**LNMIIT Outpass App**

**Documentation**

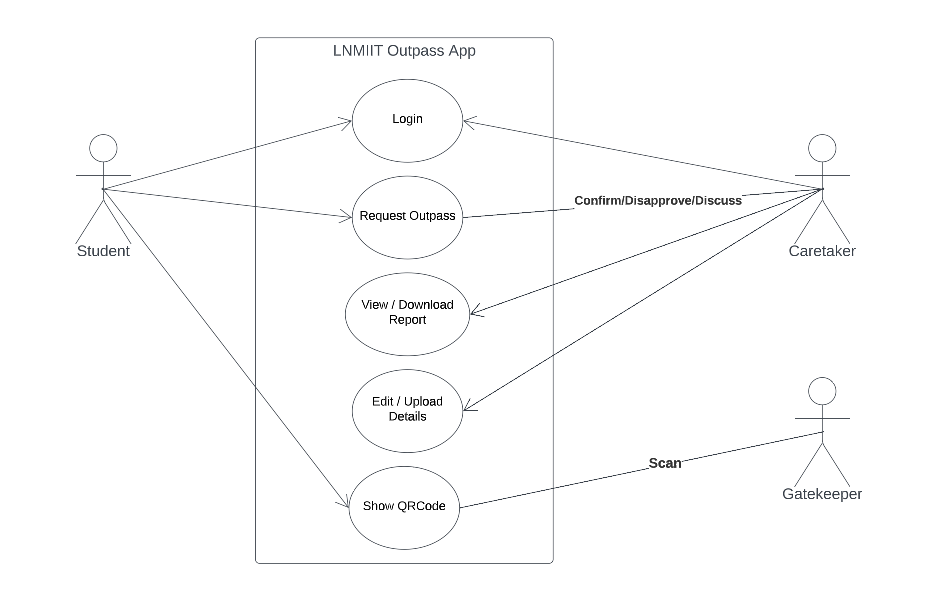
**LNMIIT**

**23rd February 2022**

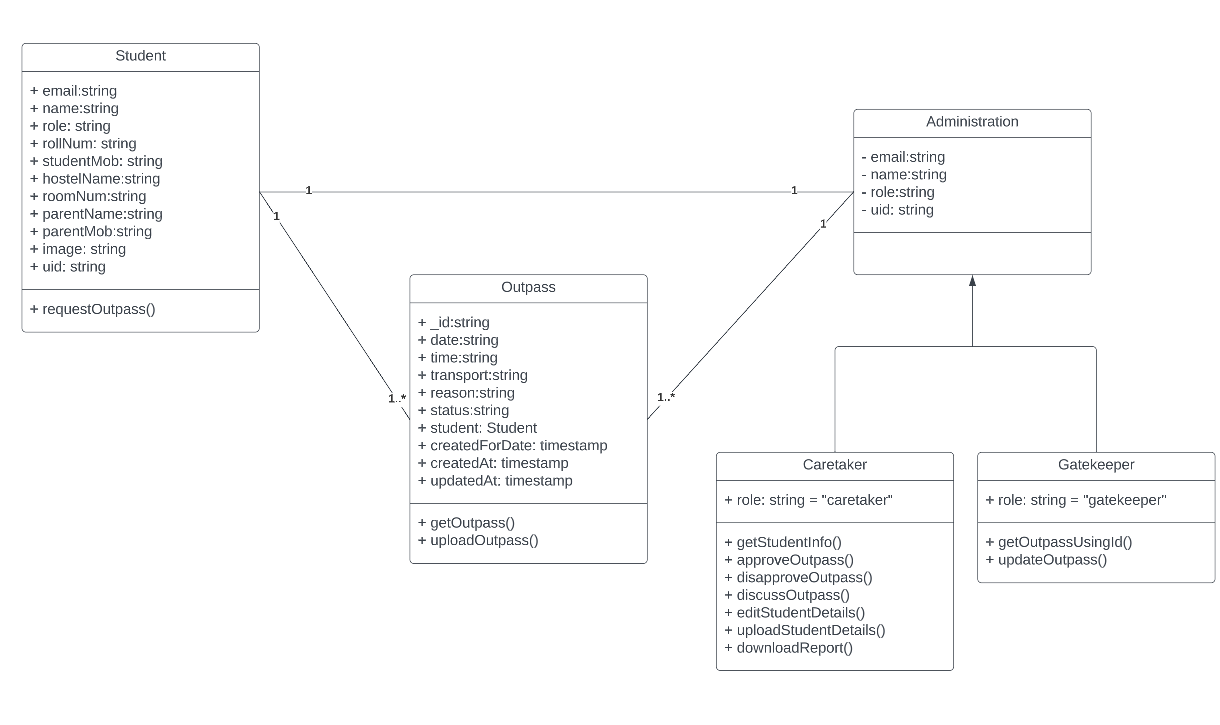
Table of Contents

1. Use Case Diagram ………………………..…………………………………………………….. 3
2. Class Diagram ……………………………………………………………………………………....3
3. ER Diagram ………………………………………………………………………………..............4
4. Firebase
5. Pricing of Firebase ………………………………………………………….............5
6. App Usage ………………………………………………………………………..............7
7. Setting Firebase ……………………………………………………………………………8
8. Collections in Firebase ………………………………………………….............10
9. Packages ……………………………………………………………………………………............9
10. Permissions …………………………………………………………………………............….11
11. Folder Structure ..…………………………………………………………………............….12

