McMaster University

SFWR ENG 3A04: T1-05

Cabshare - Deliverable 3

Authors:

Kemal Ahmed	*ahmedks*	1150377
Mitchell Spector	*spectoriscool*	1208006
Sahajmeet Bhutta	*bhuttas*	1223708
Xue Lin	*liniscool*	1151043
Yuchen Xiao	*xiaoy23*	1212583

March 28, 2015

Contents

1 Introduction

1.1 Purpose

This document is intended to formalize the classes which will be used in the final application. It depicts states of controller classes, how use cases unfold through sequence diagrams, and attributes and methods of classes. Its intended audience is the development team and software engineers.

1.2 System Description

The system is designed to securely allow users to search for cabs with passengers willing to cabpool to a users destination, or include a stop along the way to a further destination, and to allow users to offer a cab for cabpooling. The system will connect commuters and cabpoolers to lower individual fares while still providing a profit for the cab company.

1.3 Overview

This document consists almost entirely of diagrams, including state charts, sequence diagrams, and a class diagram, respectively. State charts depict the various states induced in each controller class, sequence diagrams depict messages and data passed between classes as use cases are occuring, and the class diagram depics communication between classes.

2 State Charts for Controller Classes

2.1 CancelController

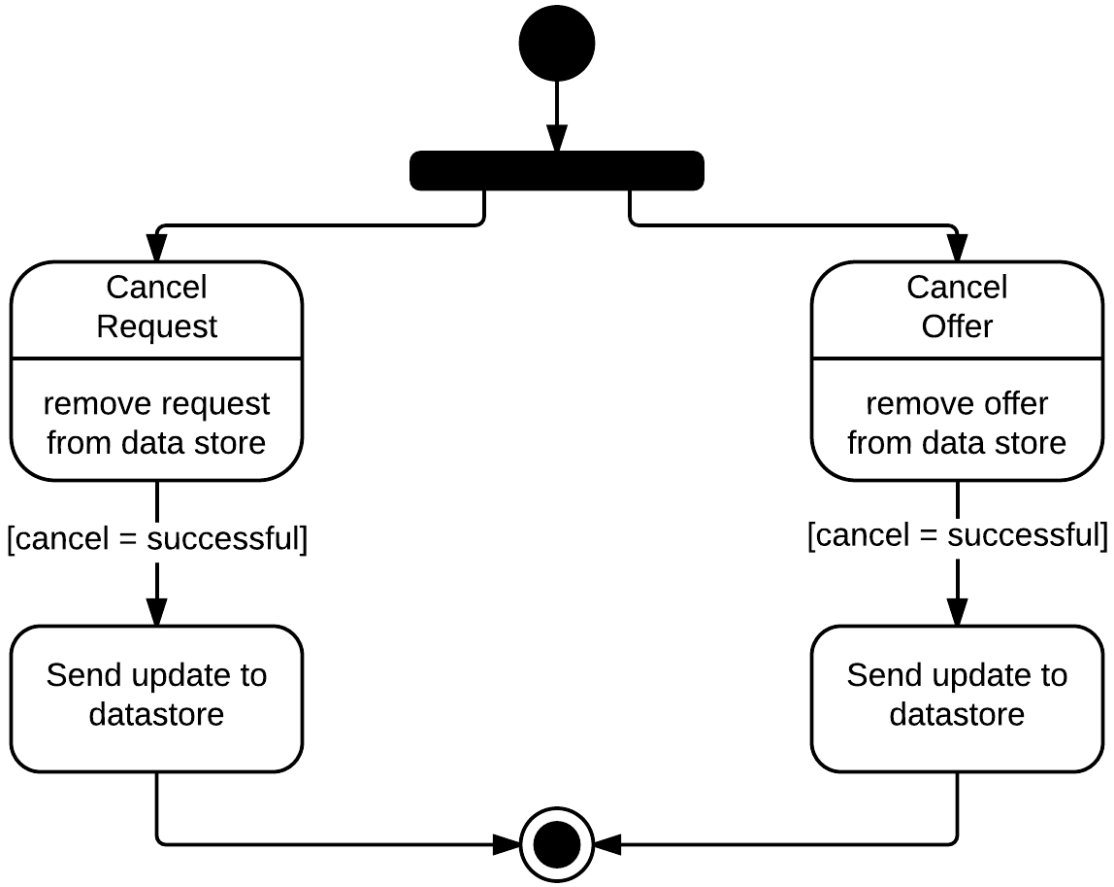

Figure 1: **State Diagram of CancelController** When user wants to cancel their request to join a cabpool.

