Trippin' Information System
Elaboration On

By: Tong, Lauren, Lenneth, Fahim, Mohamed

Table of Contents

Vision:

Our vision is to create an app that will display the cheapest trip for commuters. This app wille compare the prices of metro transit, uber, lyft and taxis. It will also

take account into the seasons and will list other types of transportation like electric scooters and bicycles. The app will update in real time, allowing users to see the shifts of availability depending on the time of day and location. The goal of our system is to make it easy for users to find the cheapest mode of transportation as easily as possible. With this we will be encouraging people to use public transportation more frequently if they observe the prices for each trip they make.

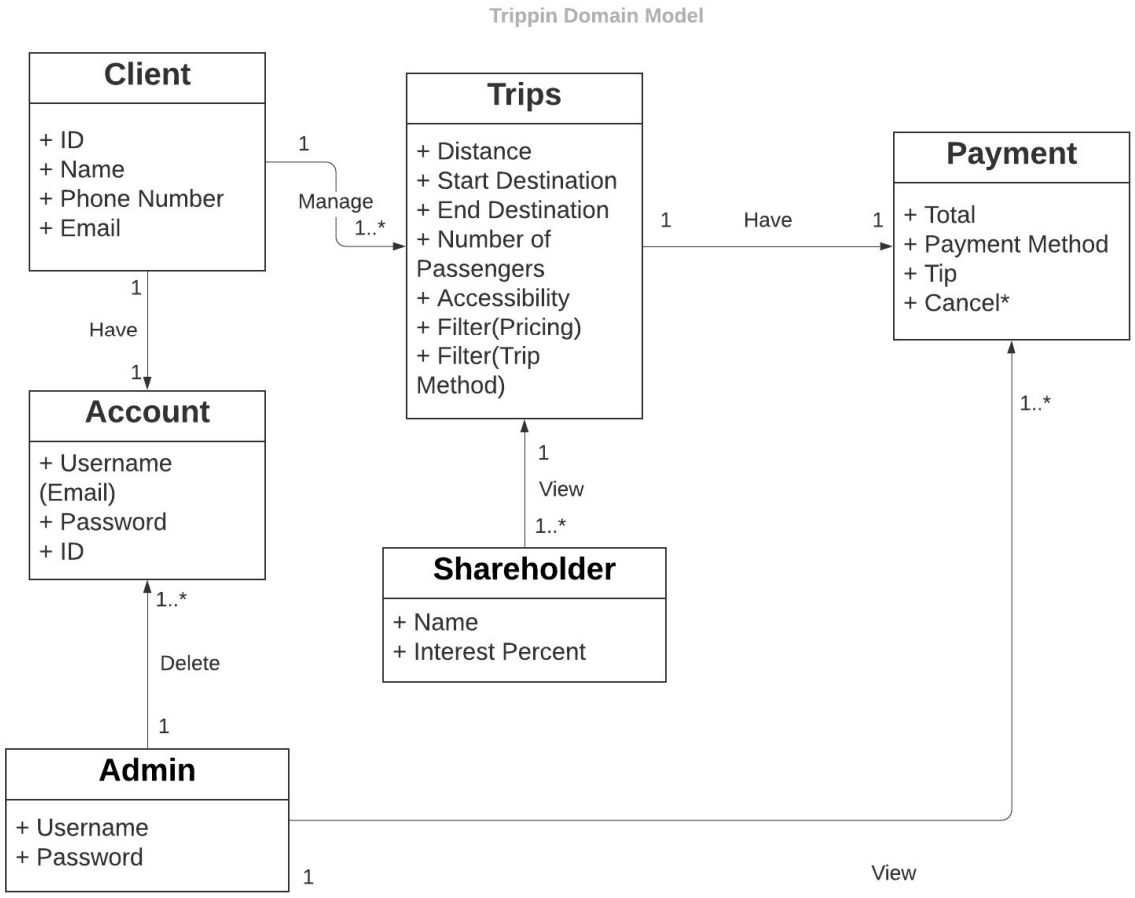
Scope:

By using GPS technology, we will display local bus pickup and drop off points, uber, taxi and lyft fees, and seasonal transportation. With this app we are hoping that users will become more aware of the various types of transportation available to them, and hopefully reduce possible carbon emissions and accidents. In addition to reducing traffic in the cities and slightly outside of the cities, giving users more options for transportation may reduce the number of car accidents related to intoxicated driving and distracted driving.

