Android Final Project Report

APP Name: Travel Planner

Team Member:
Jingxuan chen(jc2181)
Rong Zhang(rz214)
Xiuqi Ye(xy178)
Feng Rong(fr214)

Motivation

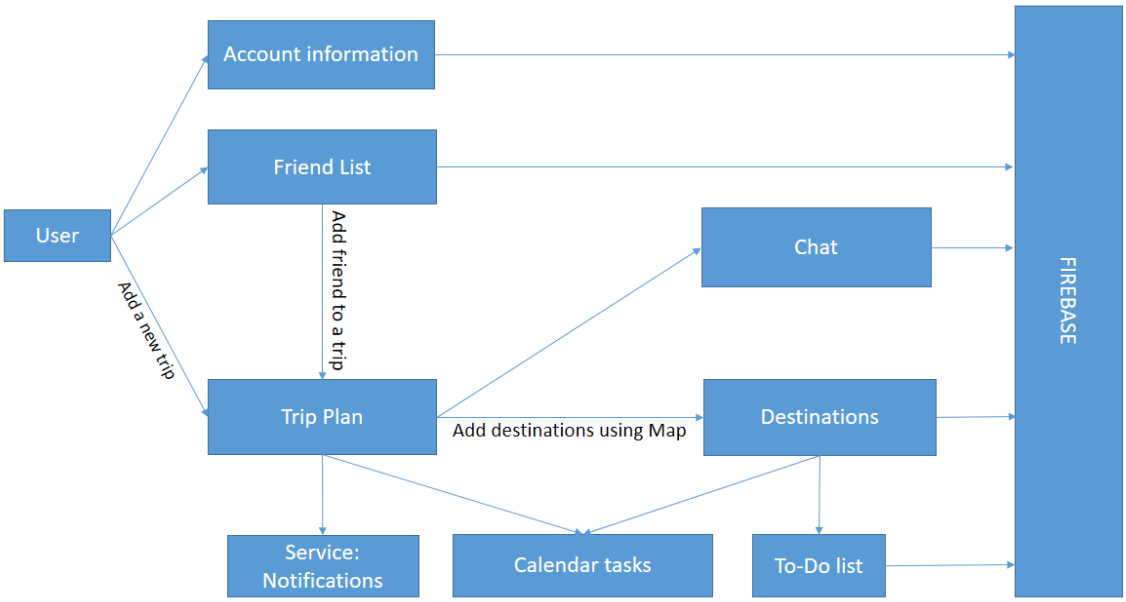
Traveling is an important part of life. Before the travel, people always want to make a plan to schedule their trip and discuss with their companions. Using this app, it is easy for a user to plan and share their trip information.

Solution

We implement an user-friendly travel planner app based on Firebase. User need to create an account to access this system. Users can create a trip plan in details easily, and the trip will be added to the system calendar. Also, users can use our app to chat with friends in the same trip. There will be push notifications for user to remind the users for their incoming trips.

User Interface

We have a clear UI which user can easily use our app. Also, there are various buttons and different gestures like click and long click can be used to achieve different goals. The manipulating system are easy for users to understand.



Key Components and Interactions

1. **Firebase:** We used Firebase, a remote database as our storage. When a user create an account, the account information will be stored in Firebase. The user can create a trip and add details to the trip, these data will be stored in Firebase too. We also implemented a chat module based on Firebase. The database of firebase mainly has four charts. The first is “user”, which has columns of “id”(key), “account”, “password” and “email”. The “travel” stores information of each trip, it has “user”, “trip_name”, “start”, “end”(store the start and end time of each trip), “clock”(mark that whether this trip has been notified), and “destinations”(contains names and time for more detailed plans, and a todo list for each trip). The chart “chat” has columns of “trip_name”, “message_user”, “message_text” and “message_time”. The last chart is “friend”, which contains user name(key) and their friends.
2. **Fragment：** We used fragment to implement map in our app. When user add a new destination in their trip, user will enter an ‘add new destination’ page where there is a map on top. Users can search a destination in the map using the editText, and the destination will show on the map. Then user can add this destination into their trip. Users can also click on the map. If the user clicked a place of interest, the name of this place will show on the editText. And if the place is not a place of interest, the address of this place will show on the editText. Then user can add the this destination into their trip.In the destination list, user can long click the name of the destination, and it will jump into a page which shows the destination in a map.
3. **Calendar:** When a user add a new trip or add a destination of a trip, a schedule will be added into the system calendar. This method is implemented in NewTrip.class and NewDest.class.
4. **Activity:**

   Account management: This component includes three activities: LogIn.class, SignUp.class and MyAccount.class. The basic function of signing-up and logging-in are provided by the Signup and LogIn activities.The application started with LogIn activity, which has two edittext for name and password input and two buttons “LOGIN “ and “SIGN UP”. When the user click “SIGN UP” button, the SignUp activity will be shown, where user can input personal information and sign up. The signed-up user data(name, password and email) are stored in the remote database. And when user click login button in LogIn page the program will compare the input username and password to the remote database. The user can also edit their basic information by click My Account button on home page such as password and email. They can also complete their personal information like phone and address in this activity.