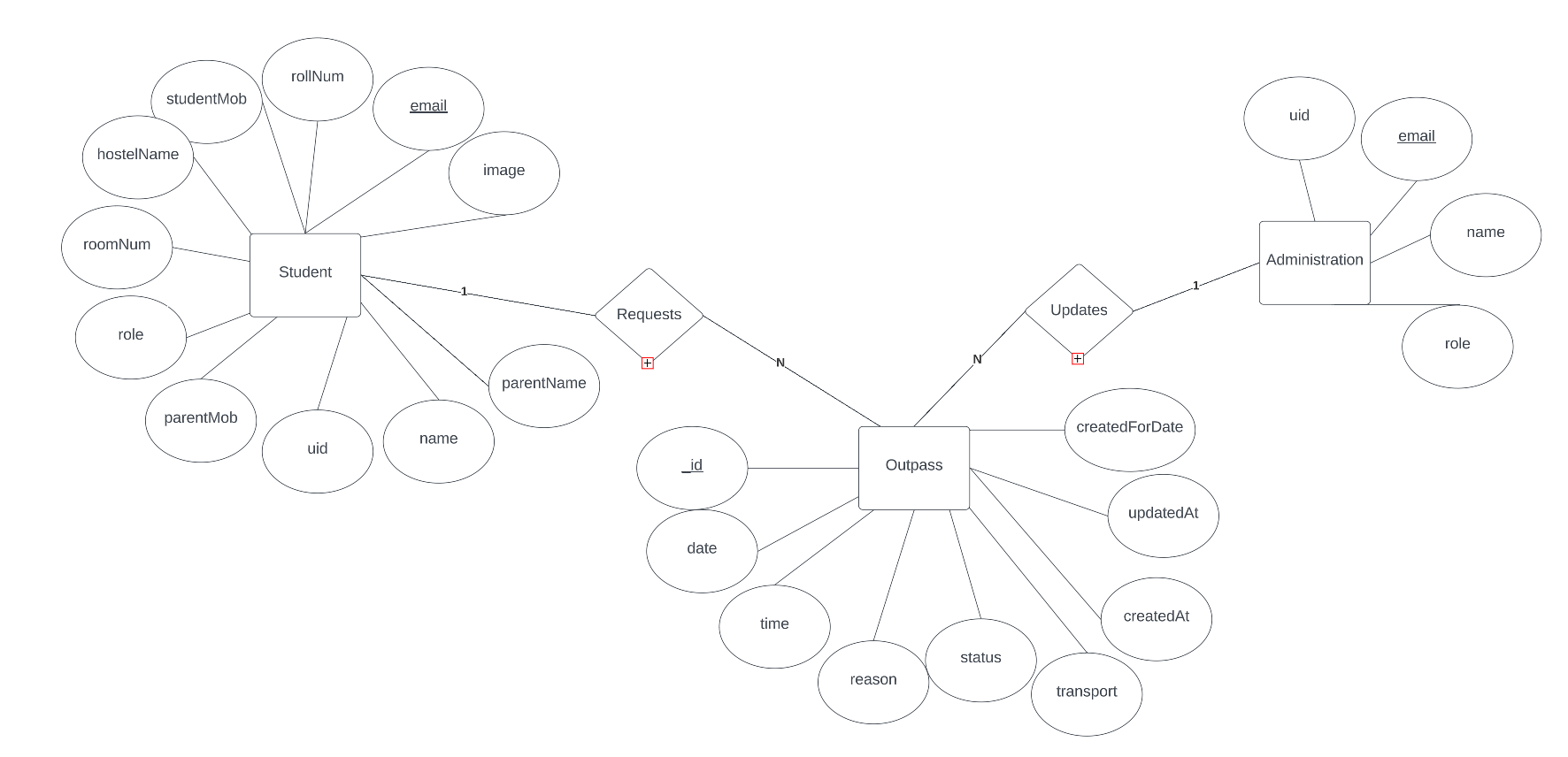
Use Case Diagram



Class Diagram

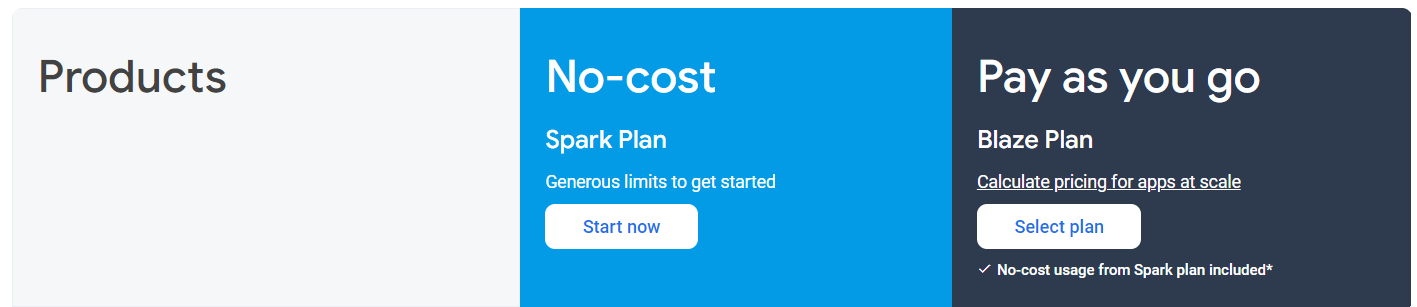


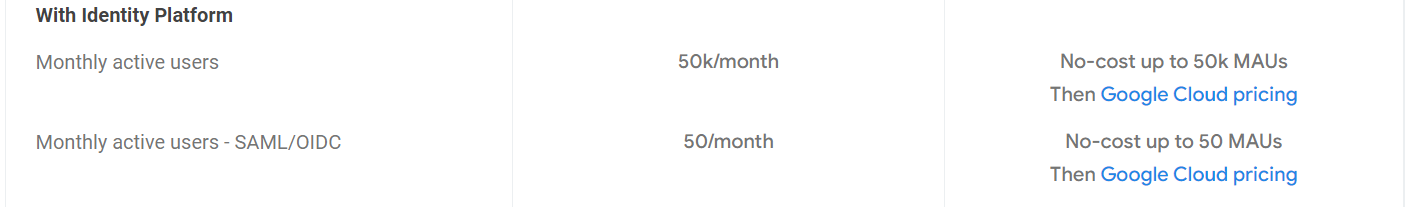
ER Diagram

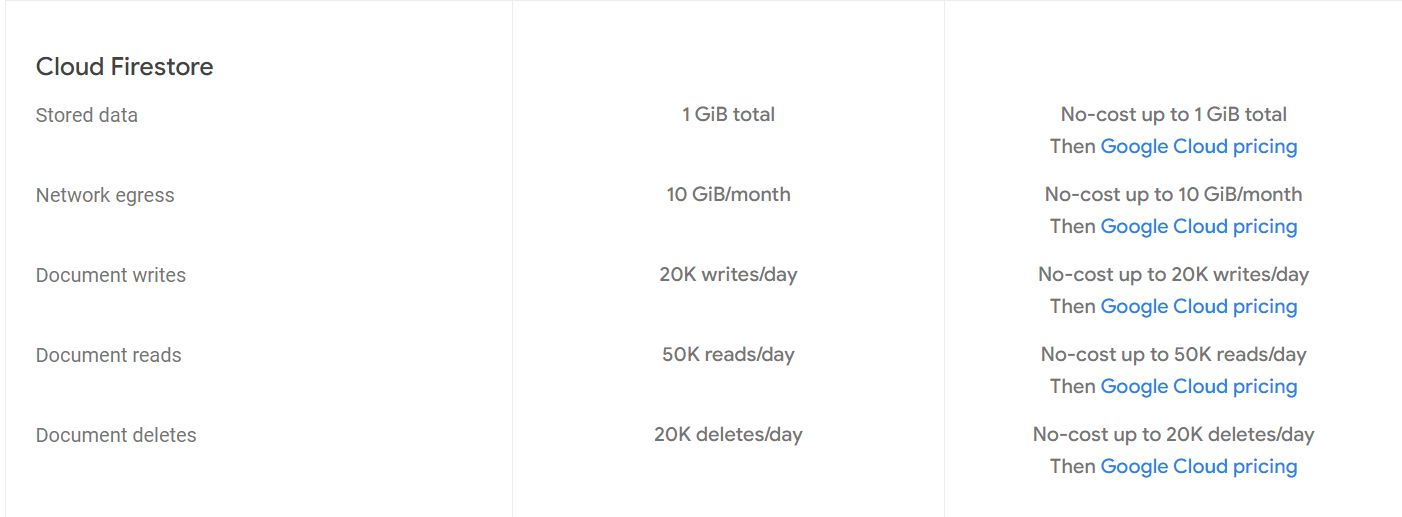


Firebase

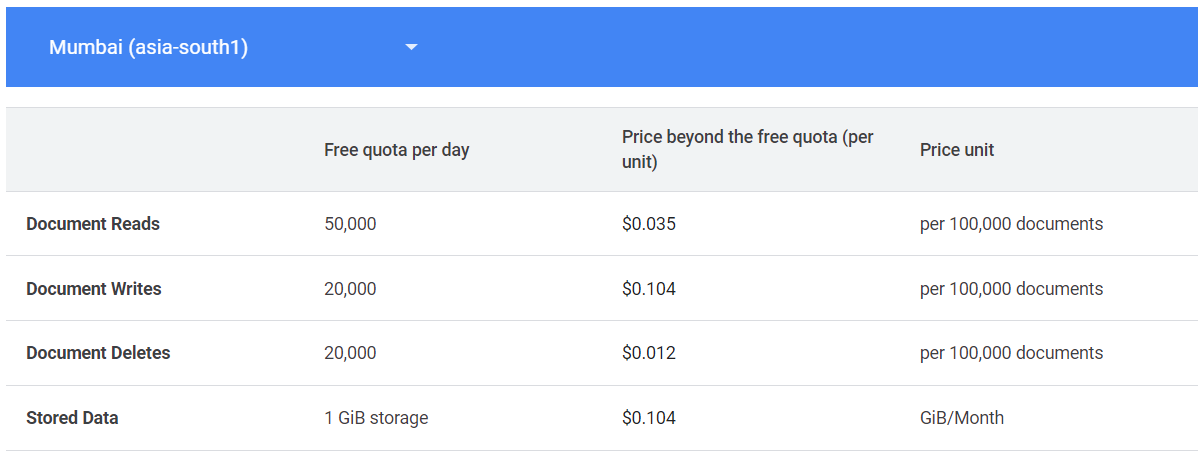
1. Pricing of Firebase:

Source:- <https://firebase.google.com/pricing> 





Source:- <https://cloud.google.com/firestore/pricing>



1. App Usage:

(i) Monthly active users -

Around 2000

Status -> No problem (Max limit - 50,000)

(ii) Write Operations -

Worst Case -> 8000/day

Status -> No problem (Max limit - 20,000/day)

(iii) Delete Operations -

No delete operation

Status -> No problem

(iv) Storage -

1GB -> Can store outpass of more than 1 year

(v) Read Operations -

On average -> 20,000 to 30,000 / day

Status -> May exceed on some busy day (Max limit - 50,000/day)

1. Setting Firebase:
   1. Creating Firebase Project -

Documentation - [Setting Up a Firebase Project for React Native](https://www.pluralsight.com/guides/setting-up-a-firebase-project-for-react-native)

