**Taxi Sharing Application**

Andrew Dodge – 100938015

An Honours Project submitted to

The School of Computer Science in partial fulfillment of

Bachelor of Computer Science Honours with

Software Engineering Stream

Carleton University

April 22nd, 2019

Approved by:

Honours Project Supervisor: Dr. Doron Nussbaum

School of Computer Science Undergraduate Advisor: Edina Storfer

**Abstract**

**Acknowledgments**

**Table of Contents**

Abstract……….……………………………………………………………………………………i

Acknowledgements………………………………………………..………………………………ii

Table of Contents………………………..………………………………….…………………….iii

[Chapter 1: Introduction 1](#_Toc5020153)

Problem Definition……………………………………………………………………………………...1

Motivation……………………………………………………………………………………….……...1

Contributions…………………………………………………………………………………………....2

[Chapter 2: Background 4](#_Toc5020154)

Definitions……………………………………………………………………………………………...4

Development Software………………………………………………………………………….……...4

Previous Work………………………………………………………………………………………....7

[Chapter 3: Main Contribution 13](#_Toc5020155)

[Chapter 4: Implementation 14](#_Toc5020156)

[Chapter 5: Conclusion 15](#_Toc5020157)

[References 16](#_Toc5020158)

[Appendix 18](#_Toc5020159)

Balsamiq Mockups……………………………………………………………………………………...TBD

# **List of Figures**

Figure 1: Ridesharing Market Share……………………………………………………………..10

Figure 2: Communication Diagram………………………………………………………………15

Figure 3: Uber Pre-Ride Screen…………………………………………………………………..18

Figure 4: Lyft Opening Page……………………………………………………………..19

Figure 5: Balsamiq Mockup of Pre-Match Screen……………………………………….20

# **Chapter 1: Introduction**

Taxi sharing is a modern method of transportation that establishes a cross between bus and taxi travel. The central idea of which being a taxi that can pick up multiple people, either at the same location or multiple locations along the way, and then drop them off at their desired location (preferably in the same general area). To give some additional information, taxi sharing is more beneficial than traditional taxis as each patron pays a cheaper fee if the ride is shared with other passengers. Furthermore, the cab driver is paid by each individual passenger meaning that they end the day with a greater income.

*Problem Definition*

The honours project for COMP4905, which was completed this semester, was to design and develop a taxi sharing system for the Android device. The taxi sharing system consists of three components: a passenger application, a taxi driven application, and a back-end. The focus of the project was on research and design, to determine what applications already existed within this concept and to design an application that functions on multiple different device sizes for a wide range of individuals. A back-end communication system was implemented; however, there was minimal attention paid to it as it was not the focus of the project. This project was done in tandem with another student, Mohamed Gahelrasoul, with an even division of labour. His task was to implement the driver application of the system as well as to implement the back-end communications. The other half of the project, which is the primary focus of this report, is the passenger application component of the system. It was decided at the beginning of development that the driver would not have a large variety of options associated with them, which will be further discussed later in the report. This was the critical reasoning behind the division of labour.

*Motivation*

The motivation for this project stems from three different sources. The first source was to develop a taxi sharing system which would form the base for a future project. This system is to be further expanded into an autonomous vehicle taxi sharing application, where driverless vehicles will be accessible to passengers to take them to and from their destinations for a fee. Especially with the introduction of self-driving cars having become more prevalent in today’s society. To have an autonomous vehicle that could transport small groups of people throughout cities and to their destinations would be the next logical step once the technology becomes realized. This source was the reasoning behind making the driver side minimal, as it will eventually be eliminated entirely. To continue, the second source of motivation stems directly from the autonomous system spoken of previously. If it is possible to evolve this system to the point of automation then this will cause a decrease of vehicles on the roads. This reasoning is why this technology is so important to research. This concept would also aid to alleviate the number of vehicles on the roads, since every person who decides to share a ride with someone would ensure one less vehicle out on the road. This is crucial in modern society, where pollution from transportation has become significant and pervasive. Reducing the number of vehicles on the road would aid in decreasing the amount of pollution emitted in cities that possess this system. Further motivation for this project stems from an unrelated source to the previous two points. It is an intrinsic desire to expand on knowledge of application development and design and contribute to this development in a meaningful way. This will aid in creating a solid foundation of skills to be taken into future career paths. To continue, as stated above, the purpose of this project is to create an application in the present that will then be evolved in a future project. The problem being addressed by this project may not appear to be globally important to the present population. However, it does not make it any less important to research and begin implementation now. Especially due to the fact that there are a number of taxi sharing applications in circulation already. There is hope that the future evolution of this project will set it above the standards of current applications.

*Contributions*

The work completed during this project was a majority of the implementation of a taxi sharing system. The project consisted of two separate applications that share a back-end database and communication system. The passenger application was the focus of this project, and consisted of two components, research and implementation. The research focused on how taxi sharing works and how to implement a working system. The ultimate goal of the project was to determine what information was required in order to match passengers with taxis. The implementation aspect focused on the design of the application itself. Firstly, to get it to be visually appealing when using it and secondly, to ensure it can function smoothly on Android devices. The manner in which the design was implemented makes it so that the application will scale based on which Android device the application is installed on. This is especially important when it comes to the text visible on screen. The design allows for the application’s text to be able to mirror the user’s text-size preference. If a user has their text default set to “large” then the application will use this decision on load. With regards to the application itself, it is fully functional to an extent, able to request a ride in order to be matched with a driver. This information is displayed on the screens for all users to inspect. More detail will be given on the interactions and abilities of the users further into the report. The driver side application implemented by Mr. Gahelrasoul as mentioned above will be further expanded upon in his report, as well as the communication system.

The report follows a basic layout. Chapter One was an introduction, which included the motivation of the project as well as a small project description. Chapter Two will discuss the background of the system. This will include a description of the software used to complete the project as well as a discussion of the work which currently exists in this field and how they were used as inspiration for this application. Chapter Three will further discuss the main goal of the project, delving into an in-depth discussion of the problem attempting to be implemented and the objectives of the application itself. Chapter Four will discuss in detail the process of the application’s development such as how the application was completed from start to finish, the testing which was conducted during development between Mr. Gahelrasoul’s app and this application. As well as a discussion of the errors which were discovered during development and how they impacted the end result of the system. The final chapter will be a conclusion that will briefly summarize the information that was started throughout the report. Following the written portion of the report there will be a references section which will include a bibliography of all the research used during the project. Finally, there will be an appendix section which includes the design of the application as well as any other referenced images mentioned throughout the report.

# **Chapter 2: Background**

This chapter reviews the systems that attempt to provide taxi sharing application. It also provides preliminary information or systems used in this work. As was discussed above, this project was to design a taxi sharing application using Android Studio that would function across a number of Android devices. This chapter will define the needed information that is required to understand the concepts of the project. It will begin with a basic definition of the purpose of the application, followed by a description of the software that was used. Finally, this section will discuss the applications similar to the one designed in the project and how they were either used as research material or inspiration towards the completed project.

*Definitions*

Taxi sharing is defined as a service that combines the simplicity of taking a taxi incorporated with the passenger-sharing ability of a bus. The main difference between this service and a regular cab is that a passenger will be matched with strangers who have similar destinations, and these additional passengers will be picked up along the route. This is opposed to a taxi, which involves the typical privacy that standard taxis provide. Furthermore, taxi sharing monetarily incentivises riders to share the ride with strangers, due to the collective fee that contributes to reduced individual fees. With respect to this, since each passenger must pay a fee to use the service, this means the driver would yield a greater income if there are a higher number of passengers. With this in mind, this was the goal of the project, to design an application that would achieve full taxi sharing capabilities. From matching similar users together, to displaying their trip on the screen, to completing the ride and displaying the proper information to the user post trip.

