Song Yang (sy540)

Tong Wu (tw445)

Xin Yang (xy213)

Report for Final Project

1. Motivation

Currently, most travel sharing platforms focus on content-oriented sharing instead of location-oriented sharing, which can cause inconvenience when users want to visit the same place but the destination is hard to find.

1. Solution

We built this app to create a travel experience sharing platform where users can interact with a real Google Map. In this app, every mentioned place can be found exactly where it is and a user can easily use Google Maps to program a navigation. Missing your way in finding your destination will never happen. Also if you’re not sure about the exact place, just search ambiguous places and all interesting travel notes will display on the map, you can conveniently make a travel plan with the help of a powerful Google map.

1. Details

The key component is the search bar and the map view, users can search the destination in a search bar then all nearby travel notes arranged by address will appear on the map, users can tap the pin on the map to see what do other users think about this place and if there are some interesting but rarely known places.

Our app will request fine location access and external data read and write access permissions to provide a better service.

The first interface user will see is the sign in page, if a user doesn’t have a registered email account there’s a shortcut to the register page. All registration information is connected to and stored by Google Firebase Authentication.

After successfully logged in, user will see the main activity, which is the most core part of our app. A bottom menu with three buttons provides the switch between three fragments. The first fragment shows the nearby travel notes by default. Also, user can search the destination using the search bar on the top of the fragment powered by Google Places API, related place names will be auto completed. Searching results will replace the previous list view. This fragment is a more data saving way for the destination search. A user can tap the item in the list and the next activity will show all related information about the note, e.g. author, pictures, title, content. And users are allowed to leave their comments below, the received comments can be seen in the “My Messages” part in AccountFragment.

The second option in the menu is the map fragment. Similarly there’s a search bar on the top of the map. The map will automatically locate to the user’s location, and a floating action button on the left bottom side allows the user to write his/her own notes. The search results of the search bar will be shown on the map by pins. Only travel notes within the specified range will be shown on the map to save the phone’s resource and memory consumption. The filter operation is based on the latitude and longitude and is completed on the cloud so it won’t drag down the speed of phone. For better user experience, notes about the same address will be classified and only one marker will be shown on the map. User can tap the pin to see all related notes in the NotesViewActivity.

The floating button with a pencil icon is the way to write user’s own travel notes. After finishing the text part, user can upload up to nine pictures.(use a plugin named filepicker developed by droidninja on github, the plugin is for choose picture only, we deal with the selected pictures ourselves.) A location picker provided by Google Places API allows user to clarify the exact location this note is about, which is the core idea of sharing notes oriented by location. All related information will be stored in Google Firebase Cloud Firestore Service, which is a more advanced and suitable NoSQL database than Firebase Realtime Database.

The last option in the menu is the account tab, user can change their nickname or avatar by tapping on the avatar image. Also, user can see all notes he/she wrote and all notes user saved as the favourite. The my message button allows user to see how other users comment on the notes he/she wrote. The user can log out with the red “Log Out” button on the bottom.

The view note activity provides the view of browsing a certain note. Besides the information corresponding to the notes, user can also add a favourite on this note and check it later in account fragment.

1. UI Design

Our app has an orange theme color, and the taskbar is also adjusted into this orange to show a unified and energetic view. Customized buttons are implemented for a better look. Neat and tidy is our target so we don’t want to add many unnecessary widgets. But all widgets shown are arranged in a clean order.

1. Contribution

Generally, we separated the app into function parts or modules, we use git(Github) to control the version and sync project among developers. We tried to develop with good habit of coding, similar programming style and commit in time, in order to avoid conflicts. Learn Markdown at the same time. Personal contribution shows as below.

Song Yang: Worked on database APIs, build structure of storing image files, saving user information and popup menu. Managing images including note images, profile pictures, with firebase storage. Design interfaces using Glide, image loading function. Implemented user management, user favorite notes selection with the classes including AccountActivity, LoginActivity, RegisterActivity, MyFavoriteActivity, Note.class. Build a better UI effect with xml skills border radius. Bugfix in fragment loading problem and gradle building.

Tong Wu: Mainly worked on retrieving and filtering data from Firebase Firestore and Storage. Showed the notes in a list and a viewing interface. Transfer the data between activities and fragments to achieve viewing notes independently and on the whole. Designed and fixed the MyNotesActivity.java, ViewNotesActivity.java, ListFragment.java and MyMessageActivity.java.

Xin Yang: Wrote the MainActivity and the switch between three fragments, menu navigation and search bar powered by Google Maps API. All contents in the MapFragment including location updates and the mapview powered by Google Maps API, the floating button, adding markers on the map, further operations through markers like list the notes according to markers and view notes, and the database operations implementing Google Firebase as well the MyAdapter.class.