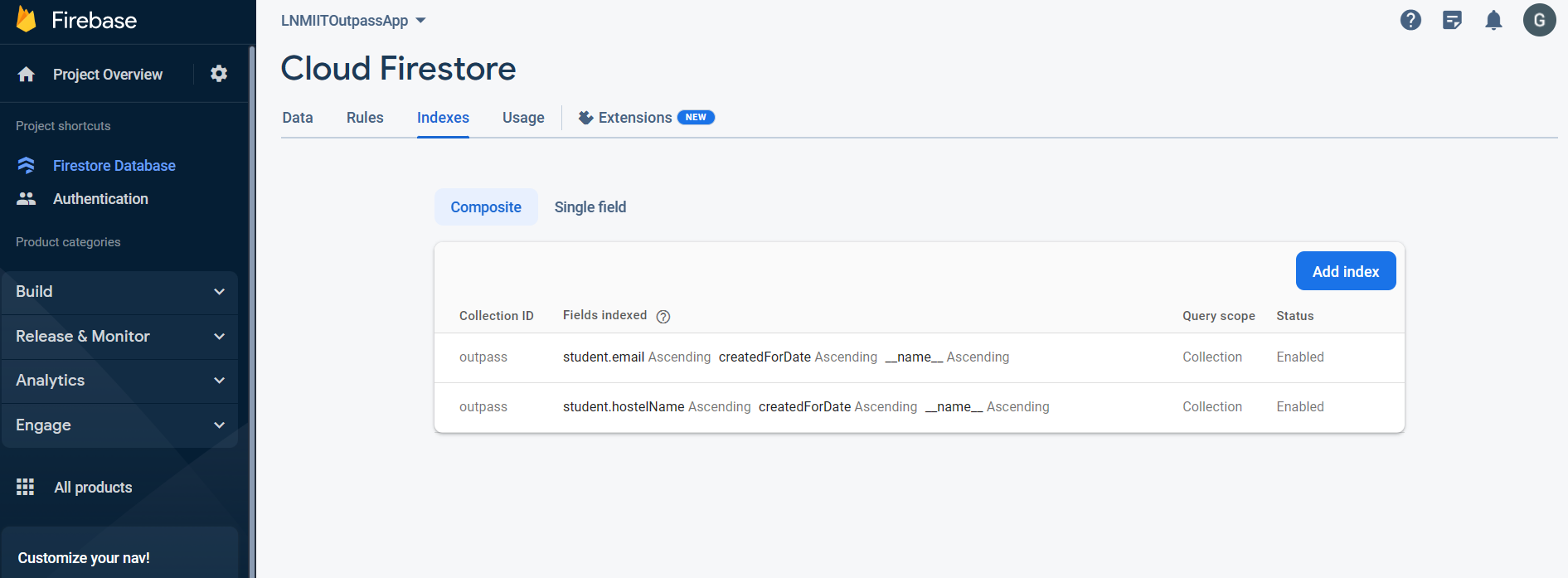
## Check document from -> Creating a Firebase Project

* 1. Setting Firebase Google Signin -

Documentation - [How to Set Up Google Login in React Native & Firebase](https://www.freecodecamp.org/news/google-login-with-react-native-and-firebase/)

* 1. Setting Index for query –

Documentation - [Add and edit indexes - Firebase Help (google.com)](https://support.google.com/firebase/answer/9137632?hl=en)



* 1. Building the app for publishing(Android) –

Documentation - [Publishing to Google Play Store · React Native](https://reactnative.dev/docs/signed-apk-android)

Check ->

* Generating an upload key (Android)
* Setting up Gradle variables
* Adding signing config to your app's Gradle config
* Generating the release AAB
  1. Building the app for publishing (IOS) –

Documentation - [Publishing to Apple App Store · React Native](https://reactnative.dev/docs/publishing-to-app-store)