*Development Software*

Before discussing the components of the application and how the system came together it is important to touch upon the program used to design the project. The application was designed using a program called Balsamiq Mockups. This program is a prototyping tool where a user can create low-fidelity prototypes, or digital sketches, of the project being designed.[[1]](#footnote-2) This gives the developer the ability to produce ideas, facilitate discussion, and expand upon understanding before any code is written. An important aspect of Balsamiq that sets it apart from other prototyping tools is the ability to link the digital sketches together as if it were a real application. With this feature, it is possible to mimic application flow. For example, if clicking a button opens the user’s profile then it is possible to mimic that interaction in the mock-up. Once the entire design has been implemented in Balsamiq, it becomes quite simple to modify elements and view how interactions function before the development begins. Development also becomes greatly simplified with the addition of examples to help base each stage of design on. All Balsamiq Mockups for this project are visible within the Appendix section at the bottom of this report, as well as a description of how the images are linked together.

To continue, the application was implemented using the program Android Studio by Google. It is the official integrated development environment, IDE for short, for application development for Android devices. Essentially, it is a program designed specifically to create applications for Android devices, with the ability to both design the user interface and create back-end functionality. Android Studio is based off the IntelliJ IDE and incorporates the shortcuts used within that development tool to aid with programming.[[2]](#footnote-3) There are two main languages a user can program in within Android Studio, users can choose between Java and Kotlin. Both with unique benefits and complications. For this project the language of choice was Java, for ease of implementation and back-end simplicity. With regards to development, Android Studio divides the workload into two distinct parts. The design of the screens (or activities as they are called in Android Studio), which is done in an extensible markup language, xml, file. It is possible to add elements, such as text, buttons, and images, by either writing the xml code straight into the source code, or by visually designing the application on the blueprint screen. When using the blueprint screen, it is possible to drag the elements needed onto the given activity. This is the visual method of creating each page of the application. Once an element has been added to the activity using this method, Android Studio will automatically add the xml source code to the file. This method of design is reminiscent of the development environment Unity, where visual elements are added to the screen in this manner and the back-end of these elements are programmed afterwards in a separate file.[[3]](#footnote-4) The separate file in this case being a Java file per activity in the application. Each respective Java file contains all the back-end functionality that the activity needs in order to accomplish the tasks assigned to it. Be it a button, referencing user information, or displaying specific information on the screen, Android Studio can also be linked with multiple different resources to aid in the production of the application. For example, Github can be linked to the Android Studio application for ease of backing up the user’s data. Furthermore, since Android Studio uses a Gradle-based system for compilation, a user can connect directly to a Github project using a compilation command. This command will direct the compiler to include the source code stored in Github and add said source base to the amalgamation of files that make up the core of the application. This Gradle-based system also allows the user to specify the required permissions and Google Play Services the application will require for installation.[[4]](#footnote-5) Finally, with regards to testing the application using Android Studio, the program has two built in functionalities where a developer can view the work they have completed thus far. Both methods will involve compiling the application and creating an Android Package, APK, that will then be installed on a device. This is where the difference lies, developers can choose to install the application on a virtual Android device, or onto a physical Android device. The virtual machines are built into Android Studio and can be downloaded at any time during development, (they are helpful tools as a developer can then test how the application will react on different Android devices), or different versions of the operating system. Furthermore, it can be used to quickly test features of the application as the developer does not have to change environments. The virtual device is stored on the same computer as the development environment. Comparatively, the developer can use a physical Android device to run the application. This is easily done by plugging an Android device into the computer at the time of compilation and selecting it as the desired location. While this method is slower than using a virtual device, it allows the developer to test how the application will function in the real world, especially if the application has network components. With the application on a physical phone the network components can be tested with the user’s data or wireless internet. Android Studio visual examples are listed in the Appendix section of the report.

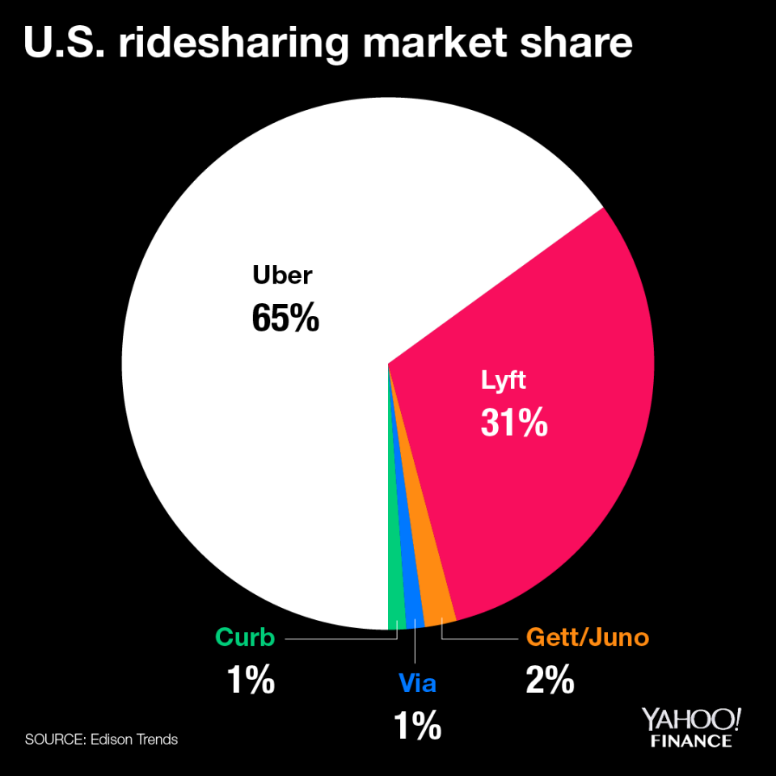
Speaking of network capabilities, this application has an online database and communication system that uses Firebase on the Google Cloud Platform.[[5]](#footnote-6) Firebase is a Backend-as-a-Service, a model for providing web and mobile app developers to link applications to back-end cloud storage. This project’s application uses Firebase for a wide array of features. The primary role that Firebase fulfills is as a database. All user information when an account is created is stored in the Firebase database and is referenced from there when called upon by the back-end code. User authentication is also controlled by Firebase, a user cannot login without a valid account stored on the database. Most importantly Firebase acts as the communication system for the application. Firebase acts a server for the driver and passenger applications to communicate between each other. While the information sharing is implemented as back-end code the communication system is done through Firebase. The majority of the back-end was implemented by Mr. Gahelrasoul, further detail will be said on this subject within his report.

*Previous Work*

Following the description of software used throughout the project, previous work is reviewed. There are numerous applications recognized in the field of taxi sharing. Such as, Uber, Lyft, Cabify, eCab, Blueline Taxi, and the Ottawa Taxi App. However, only a select few were used as examples or research tools when designing the application. The applications that became a research focus included: Uber, Lyft, Via, and Waymo One. These applications were all used as research tools at one point in development, each for their own reasons. Most of the research was used to determine how taxi sharing works at a fundamental level. For example, how the user profiles are set up, or what components are needed to determine a proper match, as well as what is a reasonable cost for a typical ride using these applications. All applications used have a taxi sharing component to them. While Uber and Lyft are apps in their own regard, they have since implemented a taxi sharing element similar to the goal of this project’s application.

