

RP Store | React Native App

Libraries/Frameworks Used to achieve MVP app:

- React Native
- Redux
- Firebase

Basic use case and features :

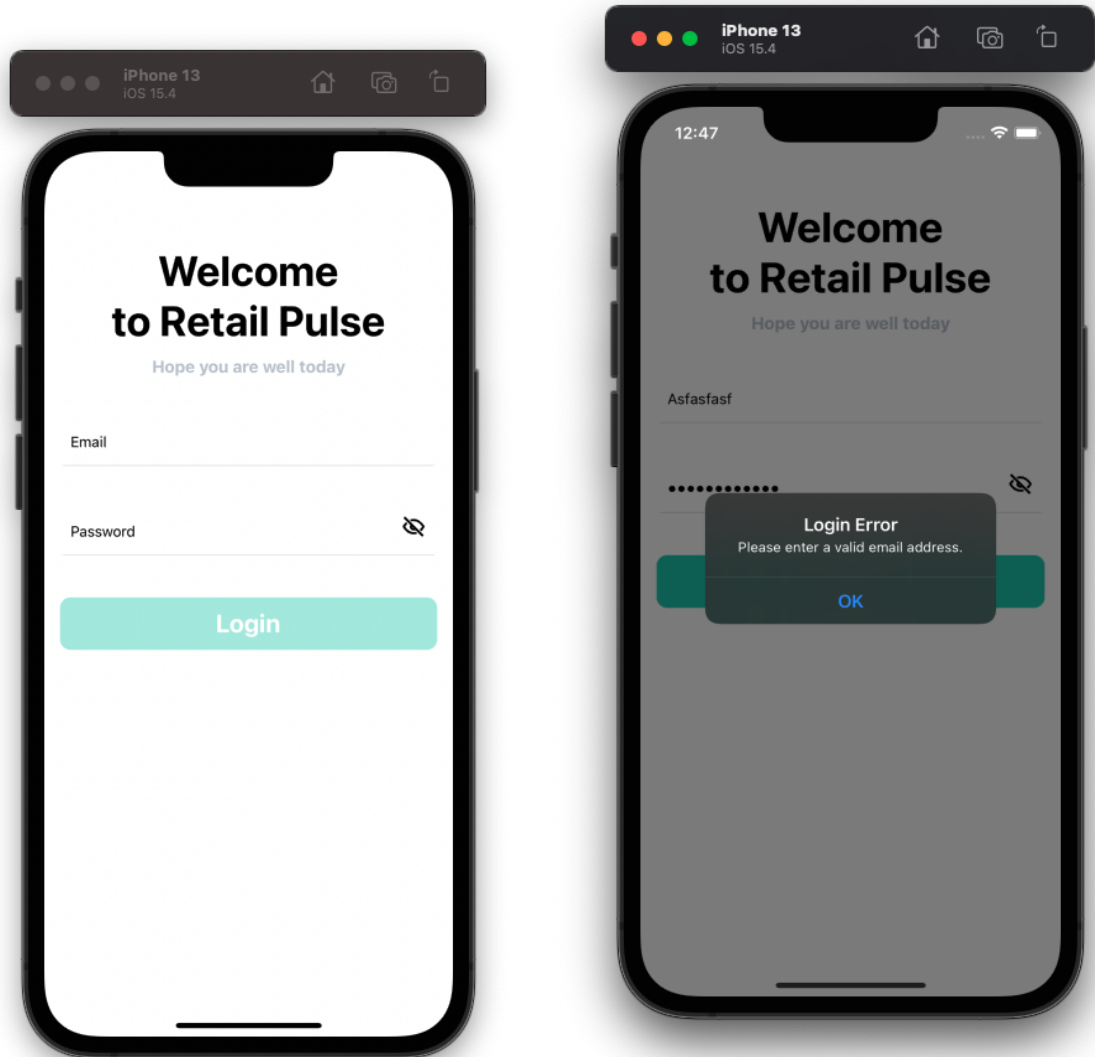
- Cross Platform (**Android / iOS**) & **landscape / portrait** mode.
- User Authentication Using Email(Firebase Auth)
- Users can access store details available to him/her.
- Users can search in stores based on name, type, route and area of the store.
- Users can apply filters based on type, route and area of store.
- Search works on filtered stores too.
- Users can upload multiple images to store in **online/offline** mode **across different sessions**.
- Provided an additional option to view uploaded images.
- Users can access app in offline mode too. (Firebase Persistence).
- Logout and re-login.

