CSE 535 Mobile Computing Project Portfolio Report

Android Utility Application "Carendar"

Jingyang Guo
Ira A. Fulton Schools of Engineering
Arizona State University
Tempe, AZ 85281, United States
jguo81@asu.edu

***Abstract*—As no existing calendar application satisfies the need of school life well, this Android application was created, with various smart functions to keep track of events, classes and groups.**

Keywords—Android; App; calendar; school life

I. INTRODUCTION

Besides academic challenges, students also encounter various kinds of problems during the life in college. For example, if a student takes a course in which he knows nobody, it can be very difficult to find an ideal team. If a student needs to take courses in an unfamiliar building, he may get lost in the campus and miss the class time. And for students with a very tight schedule, they may also miss classes if they don't remember the time and location of all courses clearly.

There are mobile applications that can ease such problems, like calendar applications, teamwork applications and map applications. However, they can't provide perfect solutions because they are not designed specially for students. Moreover, each application only aims at one problem, so students need to install and learn to use various applications to ease their college life.

Therefore, when asked to develop an Android application as the project of the mobile computing course, my team decided to create an application that can solve common problems we encountered in our college life, and we named it "Carendar". It's an all-in-one utility application based on a calendar, designed for ASU students. Using this Android application, students can create their events and course schedules to get location-based class reminder, get location and navigation of all buildings in ASU's Tempe campus, or find suitable team members from their classmates.

II. SOLUTION

The goal of this application is to solve some common problems in our college life. We focused on the following three problems: How to keep track of the schedule of all courses, how to find unfamiliar buildings in the campus, and how to find teammates even if you don't know anybody in the class.

To deal with these three problems, we divide the application into three parts, calendar related functions, map related functions and group related functions, each of them is focused on one problem. Also, as an application in which users can interact with each other, account management functions are needed as well.

In general, "Carender" application consists of four major parts: user management, calendar, map and group. They will be discussed one by one in the following paragraphs.

A. Account management

This part includes functions like account register, login and profile update, which need server-side support. We didn't setup our own server, instead we use a cloud service named Firebase[1] as the backend of the application. It provides interfaces of database management and account authentication that can be neatly integrated into Android projects.

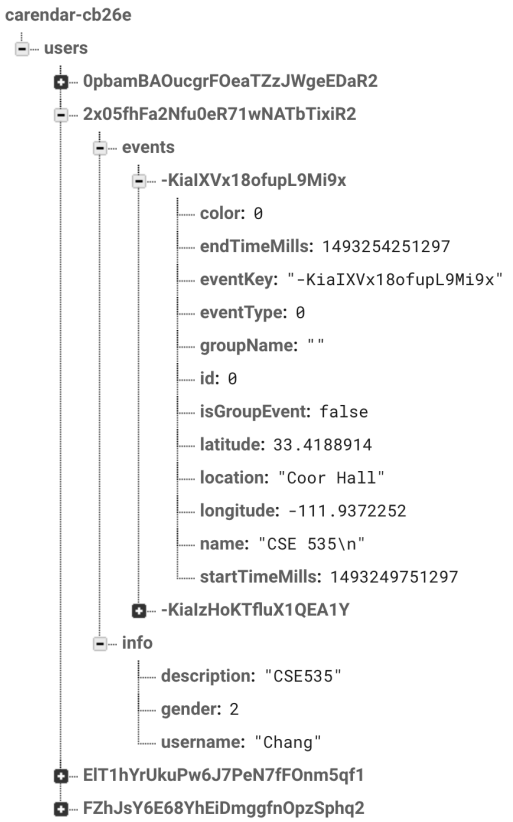

Fig. 1. Structure of the non-relational database

For account register, a new user is asked to provide an e-mail address as the account ID. We will send a link to this address, through which the user can validate the e-mail address. Besides, the password is stored after encryption, so that we won't know the original passwords of users. For profile management, the account information is formatted into a json file when being transfer between the application and the server, and it's store on Firebase in a non-relational database. In other parts of the application, we also store other data about users and their events onto Firebase in this way. Fig. 1 shows an example of the structure of the database.