To begin, the most popular modern application will be discussed. Uber currently holds the title as the most popular taxi application on the market. It is available in sixty-five countries as well as six hundred cities worldwide. There are approximately fifteen million trips completed each day using this application.[[6]](#footnote-7) Therefore, it was a prime research tool when designing this project. However, Uber is typically just a taxi service that passengers can use to request a ride for themselves, or for small groups of passengers, which will take them to their requested destination.[[7]](#footnote-8) The application being designed in this project is specifically a taxi sharing system. Meaning, Uber does not entirely fit the description of the project. This being said, during research it was found that Uber has a taxi sharing component available to their application. It is called UberPool, which is their variation of a taxi sharing application. It functions the way one would expect a taxi sharing application to function. A user can open the app and enter their destination and size of their group, (maximum 2 people), before requesting a ride.[[8]](#footnote-9) The system would then match them with a driver and send them to a pick-up location. During this waiting period other users are matched to the car until the vehicle is full. The application will also display the cost of the ride before pick-up occurs as well as the estimated travel time, so that the user knows the amount the ride will cost before getting in the Uber.[[9]](#footnote-10) During the ride the driver will either continuously pick up passengers along a similar route or drive certain passengers straight to their destination if there are no further customers. Once the destination is reached, the appropriate passenger exits the vehicle and is automatically billed via the payment method listed on the application.8 This ride system greatly inspired the system put in place by the project application. The only element left out was displaying the estimated duration. The research gained from this application, other than the overall system, aided with back-end development. Analyzing Uber’s system aided in determining the amount of luggage a user would be allowed to bring with them. It also aided in determining how to drop off riders, as well as determining how many passengers are allowed per group. It was determined from comparing the differences between Uber and UberPool that while UberPool takes slightly longer to complete a trip, it is always up to 65% cheaper than taking a regular Uber.[[10]](#footnote-11)

To continue, Lyft was also used as a research tool for this project. Lyft is a popular North American taxi application. It is currently being utilized in 350 US cities as well as the Canadian cities, Toronto and Ottawa. While Uber dominates the market with sixty-five percent of all application-based taxis, Lyft holds onto a significant thirty-one percent of that market.[[11]](#footnote-12)



**Figure 1:** Ridesharing Market Shares[[12]](#footnote-13)

Therefore, they were an important source of research for this project. Furthermore, while researching the company it was found that Lyft has also introduced their own variation on taxi sharing, called Lyft Line.[[13]](#footnote-14) Lyft Line essentially functions similarly to UberPool, however, the main difference found here was that while Uber will slowly match users with other passengers along the way (picking up and dropping off passengers as the ride progresses), Lyft will attempt to match the group of passengers together at the beginning of the ride and not pick up additional passengers once the drive has begun. However, if a passenger requests a ride while a trip is ongoing, and the pick-up and drop-off is optimal, then the Lyft driver will be instructed to pick them up along the way.[[14]](#footnote-15) This feature was found to be similar to the idea that this project was trying to achieve and thus the project was implemented in a similar fashion. Similarities were also observed between the two systems. For example, there were similarities in how they displayed the approximate cost and duration of the ride before the trip begun. Another example was how the application determined the optimal route based on all the individuals’ desired destinations. This means that while a user may be picked up first they may not necessarily be dropped off first, they will be dropped off in the most optimal order possible. These two apps heavily influenced the back-end system that the project was aiming for, therefore there are distinct similarities with regards to the order actions are carried out during a match. With regards to user interface, the design of this project was not nearly as influenced by these apps as the back-end system was. Both Uber and Lyft have their own distinct way of calculating driver payout as well as displaying information to the user.[[15]](#footnote-16) Both applications have a visible road map as the background with their assortment of elements organized on top of the map. This is the only similarity with regards to the user interface. The project’s application once logged into, has a map as the background with elements organized on top of it. However, the design of the application is unique and not based off one of these popular applications.

Moving away from the popular taxi sharing options, another application that influenced the project will be discussed. Via is a rather unknown alternative to Uber and Lyft. It currently makes up one percent[[16]](#footnote-17) of taxi sharing application services and is still only available in three cities: Chicago, Washington D.C., and New York City. but this was not a deterrent to using it as a research aspect. Via is not technically a taxi sharing application, but rather a shuttle service.[[17]](#footnote-18) Users with the application will see specific pick-up and drop-off locations and will be able to ride the shuttle for a small fee. This is similar to a city bus but without the uncertainty that the bus will never arrive. Furthermore, since the application is still relatively small, there is a large amount of customer support available to users, as the company does not have to deal with millions of requests daily. The main aspect that was taken from this service and integrated into the project was they way this application deals with pick-up. All passengers arrive at the same location and board the shuttle (or taxi, in terms of the project), at the same time. The goal of the project was to achieve this functionality; not to have to make multiple pickup stops along the route. Therefore, this element was integrated during development. To continue, since Via is closer to a shuttle service than a taxi service, it typically has the same day-to-day customers. It is a popular method for students to get to school or employees to get to work everyday. For this reason, Via attempts to develop a connection between riders and drivers, as the same driver will typically transport the same passengers each day. Via attempts to create a more enjoyable commute for all individuals involved, instead of the commute just having to be a necessary component of someone’s day.[[18]](#footnote-19) Finally, Via offers incentives to users on top of the transportation. This feature is an aspect lacking from the applications of the competitors. The incentives within their service has to do with the cost of each ride. As drivers transport more individuals over a consistent time period they begin to earn a higher income than when they first started in the employ. Furthermore, as a passenger there is a similar system in place. The more a user rides with Via the smaller the regular fare becomes. This became a noticeable feature that was under consideration to be applied to the project.

To conclude this section, there is one additional application that must be discussed. As this project’s motivation is for it to evolve into an autonomous taxi service it would be unjust to ignore the autonomous taxi service currently in development. Waymo One began as Google’s self-driving car project in 2009 and has since evolved into a full autonomous taxi service. The application is currently only in deployment in the Metro Phoenix area of Phoenix, Arizona with hopes that it will be rolled out further in the future.[[19]](#footnote-20) The application works similarly to other taxi applications where a user enters their pick-up and drop-off locations before requesting a ride. The application will then display the given fee for the ride before the user confirms their ride, in order to make sure the user is comfortable paying the given amount. The vehicle will then arrive where specified and carry passengers to their requested destination. All of this is accomplished without a driver in the vehicle. Passengers of this service always know what to expect since every ride is identical. The vehicles are always the same and carry the same branding, and if anything were deemed unsatisfactory during the ride or if users needed to get in contact with support, there is a built-in help system on the application. Users simply need to open the support tab on the application, and they will be able to discuss the issues with their ride to a live representative.18 While no features were implemented or inspired by aspects of this application it was still an interesting element found during research that deserved mentioning given the scope of this project.

To end this section, it can be concluded that there are numerous options available when it comes to designing the application for this project. A number of which had prominent influence during the design of the application, and a number of which did not have any influence whatsoever. However, regardless of the influence by outside applications, the project which will be discussed in the following two sections is entirely unique. As is the case with most elements in the world, each new age of technology will always incorporate influences from the past. However, while the back-end and flow of the application may be reminiscent of others, the design and feel of this application will be solely its own.

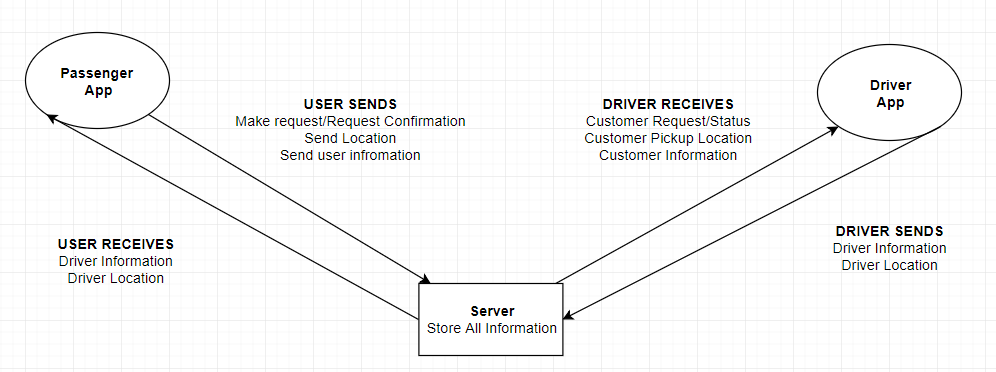
# **Chapter 3: Main Contribution and Design Decisions**

This chapter will further discuss the main goal of the project, delving into an in-depth discussion of the problem attempting to be implemented and the objectives of the application itself. This section will describe the initial plan that was had for the system, it is not meant to be an exact description of the completed software. Chapter four will discuss the implementation of the system designed in this project, this chapter will discuss the initial idea and what was set out to be created. Any elements that were not fully implemented will be explained later in chapter four. Following the description of the project objective this section will discuss the design of the application. The discussion will include the research done in terms of design choices, the influences as to why certain elements are as they are, as well as the design options that were determined through low fidelity prototype implementation. By the end of this section the goal is for the system of the application to be understandable. As well as the design choices of the application to make sense before delving into the actual implementation section.

