

## App Name: Rajneesh Assignment

### ❖ User Login Screen:

- Screen Name: login\_screen.dart
- Components:
  - TextFormField for Email
  - TextFromField for password
  - Login Button
  - Don't have an account? Register button
- Form Validation
  - Using From for the validation.
  - Validate if the Email is empty or wrong Email Format.
  - Validate if the Password is blank or the incorrect Password format.
  - Display the error message.
- Actions:
  - On successful login navigate to the Home Screen.
  - Provided a button to navigate to the Registration Screen.

### ❖ User Registration Screen:

- Screen Name: register\_screen.dart
- Components:
  - TextFormField for First Name
  - TextFormField for Last Name
  - TextFormField for Email
  - TextFormField for Password
  - TextFormField for Confirm Password
- Form Validation:
  - Using From for the validation.
  - Validate if the First Name is empty.
  - Validate if the Last Name is empty.
  - Validate if the Email is empty or wrong Email Format.
  - Validate if the Password is blank or the incorrect Password format.
  - Validate if the Confirm Password is blank or the incurent Password format.
  - Validate Password and Confirm Password must be the same.
  - Display the error message
- Actions:
  - On successful registration navigate to the home screen.
  - Provided a button to navigate back to the login screen.

### ❖ Firebase Push Notification:

- Set up Firebase in the Flutter project for the push notification.
- Added google-services.json inside the android/add.

- Installed the necessary Firebase plugins for Flutter such as 'firebase\_core', 'firebase\_messaging', and 'flutter\_local\_notification'.
- Create a Firebase service to handle the push notification.
- Requested permission from the user to receive push notifications.
- Handle incoming push notifications using the onMessage receive method of the FirebaseMessaging.
- Display the received push notification as a system notification using Flutter's native notification APIs and using 'flutter\_local\_notification' in the foreground.

#### ❖ App Flow:

- The user opens the app and is presented with the login screen.
- The user enters their email and password and taps the login button.
- Form validation is performed and if successful the user will redirect to the Home screen.
- If the user doesn't have an account, they can navigate to the registration screen.
- On the registration screen, the user enters their detail and taps the register button.
- Form validation is performed and if successful, User will redirected to the Home screen.
- The app is set up to receive push notifications using the FCM.
- Whenever a push notification is sent to the user, it is handled by the Firebase service and displayed as a system notification using Flutter native notification API (if the app is in the background or the 'flutter\_local\_notificaiton' if the app is running or in the foreground).