ECE 651 Project Proposal

1. Metadata:

1.1 Project Title:

UW Carpool Application

1.2 Team members information:

Name	Email	Student ID
Fei Ye	f26ye@uwaterloo.ca	20676353
Jiatong He	j246he@uwaterloo.ca	20738590
Shuting Lian	s6lian@uwaterloo.ca	20718506
Zhilun Chang	z9chang@uwaterloo.ca	20727042
Zizheng Jiang	z225jian@uwaterloo.ca	20716960

1.3 Project introduction:

Nowadays, students in university often have great demands of usage of private transportation. Some people may travel to another city for some particular purposes such as going to an interview, attending a music festival, and others may have a demand for catching flights. At the same time, some students have a car. When they go to other cities, they will want to carry some guests to earn some pocket money or save some gasoline fee. Limited by their financial conditions, students without cars prefer to carpool with others and students with cars tend to offer their cars with others. However, for a long time, there are only 2 main ways for finding carpooling, which are either from bulletin board system or from Facebook Groups. The shortcomings of these platform are obvious. For instance, people who provide carpool can edit posts at will, resulting in a variety of text formats, so it takes a long time for people looking for carpool to find the information they want. Also, by using these tools, users who want to find a carpooling have a high possibility of ignoring previous posts and users who want to provide carpooling may still be disturbed by phone or text message even if they have enough guests. That is why the idea of our application came out.

1.3.1 What is our project

Our project is aiming to develop a mobile application that provides a carpooling platform and organize carpooling information for both drivers and passengers on Android. Passengers can search particular carpooling information that they need rather than scan through all posts, which will save their time and improve their satisfaction. Drivers can provide the basic information such as time, price, starting point and destination according to a fixed format, and can provide some other information in the remarks.

1.3.2 Reasons for the chosen environment

As a tool for daily communication, mobile phones are highly convenient. At the same time, Android OS are widely used all over the world. So, with the convenience of mobile phones, users can quickly provide carpooling and find carpooling at anytime, anywhere, compared to desktop applications.

2. Functional Properties:

2.1 Login and register functional properties

The users can log into their user accounts. If they do not have accounts, they can register accounts in the system. The property can help developers to adopt and analyze the information of users, and for the users to cache the information and record usage history.

2.2 Driver and passenger functional properties

After log into the application, the user can choose his/her identity by click “Driver” or “Passenger” on the top of the mobile’s screen.

2.2.1 Driver function property

If the user is considered as a driver, he/she should provide the origin, the destination, date, time, car vacancy, and price. After the user clicks the confirm button, the interface will display whether the user has successfully posted or failed to post the information.

2.2.2 Passenger function property

If the user is considered as a passenger, he/she just need to provide the origin and the destination. Then, the interface will return a list of drivers’ information for the user to choose. By clicking on a driver, the interface will display information about the driver and the remarks provided by the driver.

2.3 Database functional properties

Based on the characteristics of our app, we have lots of situations involving data interaction. We believe our database (the relationship between different identities) must be simple and clear enough to make sure our app running stably and offer better service to users. When the driver provides basic information, we will parse the information provided and put it into the database to wait for the information provided by the passenger to interact with it. When the passenger provides the information, we extract the keywords based on the information the passenger provides and then generate relevant search statements, extract relevant data from the database and return it for the passenger to pick. After the passenger has selected the driver, we will confirm whether there still have seats on the driver’s car. If there is no seat, the driver’s data will be deleted from the database.

3. User Scenarios

3.1 Scenario 1 (Passenger perspective)

I am a student at the University of Waterloo. I just received an email telling me that I need to go to Toronto for an interview two days later. I searched the relevant carpool information on the internet and found that it was too time-consuming because it was difficult to find a car from Waterloo to Toronto. Also, I need to compare the price and other conditions, and all the posts are not even on the same page. So, I pull out my phone and use the UW Carpool application, just need to input the origin and the destination, then all the driver’s information I want shows up on my screen! Finally, I choose a car with a reasonable price and time to take me to Toronto.

3.2 Scenario 2 (Driver perspective)

I am a University of Waterloo student and I want to go to Toronto to participate in a dance competition. I have a car, but you know, if I drive alone to Toronto and back to Waterloo, the price of gasoline will cost me a lot of money and it will make me feel sad, so I take out my phone and log into the UW Carpool application, and then I post the information about going to Toronto and going back to Waterloo with reasonable prices and available seats. Finally, I pick up some people on my way to Toronto and back to Waterloo, saving money for myself and providing convenience for passengers.

4. Non-functional Properties:

4.1 Response time

According to various connection conditions, 95% response time for the login transaction should be less than 2s and for searching should be less than 3s for peak workload during normal operations.

4.2 File size

The overall size of the mobile application is less than 100 MB.

4.3 Scalability

The application can accept at least 20 users online simultaneously.