General Assumptions :

- Developed using iOS first approach due to unavailability of android device.
- Users can only login with valid email (not username).
- Search should work on the basis of all available store properties, and not only store's name.
- Filter cannot be applied on the address field, as it doesn't signify anything.
- Images can only be uploaded using Camera.
- Images should be able to be uploaded even if the user is offline (in another session).
- All the properties of a store are valid string (cannot be null).

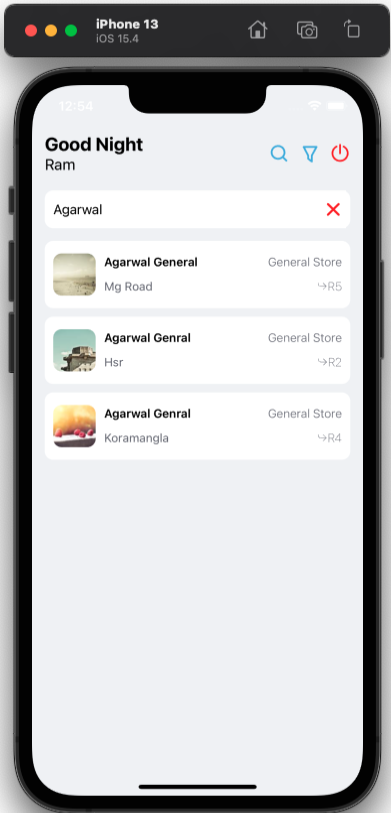
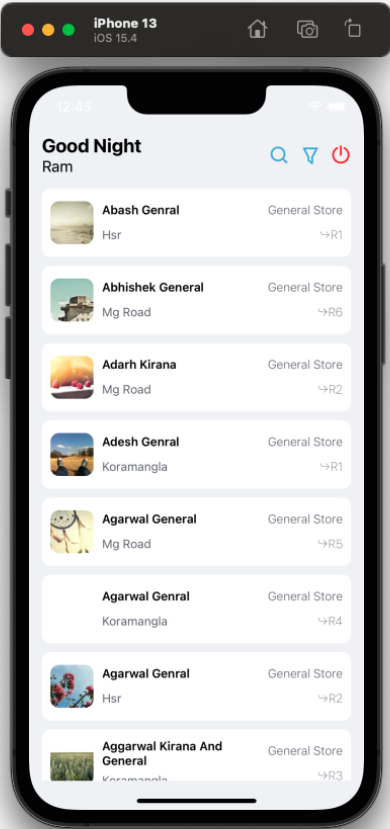
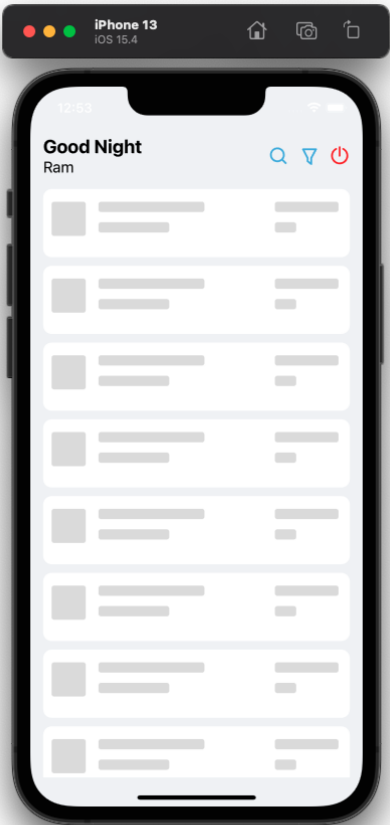
Login Screen

Requires a valid email and password. Based on the case of error, a customized alert is shown.