*Main Objectives*

To begin, as stated before the goal of the project was to develop a taxi sharing application that was capable of matching users together and displaying the information properly. The app was designed to be centered around the Ottawa International Airport; this decision was made early in the development stage. As the project was to focus on design and research and less on back end implementation. Having a general pick up area aided in this simplicity. The app was to mirror the functionality of the apps currently in the world, such as Uber, Lyft, and Waymo. There were not exact specifications when it came to the exact functionality of the application. Just minimal requirements that are typical to every taxi sharing application. The main focus of the project was research based; how to design an application, what elements must go into creating one, how to achieve functionality with these given elements, and finally to have everything communicate as seamlessly as possible. Before discussing these research points some attention will be given to the requirements.

The main requirement was to have an application system that consisted of three elements. There would be two separate applications, a passenger system and a driver system, which would be connected by a back-end communication system. The requirement was to have these three elements communicating seamlessly. To have the passenger side send information to the driver; such as, when a customer is requesting a ride, who the customer is, where the customer is, and where the customer requests to be dropped off. This information was to be saved in the server before being sent to the driver. The driver side was to receive it and then the customer was to be prompted with the matched driver’s information. Furthermore, since the application was supposed to be a taxi sharing service, there was the idea that multiple passengers could be matched together. Once a request was made and the server matched a user to a driver, the application was then to display all the information it had collected on the screen in distinct areas. This was the main objective of the application, the minor details which will be discussed shortly were all research based. The main requirement was the to create and implement the triangle architecture that is quite reminiscent of a mediator design pattern. In which there are separate elements, in this case applications, which only communicate to a middle portion. The driver and passenger applications do not know about the other, they are only aware of the information they receive from the server. This was the main goal set out to be implemented. The rudimentary design of this system is included below:



**Figure 2:** Communication Diagram

Figure one depicts the layout of the system that was required to be designed, now to clarify the requirements tasked to this project was to implement the passenger end of this given system, as well as minimal back-end in the server side. The driver side and the majority of the back-end was tasked Mr. Gahelrasoul’s project and thus will be discussed more in depth throughout his report.

With the main requirement of the system established the minor objectives will now be discussed. As this project was being done in tandem with another these requirements differ from the major system being designed and are more focussed on an individual app basis. The passenger application was the focus of this project, and consisted of two components, research and implementation. The research was two-fold, there was an element focused on how taxi sharing works, such as to determine what information was required in order to match passengers with taxis and secondly there was research based on how to implement a working system. Two of the applications which were mentioned in the background were the main focus of this research. Uber and Lyft both influenced the design of this application in multiple ways. Though that will be discussed momentarily. Before design is mentioned some attention must be paid to the requirement specified initially.

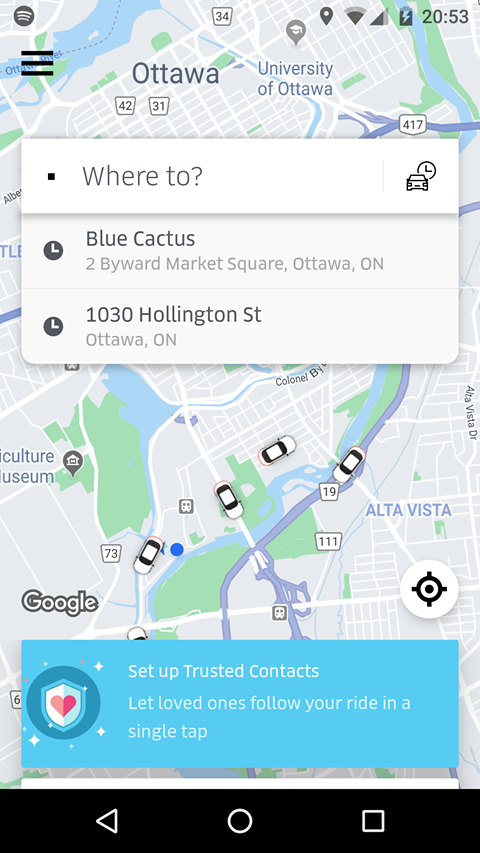
For the passenger application research had to be done to determine what was needed to be included into a taxi sharing application. This was done through the process of downloading the typical taxi sharing apps and exploring their user creation screens. Therefore, credit must be given to Uber and Lyft for the inspiration towards the user fields. The elements under question when performing this research were: what information the user must provide initially, what must they provide before every ride, how is cost calculated for a ride, and how the layout of the information is during all stages.[[20]](#footnote-21) Both apps yielded similar results through this process. With regards to information that is provided on sign up, each user had to provide their name, their phone number, some form of payment method; be it a credit card or debit card, and some form of login identification such as a username and password. Furthermore, optionally the users could provide a profile picture.[[21]](#footnote-22) These elements were all incorporated into the design of the passenger application and will be discussed in detail shortly. To continue, a portion of the research was to determine the elements for user matching within a taxi sharing service. Now both of the applications in question keep their matching algorithms secret therefore the elements chosen for this project were based more on common sense than on specific details. However, certain public elements were found to be similar between the systems. Both applications take into consideration the number of people within the group placing the ride request. Uber limits the number of people in a group using UberPool to two, this allows for additional people to be picked up during the ride.[[22]](#footnote-23) Furthermore, since taxi sharing is all about convince distance was also found to be a factor in the matching research. The application will not place a user with a driver that is halfway across the city, it will always attempt to find the closest driver and set the user up with that individual. That way there are less wait times and the app can receive more requests across the day.[[23]](#footnote-24) The rest of the research on these application pertains to the design; therefore, it will be discussed momentarily. The final element to mention with regards to logistics research is the cost calculation. Since taxi sharing has a different cost calculation that typical taxi applications the formula used required some thought. What was found demonstrated that typical taxi services have a base fee, followed by a fee per kilometer traveled, and then there is a fee for waiting for passengers. Taxi sharing also includes a reduction based on the number of passengers that a ride is shared with.[[24]](#footnote-25) Therefore, the formula that was devised for this calculation was:

While this was the formula that was decided upon there were some difficulties when it came to implementation that will be discussed in chapter four, as well as the replacement that was decided upon for the visual aspect. This formula still remains in the implementation however it is not used.

Finally, with all the preliminary research discussed now the focus will shift to the research done on the implementation of the application. The implementation aspect focused on the design of the application itself. Firstly, to get it to be visually appealing when using it and secondly, to ensure it can function smoothly on Android devices. Both Uber and Lyft influence the design in their own ways with regards to this application. However, multiple design choices were unique choices in an attempt to make this application different from those already in development. The elements of influence will be discussed first.