B. *Calendar*

To help users keep track of their schedule, our application need a build-in calendar first. Users can add, manage and view their courses and events in the calendar. An event contains information like its date, time, name, location and type. And different types of events can have different properties. For instance, once a course in Wednesday is added into the calendar, it will occupy the same time every Wednesday unless manually deleted. But other personal event will only stay at their date and time set by the user. Another kind of event is group event. After created by a user, a group event can be shared to other users in the same team, but can only be edited by its creator. To make the calendar easier to use, we also added an import/outport function to it. If a user has his schedule in another calendar application or decide to transfer to another application, he can import or outport his schedule using CSV files.

Besides calendar itself, the application also helps users keep track of their courses in a proactive way. By utilizing the course information it collects from the calendar, the application can remind users to go to class by sending notifications. Specifically, the application will collect the current GPS data 15 minutes before the start of a class, and compute the distance to the classroom. If the phone is still far from the classroom, the application will push a notification to remind the user to go to class. In addition, the application will also collect GPS data when a class begins, and if the application determines that the phone is near the classroom, it can turn the phone into silence mode to avoid disturbing the class. Similarly, the application will check the location after a class ends, and turn the silence mode off if the user has already left the classroom.

C. *Map*

Commonly used online map services, like Google Maps, don't have information about all buildings in a university campus. So it can be very difficult to find an unfamiliar building sometimes, which is a serious problem especially when students need to go to class or take exams in that building. That why we decide to include a campus map in "Carendar".

The map view in the application is built based on Google Maps API [2]. As shown in Fig. 2, users can choose normal map, satellite map or hybrid map as the base map, which are retrieved from Google Maps. In addition to that, we integrate the information of all public buildings of ASU's Tempe campus. Users can search places as they do with Google Maps, but in the application they can also search the abbreviation of a campus building and find out where and what it is. Moreover, if the user still doesn't know where to go, he can use navigation function we provide in the application to get a turn-by-turn instruction on how to get to the specified building.

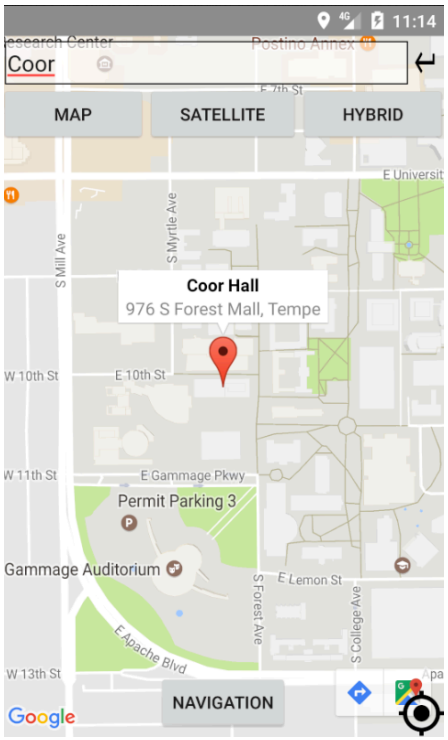

Fig. 2. The build-in map of "Carendar"

D. *Group*

Another problem in college life is that it's hard to find teammates if a student don't know any classmates. The solution provided by "Carendar" is called "shake". Users of our application can shake the phone when they are looking for teammates, and the application will use GPS data to find other users nearby, probably in the same classroom, who are also shaking the phone. When there is a match, the two users can check each other's profile, to make sure that they are in the same course and to see if their skills are what they need.

After teammates are found, a user can also create a group in the application and invite his teammates in the group to utilize other functions in the application. For example, a user can create a group discussion event and share it among all group members.

III. RESULTS

The result of our project is a ready-to-use Android utility application designed for ASU students. The following paragraphs will introduce its user interface, which is the final output of our project and what the users of the application can see.

A. *Account management*

After registering and logging in, a user is recommended to complete his account profile, which leads the user to the profile page. Fig. 3 displays a simple graphic interface for users to view and update their profile. The e-mail address is in grey, meaning that it's the account ID and cannot be changed. However, a user can customize his username, which will be