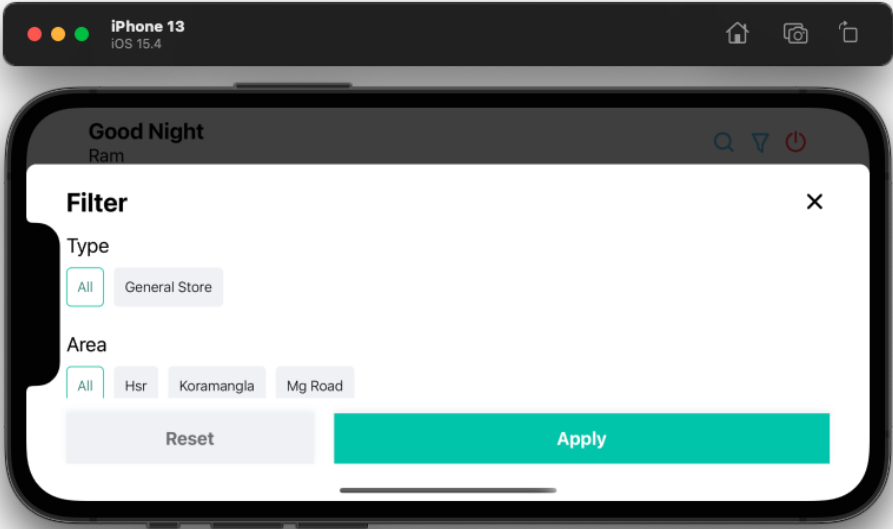
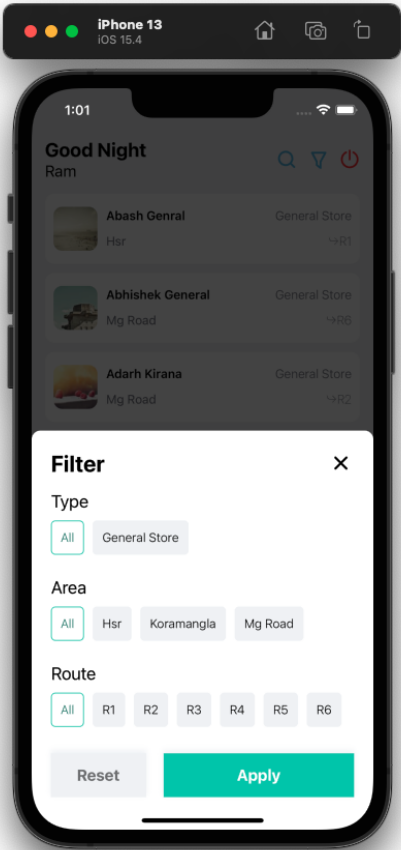


Home Screen

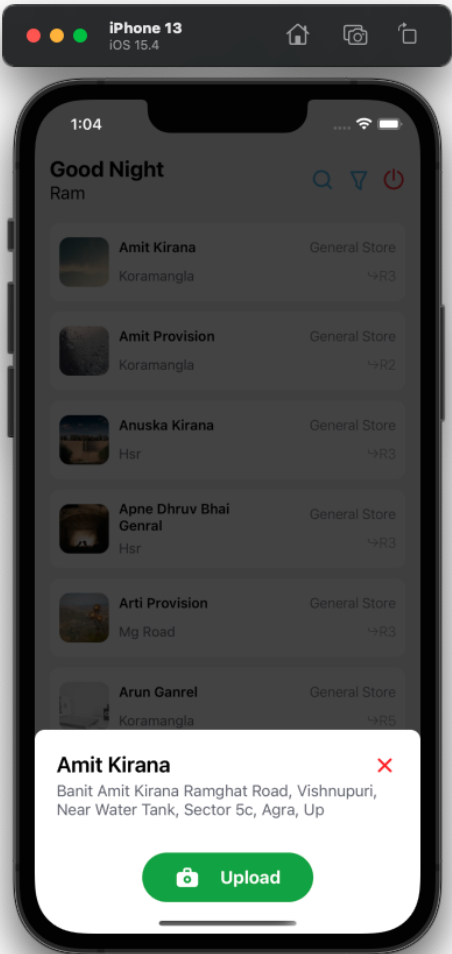
Stores list along with a personalized message and option to search, filter or logout



Filters



Store Details



Upload Image :

- 1. User presses the Upload Button.
- 2. Click image through camera.
- 3. Background upload starts.(User can continue using app)
- 4. If the user remains on the same screen can see successfully uploaded msg.
- 5. Users can anytime click on the Picture icon on the upper right corner to see uploaded images.