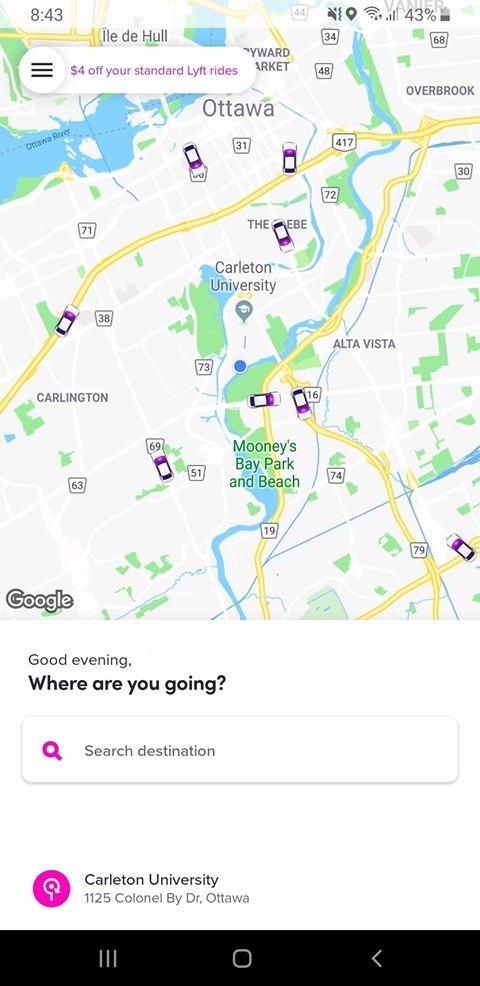
*Design Decisions*

The main elements of interest with regards to the design were how does the information display on the screen. The main requirement was to be able to connect users together and then display thig information on the screen, so the question became: how should this information be displayed? Was there a perfect method in an app that already exists, or would a combination of elements be the preferred decision. Through observing the Uber screen as well as the Lyft screen it was determined that a combination would be preferential. Refer to figure 2 below to begin.



**Figure 3:** Uber Pre-Ride Screen.

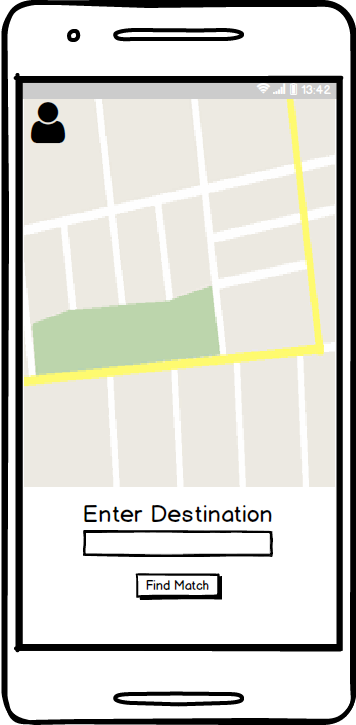
Figure 3 is the screen a user is presented with when loading the Uber application. It features a search bar at the top of the screen as well as a user menu in the top left corner. While the design is sleek and professional a concern stems from the fact that a large majority of the screen is covered with information the user may be uninterested in. Take the blue rectangle towards the bottom of the page. This is covering up a large portion of the map that could be helpful for the user to view if there are other drivers available. As well as the previous locations area under the search bar, this would be better served as a viewable option in the user menu instead of taking up a portion of the screen. This project did draw some influence from Uber, the user information location was a perfect location given the design elements currently in place for the application.

 Moving on from Uber, Lyft was the final app included in the design research. These two applications were the only two chose due to their popularity, as stated in chapter two these applications rank first and second respectively in the taxi application community. Anyhow, Lyft influenced the design of the application far more than Uber in a sense. While Uber tended to cover up a majority of the application’s map Lyft left it open for the user to view. Figure 4 below demonstrates this.

**Figure 4:** Lyft Opening Page

Figure 4 shows the screen a user is shown when logged into Lyft, once again it presents the user with a very professional design. A stark difference between Uber and Lyft is that the search area is at the bottom of the screen. As well as the previous locations listed below it, this design seemed a lot more space oriented and thus was used as inspiration for the project’s implementation. Furthermore, similar to Uber, the user profile information is placed in the top left corner of the screen. Further justifying the reasoning behind the project’s placement of it.

With these inspirations in mind the design of the application began to take shape. To ease in implementation further in the process a Balsamiq low-fidelity prototype was developed for the application. A low-fidelity prototype is essentially a drawing, or mock-up in this case, of the final design of a product. It gives the creator the ability to view the ideas before actually implementing them. Therefore, as this project was majority design focused based upon research it seemed to be the correct idea to test out the implementation before actually creating it.



**Figure 5:** Balsamiq Mockup of Pre-Match Screen

Figure 5 features the Balsamiq Mockup created for the first screen of the application once logging in. The entire Balsamiq creation is included within the appendix section of the report. As well as a control flow and legend for the different elements on the images. The control flow demonstrates how each screen of the design meshes together to form what will be the final application. At any rate, when comparing figure 5 to figures 3 and 4 there are certain similarities and certain differences. Firstly, the map takes up a majority of the screen, since it is a taxi application the map should be a main focus of the application. This is in contrast to Uber and similar the Lyft. Furthermore, the search area is at the bottom of the screen, once again similar to Lyft. However, absent from this design is the previous ride element, which was a feature that was not thought to be needed in the scope of the application. Since the application is set to always begin at a set location the history of rides was omitted. Finally, the user profile area which was similar in both Uber and Lyft was changed slightly in the project’s design. As Uber and Lyft are more robust applications with a greater number of features available to the user,[[25]](#footnote-26) their user menu needed to reflect this ability. The app designed for this project had a much simpler scope in mind for the design. Therefore, this simplicity took the form of the user menu becoming a button that would display a user profile. These elements discussed form the basis of the main similarities and differences between the applications researched, and also form the basis of the major design decisions for the application. Each other screen follows the approach of keeping the map as the main focus of the screen and attempts to keep the information to the bottom area as to not impede on the visuals in the map.

With these requirements in mind, as well as the design decisions chosen for the application, all the information has been stated as to why decisions were made. Following this point the actual implementation of the application will be discussed. Detailing how the decisions and objectives were carried out. As well as, if they were modified or eliminated once development began.

# **Chapter 4: Implementation**

# **Chapter 5: Conclusion**

# **References**

Balsamiq Studios. (2019). Balsamiq Mockups 3 Application Overview - Balsamiq for Desktop Documentation | Balsamiq. Retrieved March 30, 2019, from https://balsamiq.com/wireframes/desktop/docs/overview/

GmbH, S. M. (2019). Get your Taxi Fare now! Retrieved January 15, 2019, from https://www.taxi- calculator.com/

Google. (2019). Firebase Overview. Retrieved March 30, 2019, from https://firebase.google.com/

Guzman, Z. (2019, March 27). Why Lyft's IPO may be more attractive than Uber's. Retrieved April 19, 2019, from https://finance.yahoo.com/news/lyfts-ipo-may-attractive-ubers-193328062.html

Iqbal, M. (2019, March 25). Lyft Revenue and Usage Statistics (2019). Retrieved March 30, 2019, from http://www.businessofapps.com/data/lyft-statistics/

Iqbal, M. (2019, February 27). Uber Revenue and Usage Statistics (2018). Retrieved March 30, 2019, from http://www.businessofapps.com/data/uber-statistics/

Lyft, Inc. (2018). About Shared rides. Retrieved March 30, 2019, from https://help.lyft.com/hc/en- ca/articles/115013078848-About-Shared-rides

Lyft, Inc. (2019). Ride with Lyft – Friendly drivers and serious safety. Retrieved March 30, 2019, from https://www.lyft.com/rider

Pratap, M. (2018, October 26). How to Build an App like Uber? (Complete Guide) - Uber Clone. Retrieved January 30, 2019, from https://www.engineerbabu.com/blog/how-to-build-an-app-like- uber/

RideShareApps. (2015, August 24). Lyft Line - What Is It And How Does It Work? Retrieved March 30, 2019, from https://rideshareapps.com/lyft-line/

Uber. (2017). UberPOOLSharing is saving. Retrieved March 30, 2019, from https://www.uber.com/info/uberpool-nj/

Uber. (2019). About Uber - Our Story - Vision for Our Future. Retrieved March 30, 2019, from https://www.uber.com/en-CA/about/