2.2 Encryptor

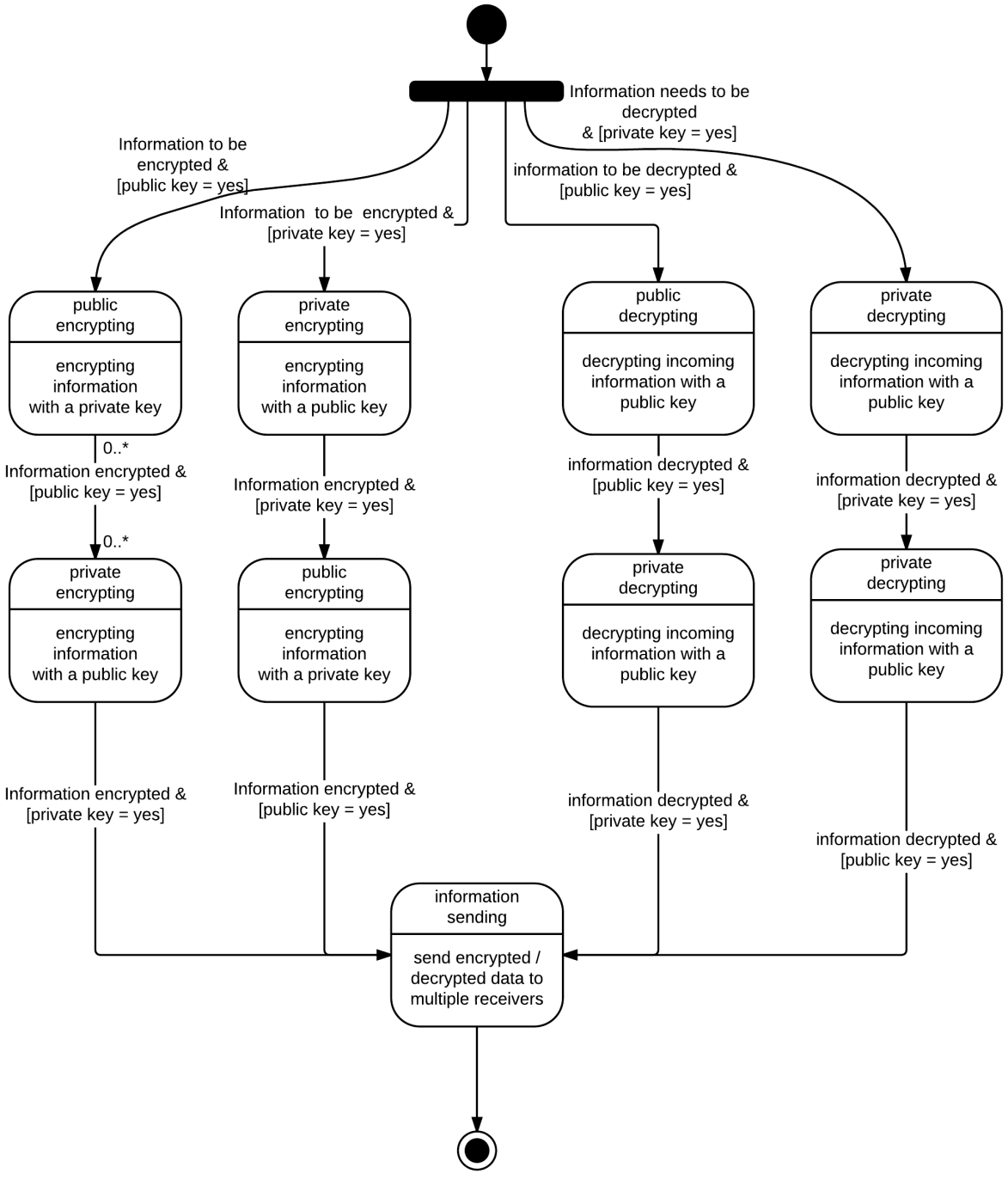

Figure 2: **State Diagram of Encryptor** Encrypting messages between people waiting for their ride and the people already in the cabpool.

