > In-app Products > Create product**.
2. After you have created the Product ID in the above step, copy the Product ID from the Play Console and then go to your **Admin app > App Settings >** Enter the copied **Product ID** in the In-app Purchase Product ID field and click **Save** button to add and save the changes.

Manage Ads Settings

There are 3 types of AdMob Ad Types integrated in your app; Carousel, Banner and Interstitial ads.

Carousel Ad: Carousel Ad is the ad that is shown in between the content as you scroll through the content list.

Banner Ad: Banner ad is the ad that is shown on the footer section of the app.

Interstitial Ad: Interstitial ad is shown as a full screen ad when you open the app.

To Enable and configure each of the above ad type, follow the steps below:

1. Open your **Admin** app and go to **Ads Settings** and select the **Ad unit**.
2. Next, select the Ad format based on the selected Ad Unit.
3. For **Carousel Ad** unit, choose either **Medium Rectangle, Large Banner or Adaptive Banner**. For **Footer Banner**, choose **Banner** Ad format. Choose **Interstitial** Format for **Interstitial Ad** Unit.
4. Select **AdMob** as the Ad Provider.

5. Fill the Ad Unit ID and choose Enabled and set position to Default.
6. Click **Save** to save the changes.