Trip management: Users can see the existing trip on home page, and can add a new trip using trip name, start end time and member by clicking the add button on home page. Users can also delete the trip by long click the trip name. By clicking the trip name, it will switch to a page where lists the period member and destinations. Users can also chat with the trip member using the chat button. By clicking the add button on this page, the app will switch to a page to add new destination, which is described in fragment.

Friend management: Users can click the Friend button on home page to access a page where users can see all their friends. They can send a friend request by typing the user name of their friend. However, users can only send request to other user in the system. They can also accept or deny the request sent by other users.

Chatting: Users with the same trip can talk each other. There is a button on details of trip. When users click the button, the program will go to chat activity. In chat activity, there are a listview of messages, an edit text and a send button. The listview of message uses FirebaseListAdapter of firebase UI so that all messages can show up immediately when messages were sent. When users click the send the button, the message will be added into firebase. There is a class for message in ChatMessage.java containing message, user and time.

Service：The service component consists of a MyService.class and a AlarmReceiver.class. The Service component does not have user interface because it runs in the background. The main function of the Service component is a reminder of planned trips. When any trip's departure time is coming(within 2 days), the Service component will acknowledge the user by a push notification. The Service is started by HomePage Activity, and receives a name parameter(which is the name of the current user) from Homepage Activity.The Service will continue to run in the background even when the user go to other applications. On started, the Service will send a signal to Receiver(together with the name parameter so the Receiver will only focus on the current user). When the Receiver receives the signal it will return a signal to the Service and restart the Service, forming a loop (about 10 seconds). The code of push notification in the AlarmReceiver.class, Receiver will traverse all departure time of the current user, if the interval between current time and departure time is less than 48 hours(we transform both time to millisecond to do the comparison), then the application will send a push notification to the mobile phone, the user can click the pushed notification to back to their Homepage. Once a trip has been notified, there will be a "clock" mark add to the firebase of that trip so it wouldn't be notified again.

Organization and Concepts

UI design: Feng Rong, Rong Zhang
Firebase: Jingxuan Chen, Xiuqi Ye
Fragment(map): Feng Rong, Rong Zhang
Calendar: Rong Zhang
Service: Xiuqi Ye
Account and Trip Management: Feng Rong, Xiuqi Ye
Friend Management: Jingxuan Chen
Chatting: Jingxuan Chen, Feng Rong
The work organization is roughly divided, the overall project is designed and implemented by all the team members together. You can check specific java file to find the author of that java class.
Concept: We applied service, fragment, remote database and some basic concepts such as UI design listView that we learned from the class. Details are described in the previous section.

User's guide:

1.At first, you need to create an account for our app. You need to provide some information to complete this process.
2.Log in your account, and our app will show the homepage. There are three buttons, you can click them for further work:

(1) By clicking 'friend', you will access a friend management page. You can search an user by name and send a request to him/her. Also you can see the requests from others, you can add them to your friend list.
(2) By clicking 'account', you will access your information page. Here you can edit your private information except your name.
(3) By clicking 'add new trip', you will create a new trip plan. You need to type in a trip name, choosing friends in your trip, and choose a start and end date.

There are also your current trip plan list. You can click the item for further work, or long click the item to delete that trip plan.
3. When you click a trip plan, you will access the information page of this trip. There are three buttons, then

(1) You can click 'chat' to start a group conversation with friends in the list.
(2) You can click 'add' to add a new destination for your trip. When you click 'add', you will switch to a page for you to add a destination. You can use the map to find a destination and click the map to get the name or address of it. Also, you can type in your destination and search your destination on the map for further explore. Click' add' to add current destination to your trip.
(3) You can check the checkboxes in the destination list and click 'delete' to delete them. There are your current destination list. You can click the item and access a to-do list. You

can add some events and delete them when you want. Also you can long click the destination, and search your destination in a internet explorer or in a map

(4) When you add a trip, you will also add this trip to your calendar. Also, there will be a notification before your trip as a reminder.

.