2.3 LogoutController

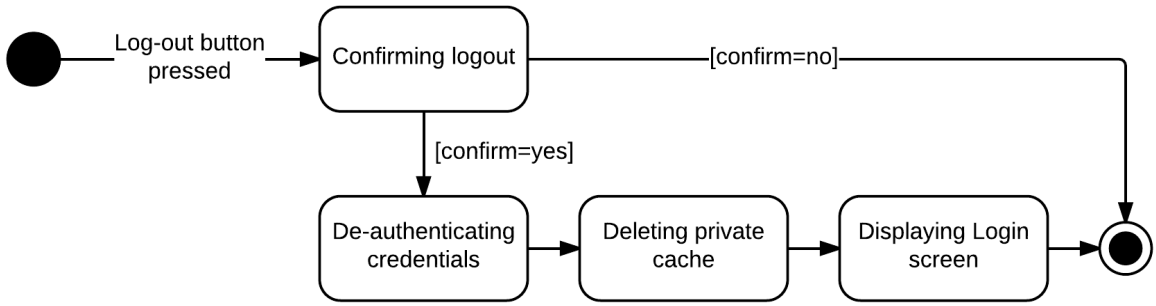

Figure 3: **State Diagram of LogoutController** When a user is logging out.

2.4 PaymentController

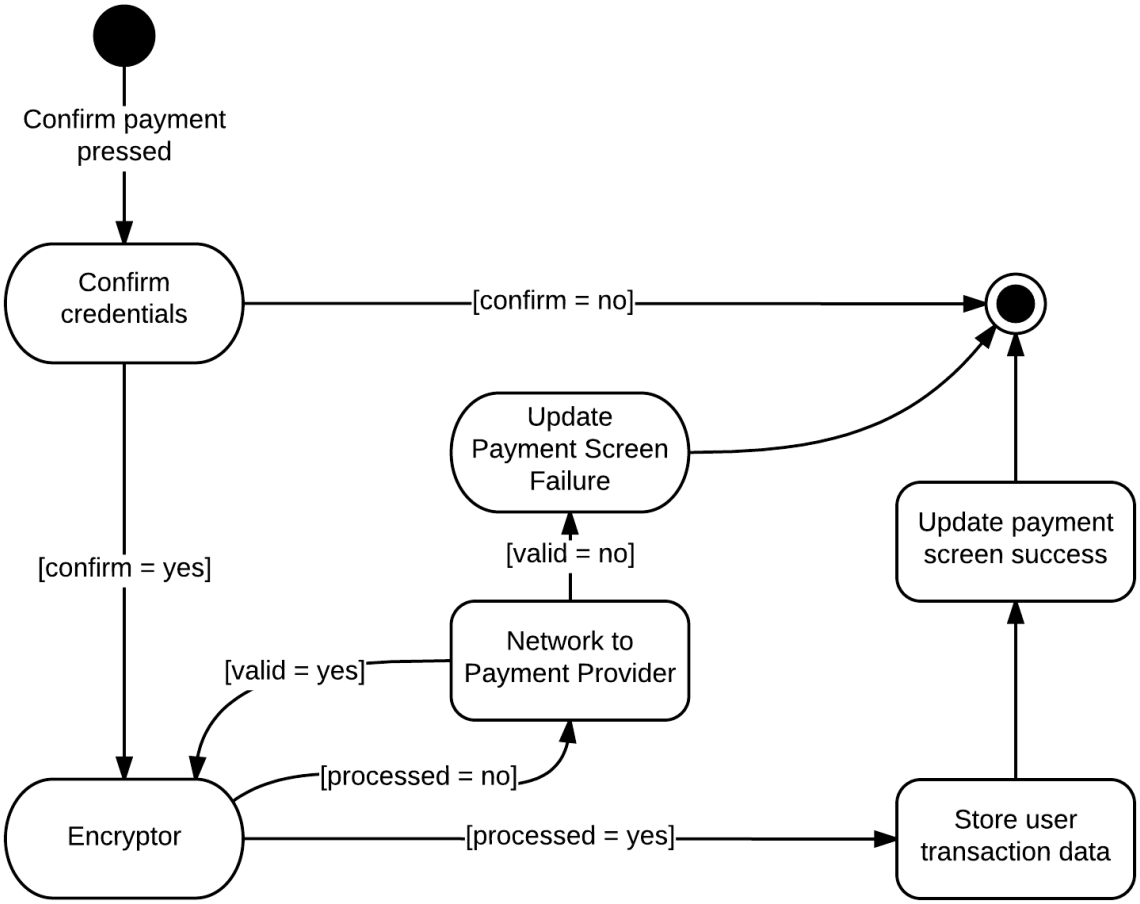

Figure 4: **State Diagram of PaymentController** When a user is paying for their ride.