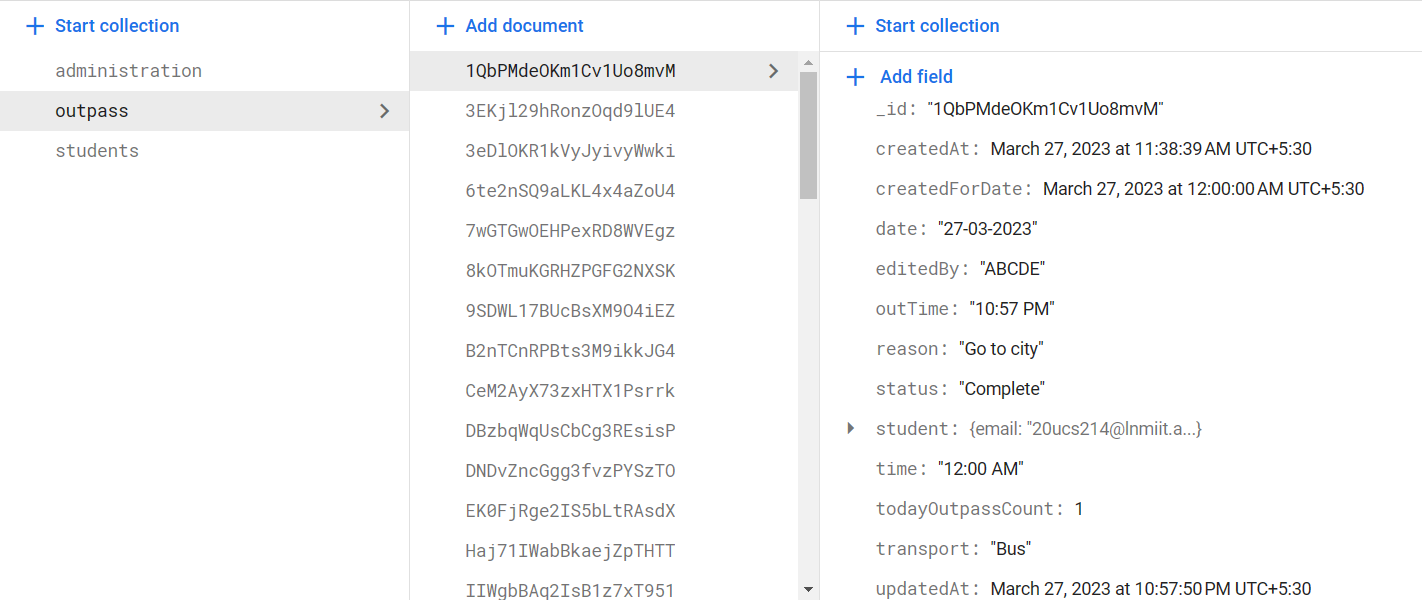
1. Collections in Firebase:

(i) administration-

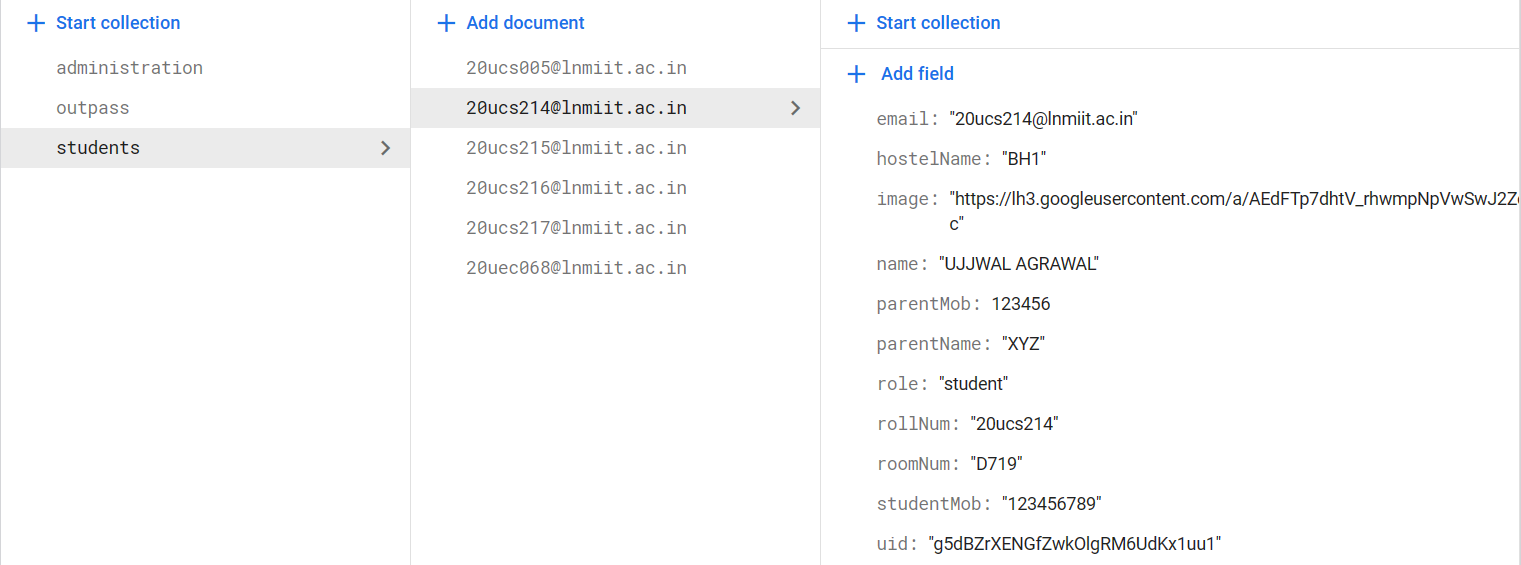
Graphical user interface, application

Description automatically generated

(ii) outpass-



(iii) students-



Packages

1) npm uninstall -g react-native-cli @react-native-community/cli

npx react-native init LNMOutpassApp

2) npm install @react-navigation/native

npm install react-native-screens react-native-safe-area-context

npm install @react-navigation/native-stack

3)Firebase

npm install --save @react-native-firebase/app

npm install @react-native-firebase/auth

npm install @react-native-google-signin/google-signin

npm install @react-native-firebase/firestore

4) Redux

npm install redux

npm install redux-thunk

npm install react-redux

5) npm install @react-navigation/bottom-tabs

npm i react-native-vector-icons

6) npm install moment

7) npm install react-native-form-component

npm i react-native-modal-datetime-picker @react-native-community/datetimepicker

8) npm install @react-native-picker/picker

9) npm install react-native-document-picker

10) npm install react-native-fs

11) npm install xlsx

12) npm i react-native-toast-message

13) npm i react-native-splash-screen --save

npm i lottie-react-native

14) For QR Code -

npm i -S react-native-svg react-native-qrcode-svg

npm i js-base64

npm install react-native-qrcode-scanner --save

npm install react-native-camera --save

npm install react-native-permissions

Permissions

1. Internet Permission:

* Primary requirement for an app to work with the internet.

1. Read External Storage Permission:

* This permission is required by only the caretaker.
* While uploading the excel sheet to the application to update student data.

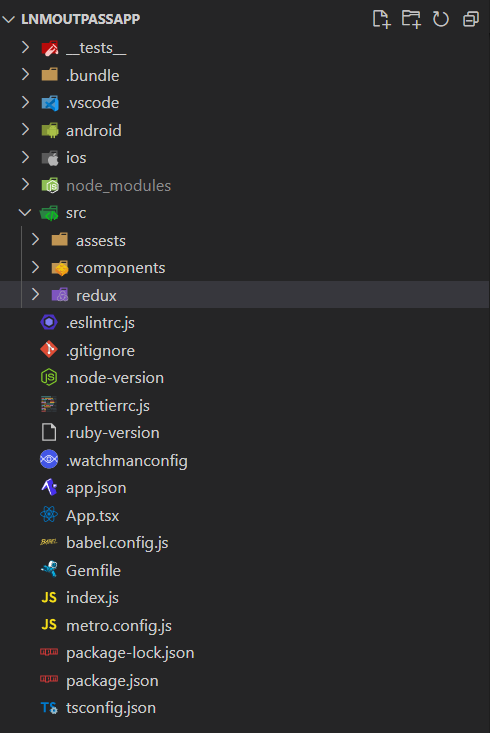
1. Write External Storage Permission:

* This permission is required by only the caretaker.
* While downloading the excel sheet to the application to the student’s outpass report.