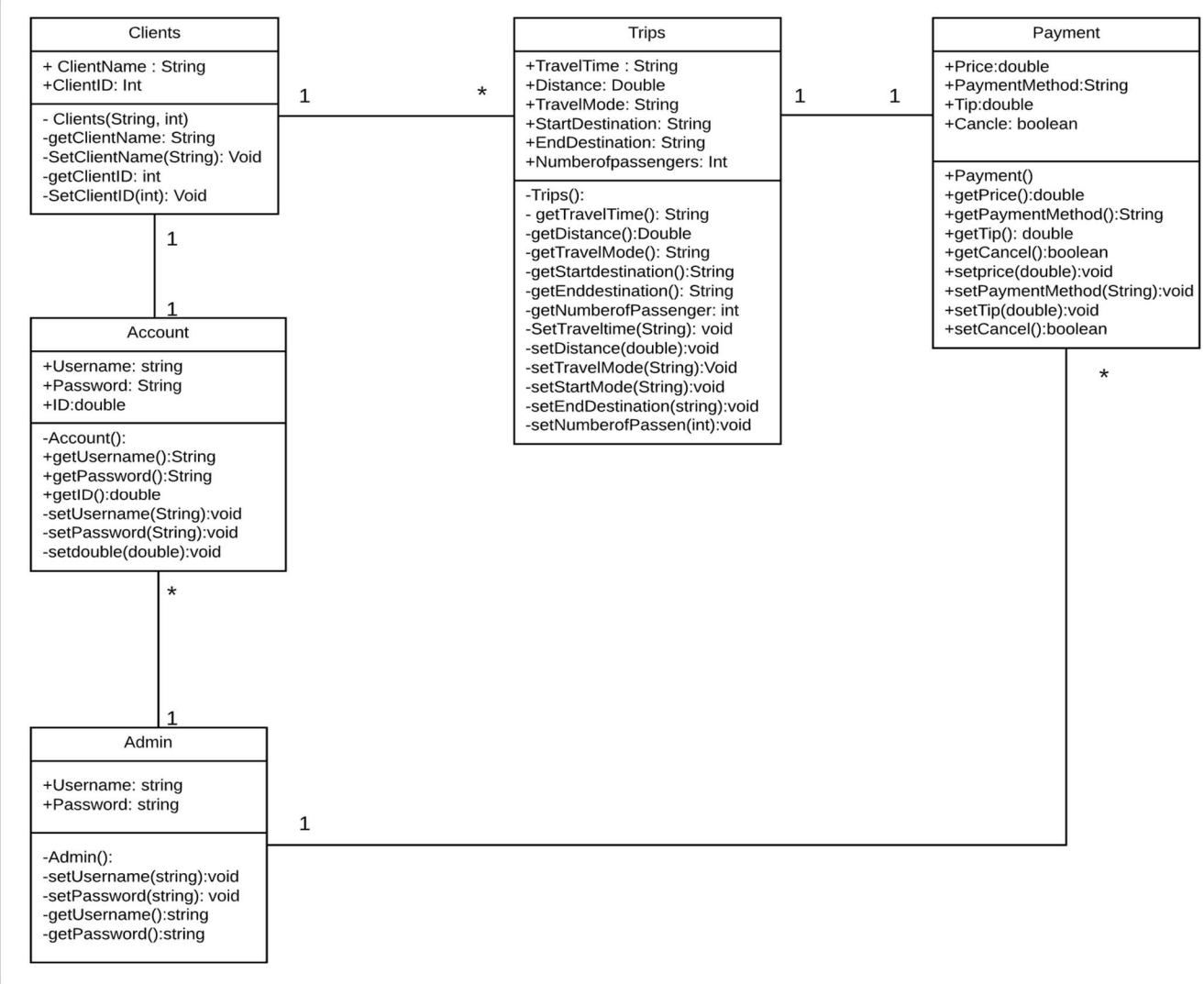
Business Case:

With using multiple apps for transportation and switching back and forth for looking for the cheapest transportation can be too time consuming and annoying. In addition, to having an all encompassing app, this may make it easier for users of all ages. Also, since most individuals need transportation quickly, the use of the seasonal transportation may make it quicker for an individual to get to another destination where they may be able to get an uber,lyft, taxi or public transportation easier.