2.5 RoutesController

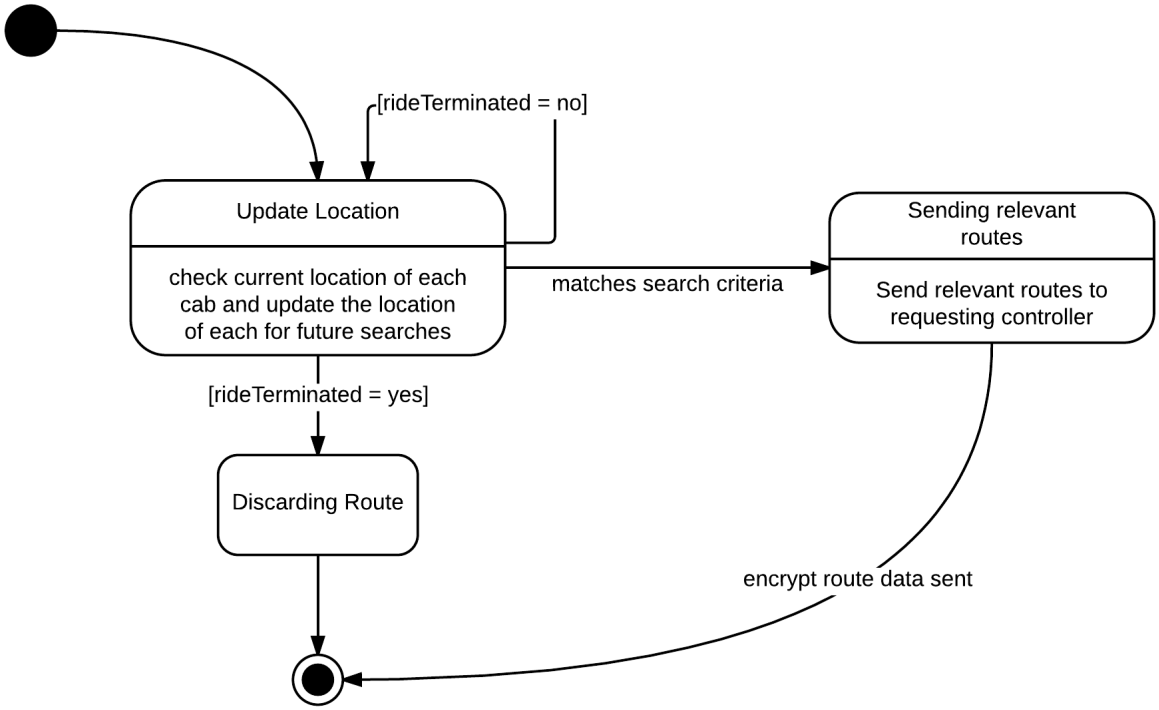

Figure 5: **State Diagram of RoutesController** When a user is determining potential routes to their destination.

2.6 ScanController

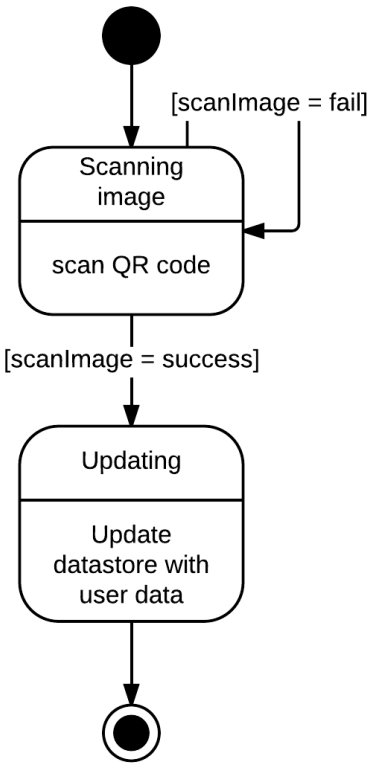

Figure 6: **State Diagram of ScanController** When the user enters the cab and scans the QR code.