4.4 Reliability

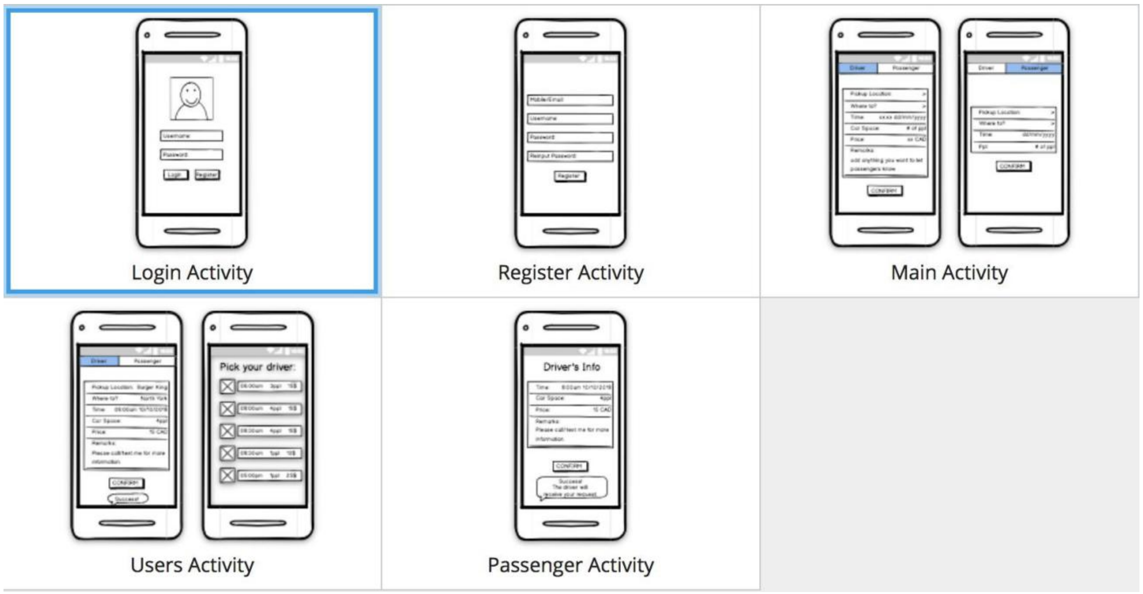
The back-end server of the application will be deployed on a professional third-party platform which has 24×7 customer support, which ensures reliability.

4.5 Compatibility

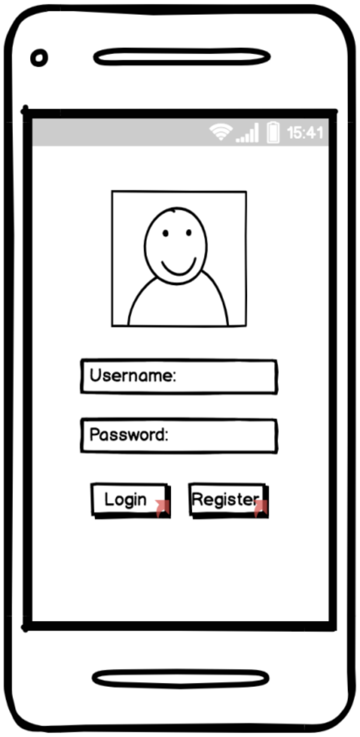
Our Android project will use API 21: Android 5.0 (Lollipop), by targeting it, the app will run on approximately 85.0 % of devices. This data was referenced from Android Platform/API Version Distribution.

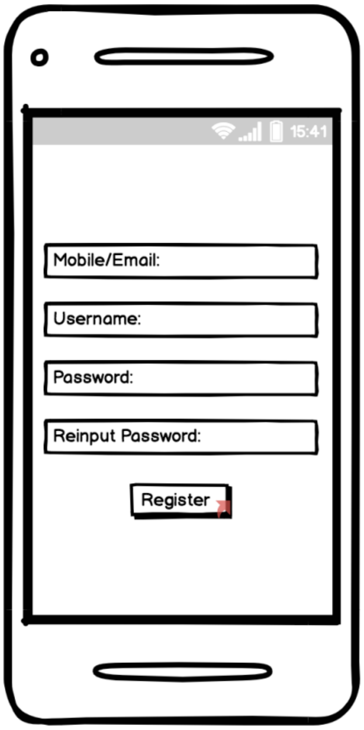
5. Low-fidelity mockups

5.1 Summary



5.2 Login & register activities



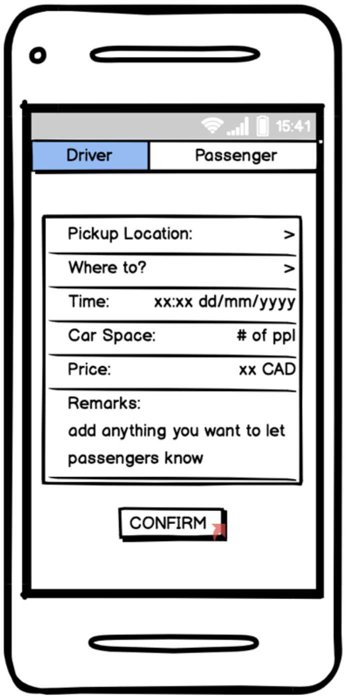


The pictures on the left are login and register activities.

Login activity: users can input their username and password to log into UW carpool application.