Domain Model:



Class Diagram:



Software Architecture Methodology:

An agile methodology will be used in the development of this application. By using the Agile method, we are able to continue to create a user-oriented application that can continue to grow as our user base grows.

Software Architecture Methodology:

Presenting an architectural series of views: use case view, process view, sequence views, and implementation view will help give more an in depth understanding of the project. These views will be presented as UML diagrams and SSD's.

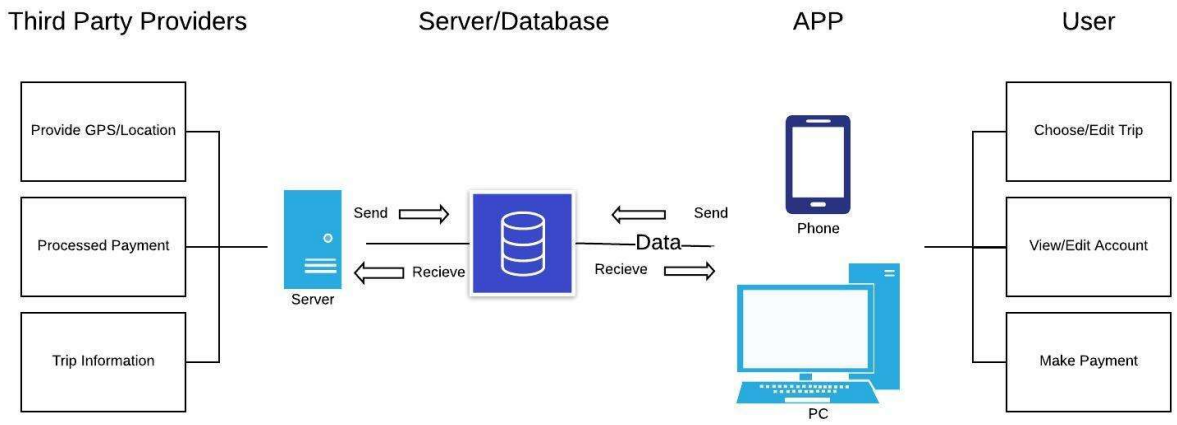
Architectural Goals and Restraints:

Some architectural goals would be: the app being able to handle multiple transactions and large amounts of traffic, an all functions of the app must be available and working correctly. A few architectural restraints would be: all credit card and financial and user information must be transmitted securely. In addition, ensuring that a database is secure and consistent in pick up location and times.

Logical View:

a. This view contains some of the most important classes and a brief description of each class.
    i. User:
        1. Contains class for information provided by actors used for updating accounts, making payments, and viewing trips. It should also contain classes for provided feedback.
    ii. Administrator:
        1. Contains classes for content management and profile management.
    iii. Investor:
        1. Contains classes for Trip management and view.

Software Architecture:



Data Model:

Class	Attributes	Data Type
Client	ID Name Phone Number Email	Int String Int Varchar (30)
Administrator	Username Password	Varchar (10) Varchar (30)
Account	Username Password ID	Varchar (10) Varchar (30) Int

Client Use Case Diagram:

Client Diagram

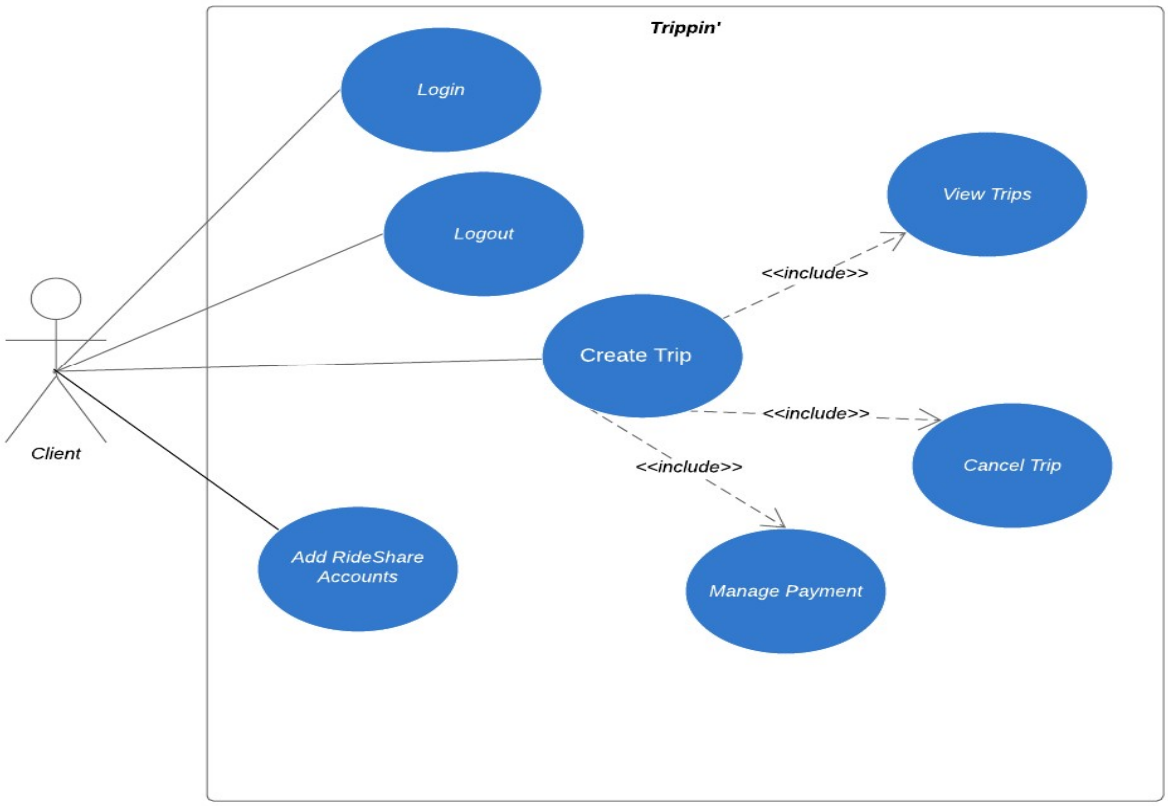


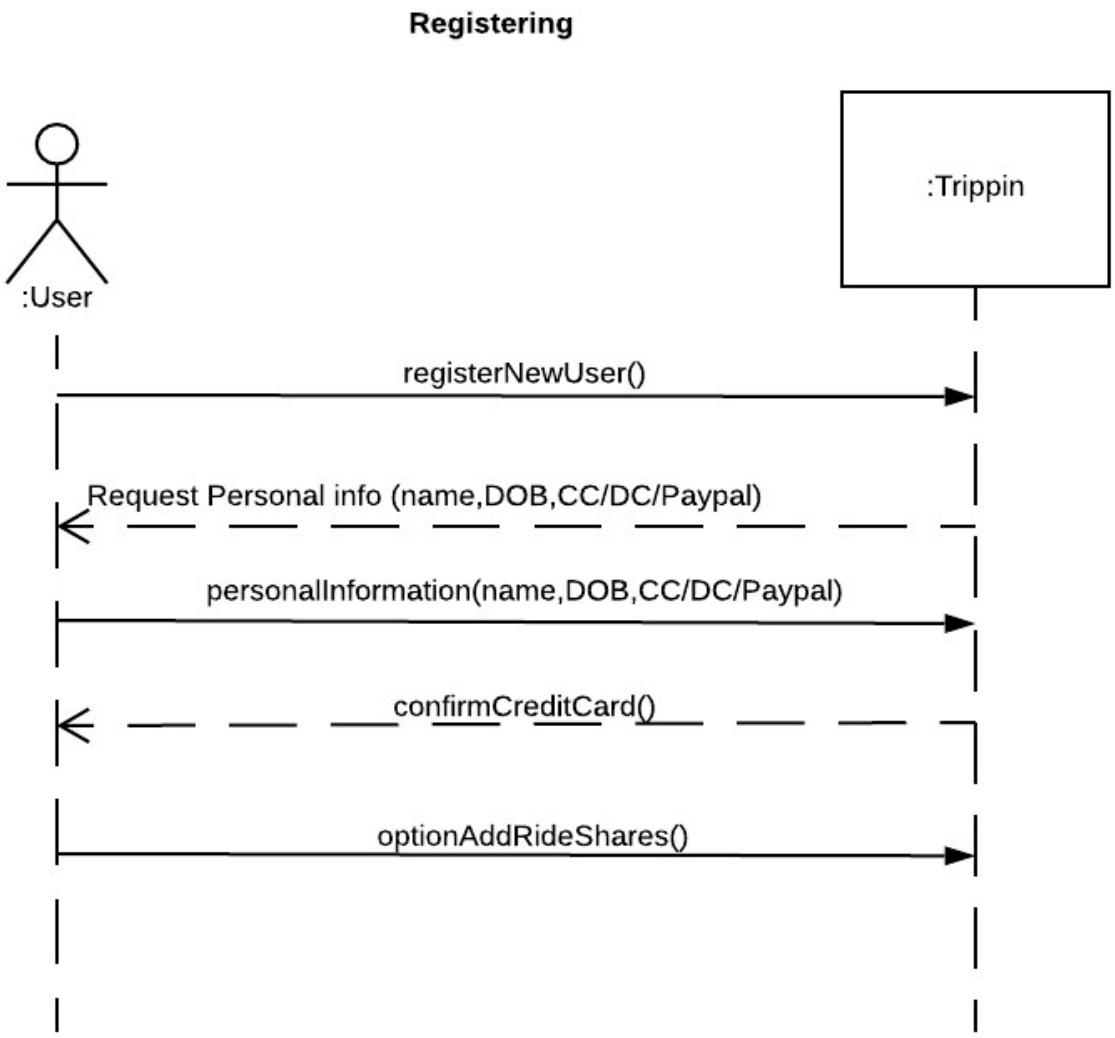
Figure 1: Use Case Diagram for client actor