Uber. (2019). UberPool vs. UberX - How Does UberPool Work? Retrieved March 30, 2019, from https://www.uber.com/en-CA/ride/uberpool/? state=G7iYn0jH75cgczSfSkAjBQ4xyOieMG7m\_\_FlamxG93o=&\_csid=gAM22lbigQKqssDDO uFa4w#\_

UberLyftDriver. (2017, October 20). Changes to Lyft Line Payment. Retrieved March 30, 2019, from https://www.ridesharingforum.com/t/changes-to-lyft-line-payment/131

UberLyftDriver. (2018, March 04). Via, the Alternative to UberPOOL and Lyft Line. Retrieved March 30, 2019, from https://www.ridesharingforum.com/t/via-the-alternative-to-uberpool-and-lyft- line/773

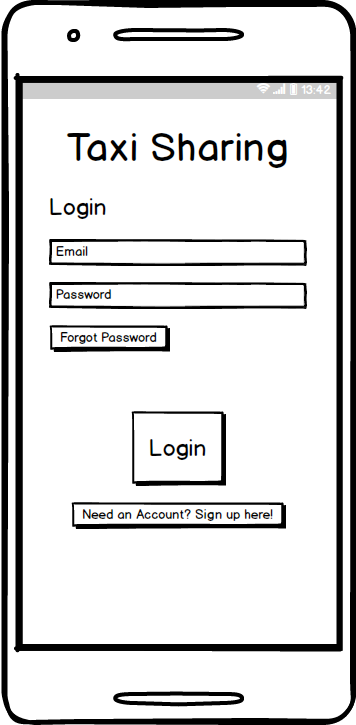
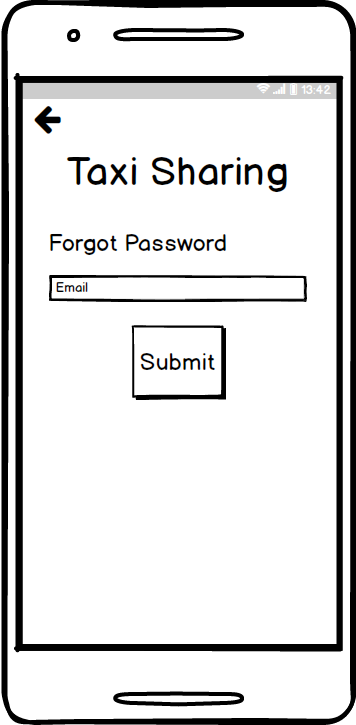
Technologies, U. (2019, January). Learning the interface. Retrieved April 18, 2019, from https://docs.unity3d.com/Manual/LearningtheInterface.html

Walter, D. (2018, October). What is Android Studio? - Definition from WhatIs.com. Retrieved March 30, 2019, from https://searchmobilecomputing.techtarget.com/definition/Android-Studio

Waymo. (2018). Waymo – Waymo. Retrieved March 30, 2019, from https://waymo.com/

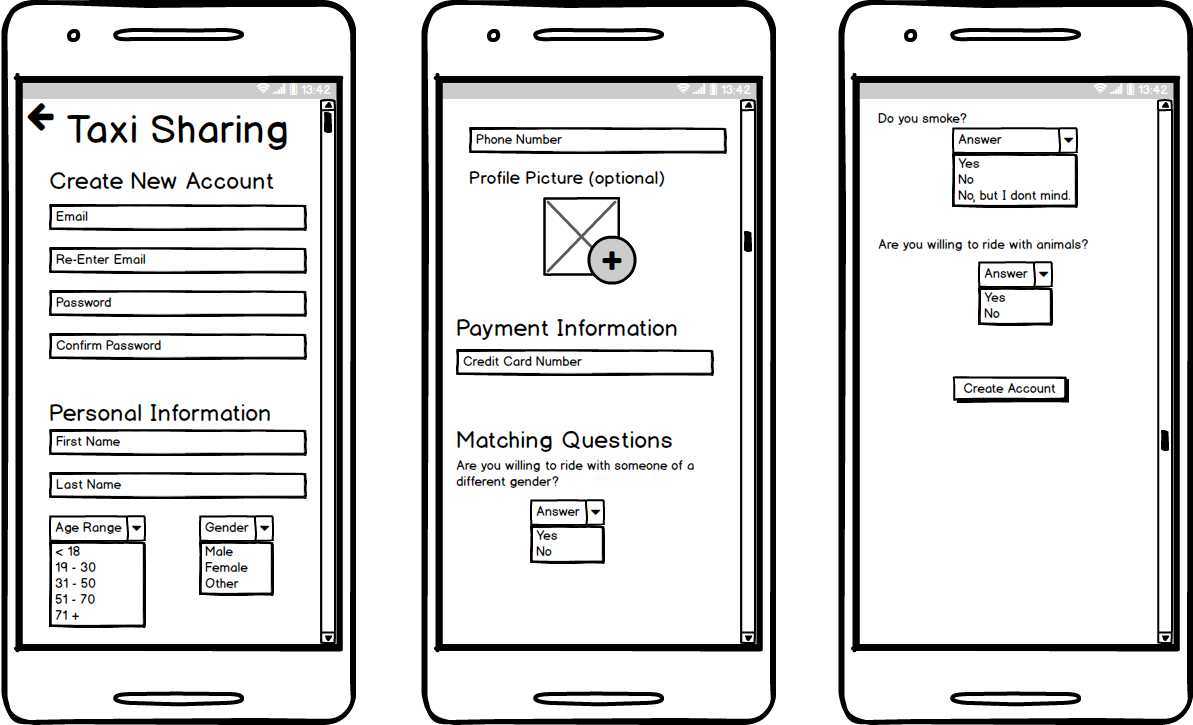
# **Appendix**

|  |  |
| --- | --- |
| **Legend for Mockups** | |
| Symbol | Meaning |
| Red Arrow | Button clicked leads to the screen the arrow is pointing to. |
| Red Star | Buttons with this symbol lead to the Pre-Match Screen. |
| Blue Star | Buttons with this symbol lead to the Login Screen. |
| Red X | Feature was removed during development. |
| Green Star | Button with this symbol lead to Pre-Match Screen 2. |
| Blue X | Feature was designed but not implemented. |