2.7 SecurityController

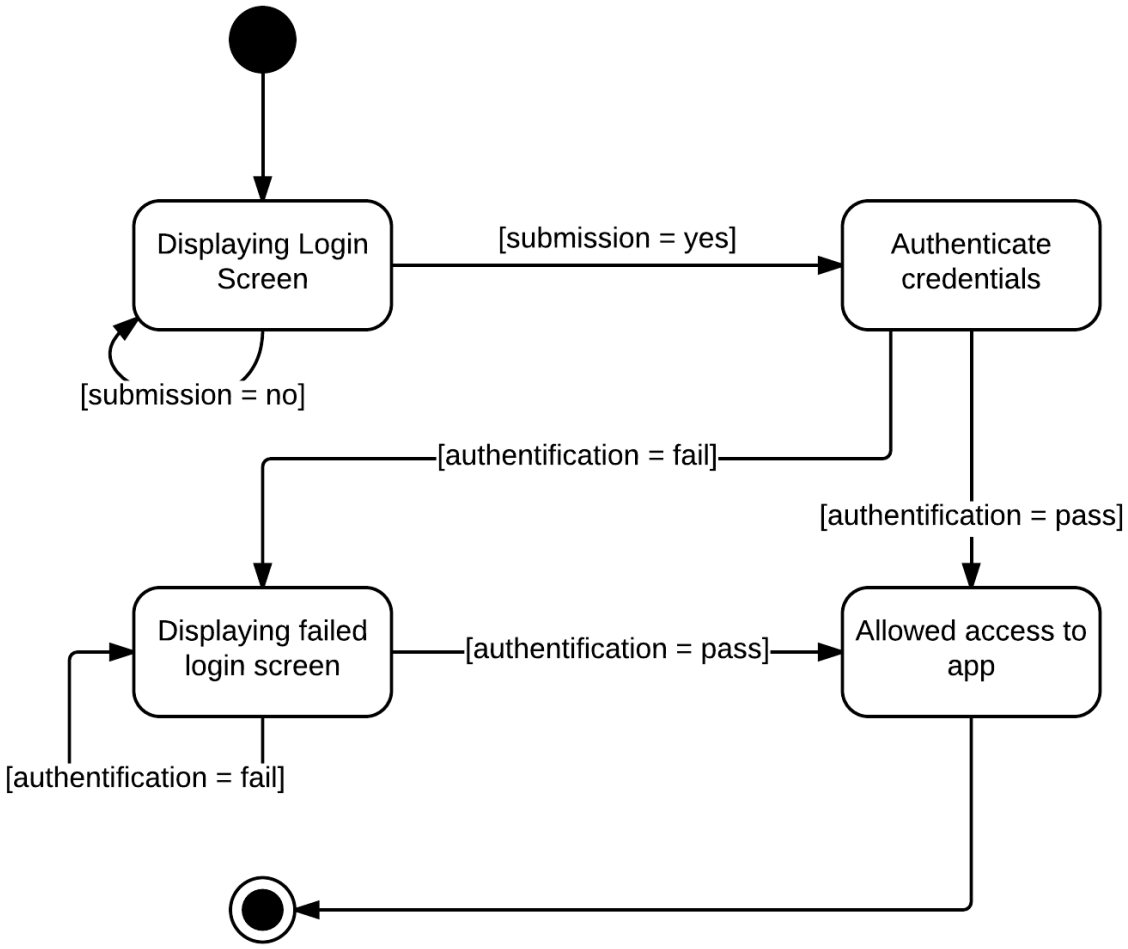

Figure 7: **State Diagram of SecurityController** When the user is attempting to login.