Admin Use Case Diagram:

Use Case Diagram: Admin

Figure 1: Use Case Diagram for Admin actor

Registering Sequence:



Login Sequence:

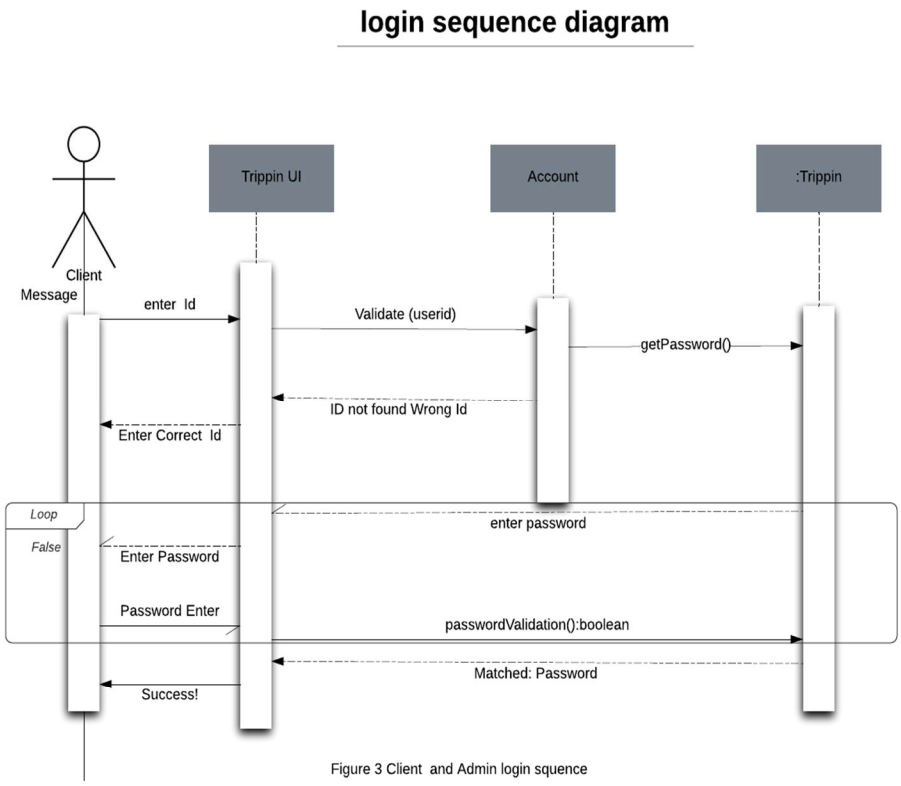


Figure 3 Client and Admin login squence