1. Camera Permission:

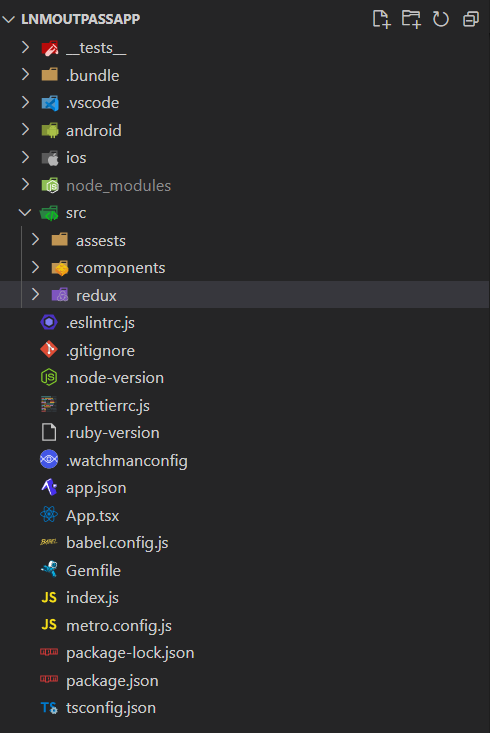
* This permission is required by only the gatekeeper.
* While scanning the QRCode gatekeeper has to allow the camera to open.

Folder Structure



Main files where all code is written.

1. App.tsx:



* After splash screen app moves to this component which first hide the splash screen and then moves the screen to main app.
* It creates stack where it put the MainScreen component.
* It also create header for the app.

1. Src folder:

Text

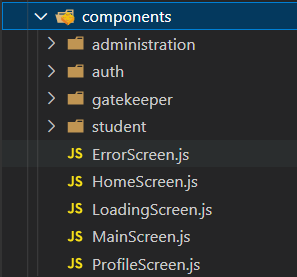
Description automatically generated

* 1. Assests folder-

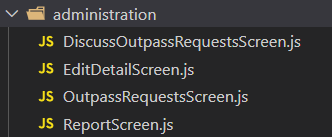
Text

Description automatically generated

* All the extra resource used is kept in this folder.
* Contains –
  + All the images in the images folder which are used in the app,
  + Contact information of the administration, and
  + Loading Animation file.
  1. Components folder-



* This is the main code of the app.
* Starts with the MainScreen which then pass the app either to HomeScreen if the login is there otherwise to the LoginScreen in Auth folder.
* MainScreen forms the BottomNavbar and is responsible for redirecting the user according to his role in the User data.
* ProfileScreen shows the details of the user.
* LoadingScreen show the loading animation.
* ErrorScreen shows the error if any major error occurs while accessing data.
* Administration Folder :-



* + Contains all the code files corresponding to administration users.
  + OutpassRequestsScreen – All the outpass requests the visible with pending status.
  + DiscussOutpassRequestsScreen - All the outpass requests the visible with discuss status.
* Auth folder :-

Text

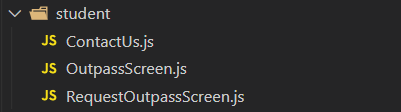
Description automatically generated with medium confidence

* + Only contains LoginScreen which helps user to login through two options –
    - Email and Password (For Administration),
    - Google Login (For Students)
* Gatekeeper folder :-

A screenshot of a computer

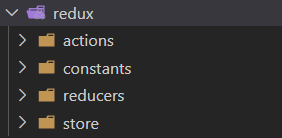
Description automatically generated with medium confidence

* + QRScannerPage – Opens QRCode Scanner to scan QRCode of outpass of the student.
* Student folder :-



* + RequestOutpassScreen – Opens form for student to request for outpass.
  + Outpass Screen – Shows all the outpass requested by student.
  + ContactUs page shows the information of the administration of their respective hostel.

1. Redux folder:



* It is mainly the backend part of the app.
* It uses react-redux for the proper data flow in the app.
* Takes data from the firebase using functions in actions and then passing it to their respective reducers and moving to the app using store.
  1. Actions folder-



* + Contains all the functions to fetch, modify and add data to or from firebase.
  + These functions are imported by the Screens to perform the required operations.
  1. Constants folder-



* + Contains all the types of actions that can be created to perform on the state of particular data.
  1. Reducers folder-

Graphical user interface, text, application

Description automatically generated

* + Contains files which stores state of the different data and different actions that can be taken on it.
  + Outpass -> Store state for outpass data (used by student as well as administration).
  + User -> Store state for user data(used by student as well as administration).
  + StudentDetails -> Store state for studentDetails data (used by caretaker to edit details of student).
  1. Store folder-



* + It configures the store which contain all the data got from the reducers.
  + This store is then imported by the Screens to use the data when required.