Register activity: users need to provide their basic information such as phone number, email, username and password.

5.3 Main activity



Pickup Location: >

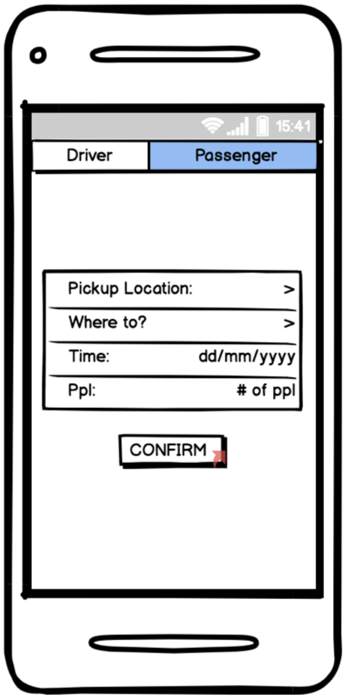
Where to? >

Time: xx:xx dd/mm/yyyy

Car Space: # of ppl

Price: xx CAD

Remarks:
add anything you want to let passengers know



Pickup Location: >

Where to? >

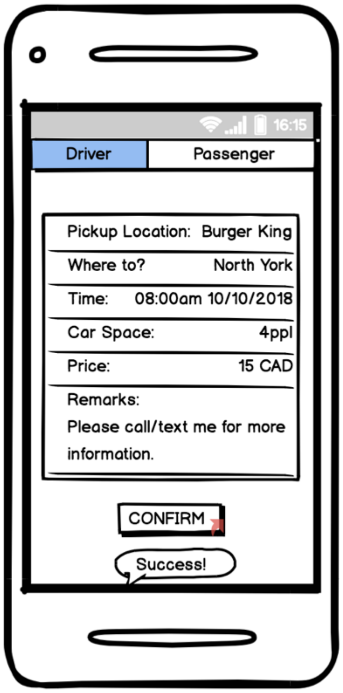
Time: dd/mm/yyyy

Ppl: # of ppl

We split this activity into two parts based on the user's identity.

If the user is a driver, he/she needs to provide origin and destination. The user also needs to provide the time, the number of the seats left on the car and the price. Driver can also add some remarks for passengers. The passenger only needs to provide basic information.

5.4 User activity



Pickup Location: Burger King

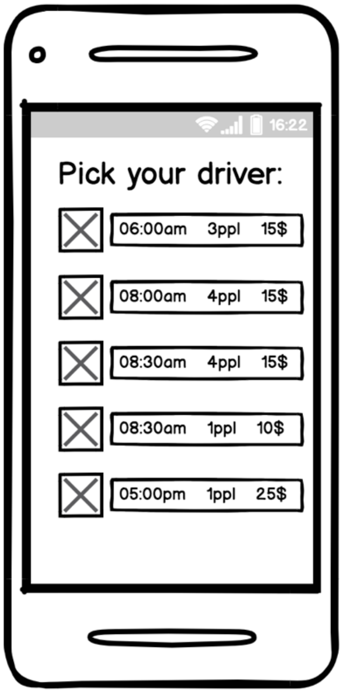
Where to? North York

Time: 08:00am 10/10/2018

Car Space: 4ppl

Price: 15 CAD

Remarks:
Please call/text me for more information.



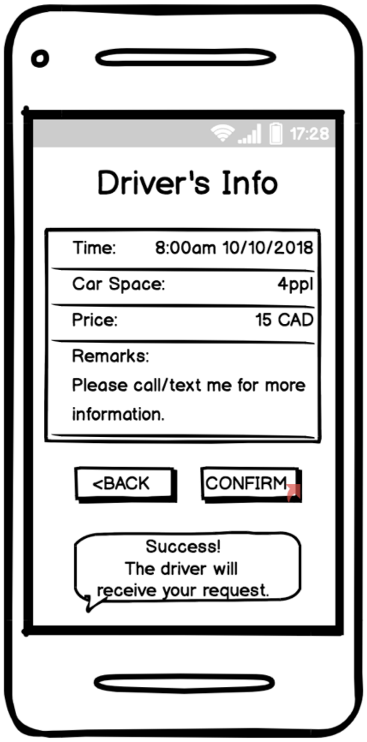
Pick your driver:

☒	06:00am 3ppl 15$
☒	08:00am 4ppl 15$
☒	08:30am 4ppl 15$
☒	08:30am 1ppl 10$
☒	05:00pm 1ppl 25$

After driver confirms, the interface will pop up a bubble shows information such as success or fail.

After passenger confirms, the interface will return a list of drivers with depart time, the number of seats and prices.

5.5 Passenger activity



Driver's Info

Time:	8:00am 10/10/2018
Car Space:	4ppl
Price:	15 CAD

Remarks:
Please call/text me for more information.

Success!
The driver will receive your request.

After the passenger selecting a driver from the list, the interface will show details of the driver’s information.

If the passenger doesn’t want to choose this driver, just go back by clicking the back button.

If the passenger wants to choose this driver, click the confirm button and then a bubble will show basic information such as success or failure.