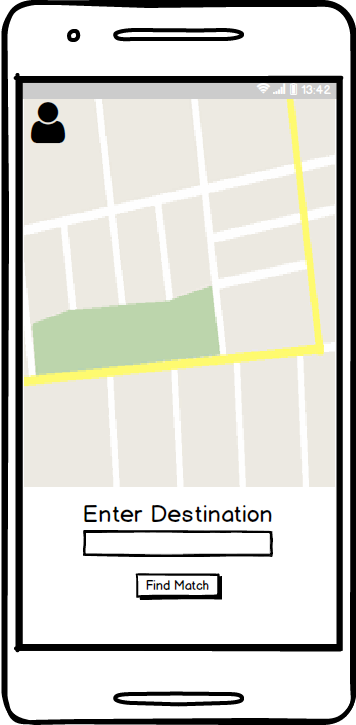
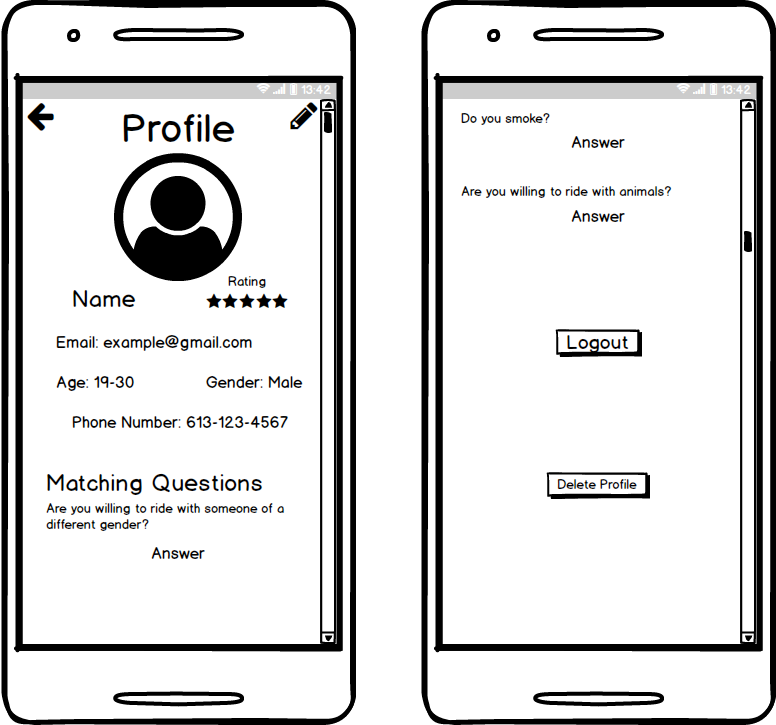
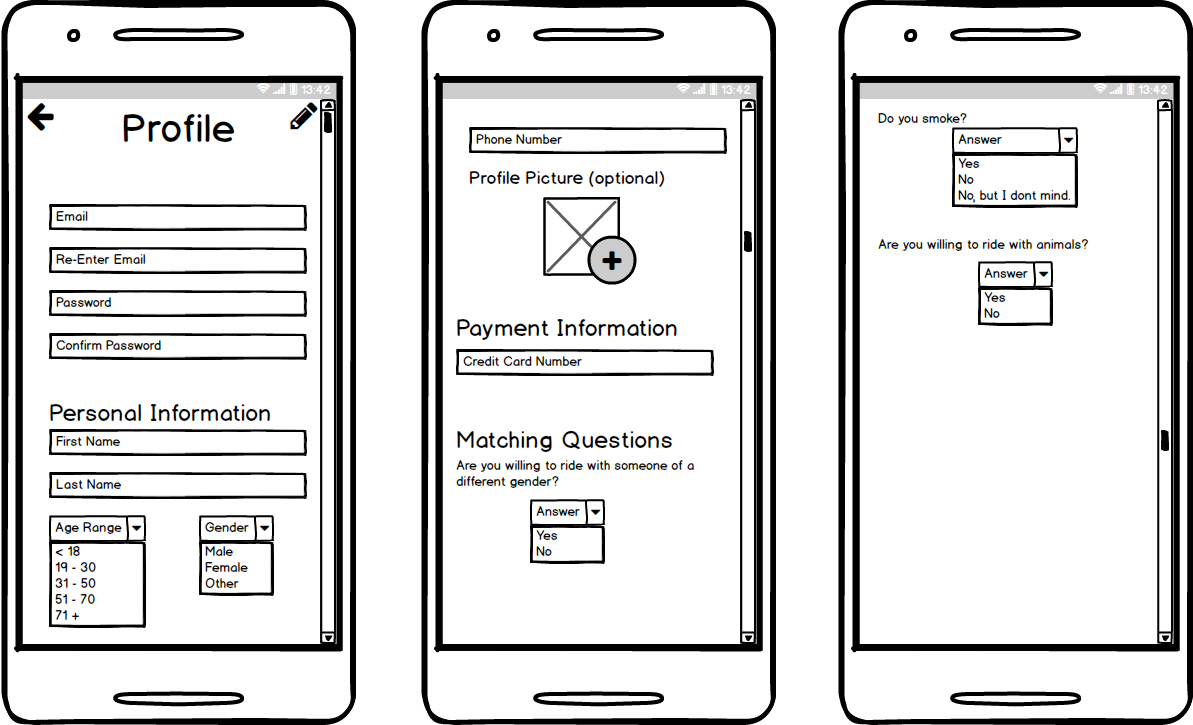
*Balsamiq Mockups*

Forgot Password

Login Screen



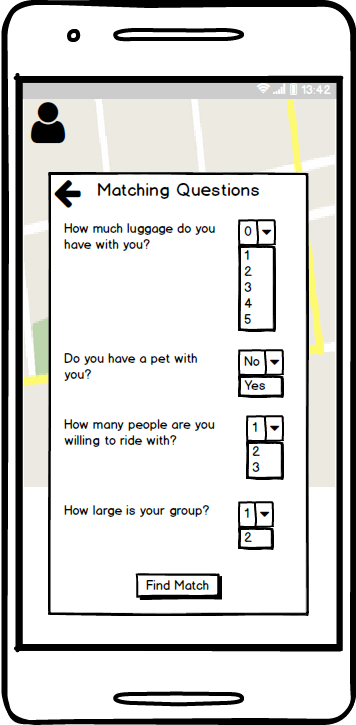
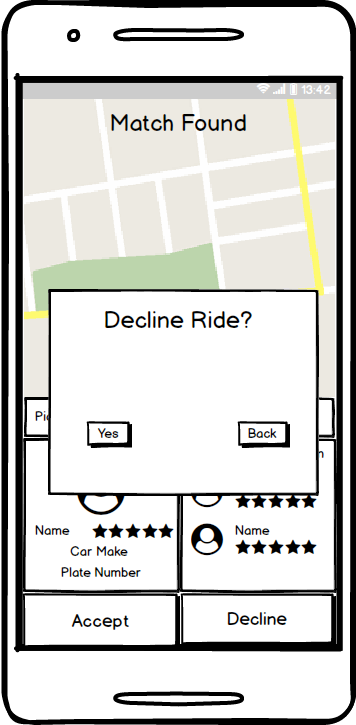
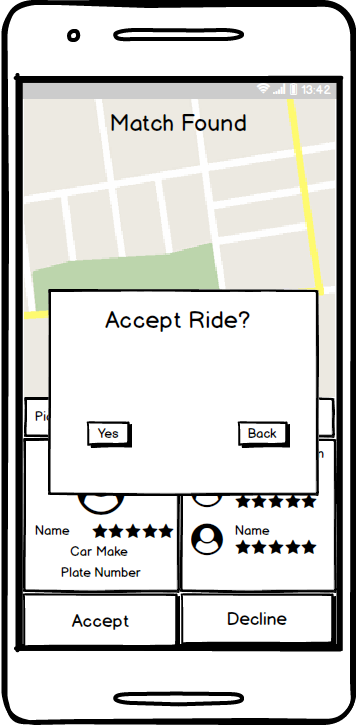
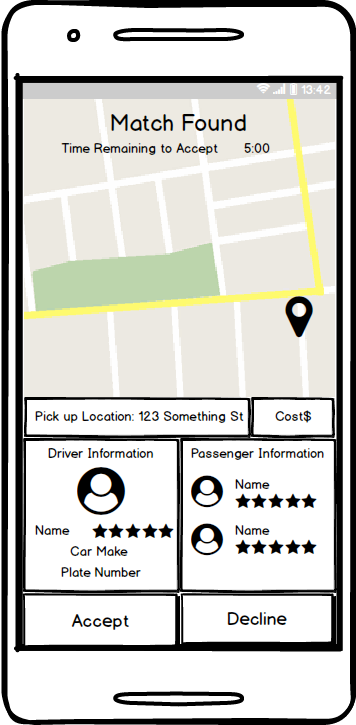
Create Account