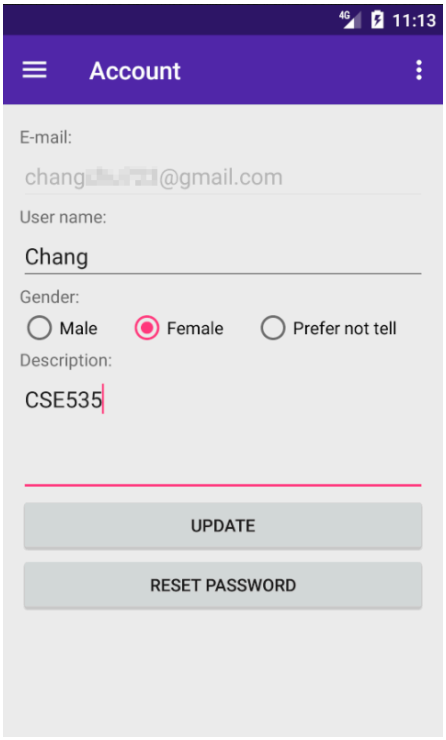

Fig. 3. Account profile page

shown to other users. Gender section is also provided, and users can choose wether to tell or not. Furthermore, users can write more description about themselves, such as courses and skills, to help them find a good match when "shaking" for teammates. Besides editing profile, a user can also reset the account password here. The user will be redirected to a webpage powered by Firebase to complete the process.

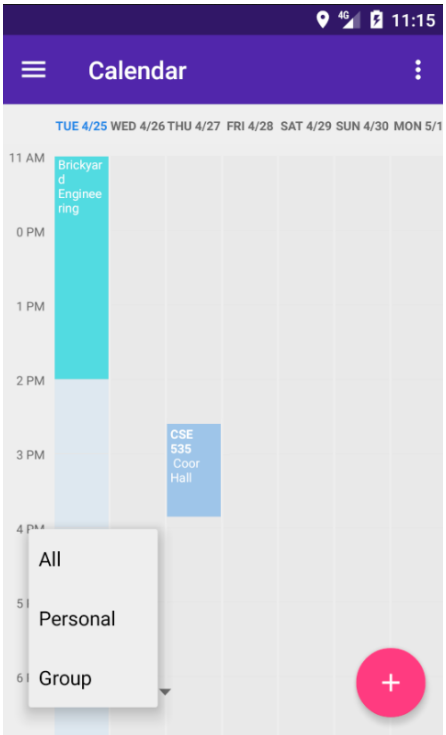

Fig. 4. Calendar page

B. *Calendar*

The calendar page is basically a daily view of of user's schedule. As shown in Fig. 4, each column represents a day, and vertical direction represents 24 hours of a day. The place and space took by a event depends on its time and date, and the color of a event is decided by its type. When wish to add a event, nobody could miss the big red button of plus on the bottom right.

Users can slide and zoom to navigate through the calendar, and by clicking an event, users can enter the event detail page to view, edit or delete the event. The small popup window on the bottom left allows users to filter the events and hide events they don't like to see. Moreover, After pressing the option button on the top right corner, users can find import and export functions.

IV. CONTRIBUTIONS

In this project, I am mainly responsible for the account management part, which include the database connection, account authentication and profile update on the backend, and the user interface of register, logging in and account profile.

Besides account related functions, I also implemented the user interface and logic of event detail page and event filter. The latter function can be seen in Fig. 4. Also, I helped solve some issues in the development of other parts of the application.

V. ACQUISITION

Many tasks in this project is new to me, so I have learnt a lot in the way of finishing them.

For example, developing Android application is actually a new experience for me. I had experience developing iOS applications, but that's in another programming language and the system logic is very different in some aspect. So, this project gives me a more comprehensive understanding of mobile development.

Integrating a cloud service is also a new skill I acquired through this project. In the past, I usually would setup a server from scratch if my project needs backend support, which can be very time-consuming sometimes. But now I learn that you can easily integrate a cloud service as the backend and spend more time on other parts of a project. In addition, Firebase also shows me a new kind of database besides relational database.

ACKNOWLEDGMENT

This course project was finished together with my teammates in the CSE 535 course, Lei Guo, Chang Chu and Zhenchao Cai. I'd like to appreciate their work and cooperation on this project.

REFERENCES

[1] "Add Firebase to Your Android Project", *Firebase*. [Online]. Available: https://firebase.google.com/docs/android/setup. [Accessed: 09- Apr- 2018].

[2] "Google Maps Android API", *Google Developers*. [Online]. Available: https://developers.google.com/maps/documentation/android-api/. [Accessed: 09- Apr- 2018].