Edit Profile

User Profile

Pre-Match Screen

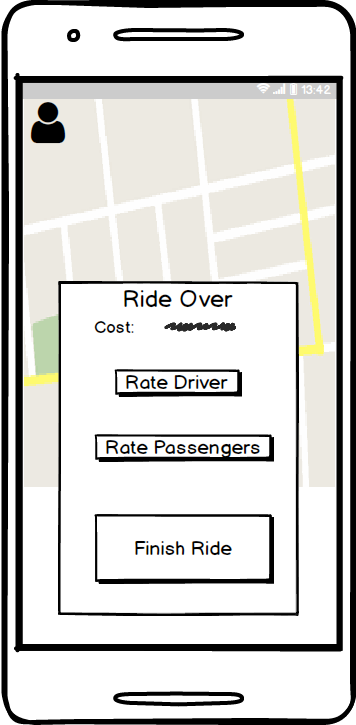
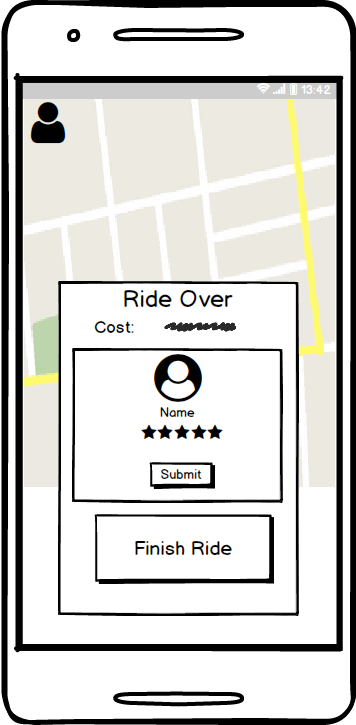
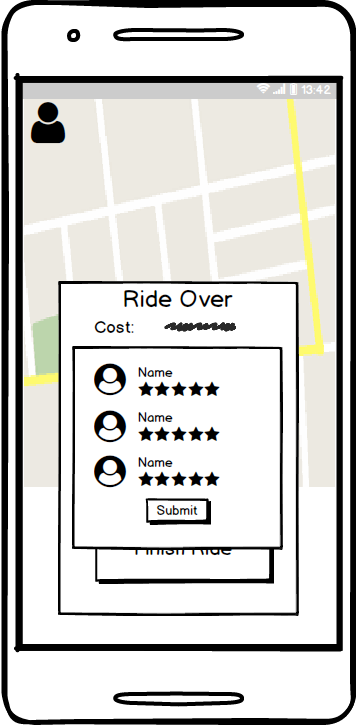


There will be a slight delay in this transition while the application references the database and find the match.

During Ride Screen

Match Found Screen

Pre-Match Screen 2



After the ride has concluded and the passenger has been dropped off the app screen will change to this view.

Rate Passengers Prompt

Rate Driver Prompt

Post-Ride Screen

*Android Studio Design*

1. Balsamiq Studios. (2019). Balsamiq Mockups 3 Application Overview - Balsamiq for Desktop Documentation | Balsamiq. Retrieved March 30, 2019, from https://balsamiq.com/wireframes/desktop/docs/overview/ [↑](#footnote-ref-2)
2. Walter, D. (2018, October). What is Android Studio? - Definition from WhatIs.com. Retrieved March 30, 2019, from https://searchmobilecomputing.techtarget.com/definition/Android-Studio [↑](#footnote-ref-3)
3. Technologies, U. (2019, January). Learning the interface. Retrieved April 18, 2019, from https://docs.unity3d.com/Manual/LearningtheInterface.html [↑](#footnote-ref-4)
4. Walter, D. (2018, October). What is Android Studio? - Definition from WhatIs.com. Retrieved March 30, 2019, from https://searchmobilecomputing.techtarget.com/definition/Android-Studio [↑](#footnote-ref-5)
5. Google. (2019). Firebase Overview. Retrieved March 30, 2019, from https://firebase.google.com/ [↑](#footnote-ref-6)
6. Iqbal, M. (2019, February 27). Uber Revenue and Usage Statistics (2018). Retrieved March 30, 2019, from http://www.businessofapps.com/data/uber-statistics/ [↑](#footnote-ref-7)
7. Uber. (2019). About Uber - Our Story - Vision for Our Future. Retrieved March 30, 2019, from https://www.uber.com/en-CA/about/ [↑](#footnote-ref-8)
8. Uber. (2019). UberPool vs. UberX - How Does UberPool Work? Retrieved March 30, 2019, from https://www.uber.com/en-CA/ride/uberpool/?state=G7iYn0jH75cgczSfSkAjBQ4xyOieMG7m\_\_FlamxG93o=&\_csid=gAM22lbigQKqssDDOuFa4w#\_ [↑](#footnote-ref-9)
9. Uber. (2017). UberPOOLSharing is saving. Retrieved March 30, 2019, from https://www.uber.com/info/uberpool-nj/ [↑](#footnote-ref-10)
10. Uber. (2017). UberPOOLSharing is saving. Retrieved March 30, 2019, from https://www.uber.com/info/uberpool-nj/ [↑](#footnote-ref-11)
11. Iqbal, M. (2019, March 25). Lyft Revenue and Usage Statistics (2019). Retrieved March 30, 2019, from http://www.businessofapps.com/data/lyft-statistics/ [↑](#footnote-ref-12)
12. Guzman, Z. (2019, March 27). Why Lyft's IPO may be more attractive than Uber's. Retrieved April 19, 2019, from https://finance.yahoo.com/news/lyfts-ipo-may-attractive-ubers-193328062.html [↑](#footnote-ref-13)
13. RideShareApps. (2015, August 24). Lyft Line - What Is It And How Does It Work? Retrieved March 30, 2019, from https://rideshareapps.com/lyft-line/ [↑](#footnote-ref-14)
14. Lyft, Inc. (2018). About Shared rides. Retrieved March 30, 2019, from https://help.lyft.com/hc/en-ca/articles/115013078848-About-Shared-rides [↑](#footnote-ref-15)
15. UberLyftDriver. (2017, October 20). Changes to Lyft Line Payment. Retrieved March 30, 2019, from https://www.ridesharingforum.com/t/changes-to-lyft-line-payment/131 [↑](#footnote-ref-16)
16. Iqbal, M. (2019, March 25). Lyft Revenue and Usage Statistics (2019). Retrieved March 30, 2019, from http://www.businessofapps.com/data/lyft-statistics/ [↑](#footnote-ref-17)
17. UberLyftDriver. (2018, March 04). Via, the Alternative to UberPOOL and Lyft Line. Retrieved March 30, 2019, from https://www.ridesharingforum.com/t/via-the-alternative-to-uberpool-and-lyft-line/773 [↑](#footnote-ref-18)
18. UberLyftDriver. (2018, March 04). Via, the Alternative to UberPOOL and Lyft Line. Retrieved March 30, 2019, from https://www.ridesharingforum.com/t/via-the-alternative-to-uberpool-and-lyft-line/773 [↑](#footnote-ref-19)
19. Waymo. (2018). Waymo – Waymo. Retrieved March 30, 2019, from https://waymo.com/ [↑](#footnote-ref-20)
20. Pratap, M. (2018, October 26). How to Build an App like Uber? (Complete Guide) - Uber Clone. Retrieved January 30, 2019, from https://www.engineerbabu.com/blog/how-to-build-an-app-like-uber/ [↑](#footnote-ref-21)
21. Lyft, Inc. (2019). Ride with Lyft – Friendly drivers and serious safety. Retrieved March 30, 2019, from https://www.lyft.com/rider [↑](#footnote-ref-22)
22. Uber. (2019). UberPool vs. UberX - How Does UberPool Work? Retrieved March 30, 2019, from https://www.uber.com/en-CA/ride/uberpool/?state=G7iYn0jH75cgczSfSkAjBQ4xyOieMG7m\_\_FlamxG93o=&\_csid=gAM22lbigQKqssDDOuFa4w#\_ [↑](#footnote-ref-23)
23. RideShareApps. (2015, August 24). Lyft Line - What Is It And How Does It Work? Retrieved March 30, 2019, from https://rideshareapps.com/lyft-line/ [↑](#footnote-ref-24)
24. GmbH, S. M. (2019). Get your Taxi Fare now! Retrieved January 15, 2019, from https://www.taxi-calculator.com/ [↑](#footnote-ref-25)
25. Pratap, M. (2018, October 26). How to Build an App like Uber? (Complete Guide) - Uber Clone. Retrieved January 30, 2019, from https://www.engineerbabu.com/blog/how-to-build-an-app-like-uber/ [↑](#footnote-ref-26)