2.8 SelectController

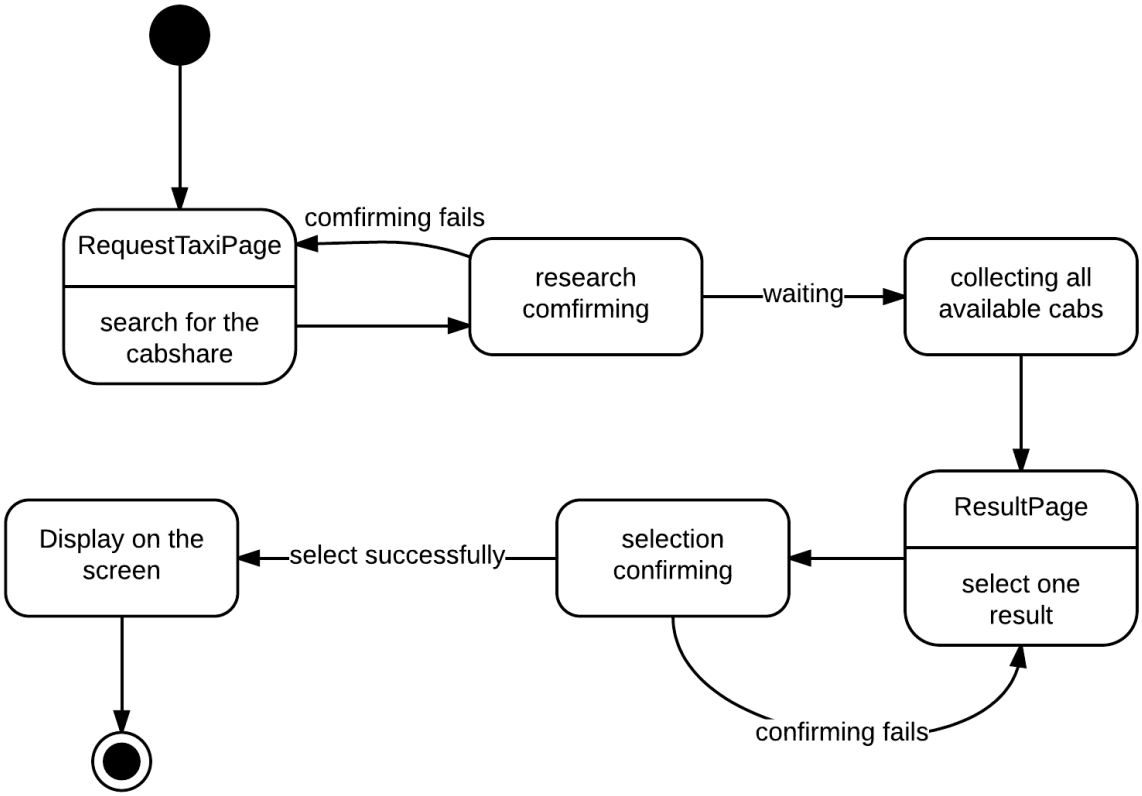

Figure 8: **State Diagram of SelectController** When user selects a cabpool to join.

2.9 ServerController

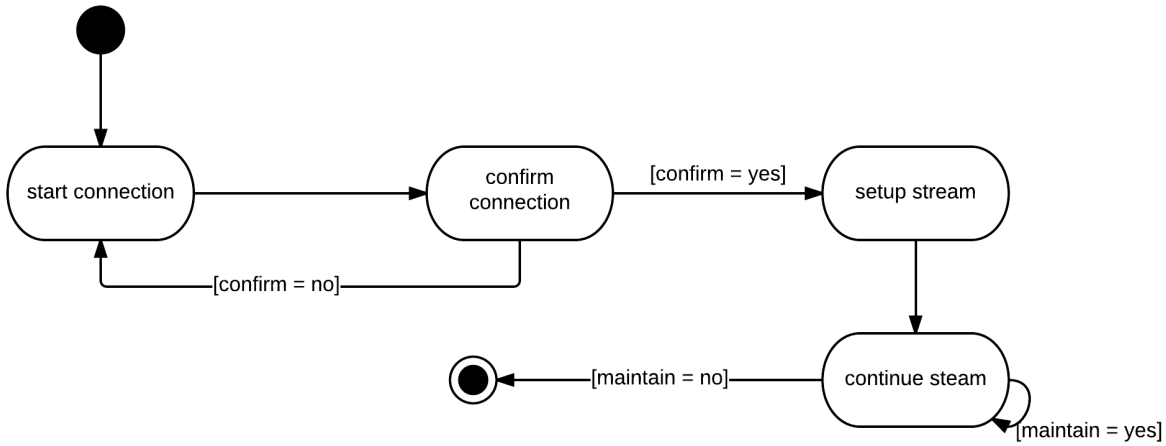

Figure 9: **State Diagram of ServerController** When the user tries to connect to the server initially.

2.10 TalkController

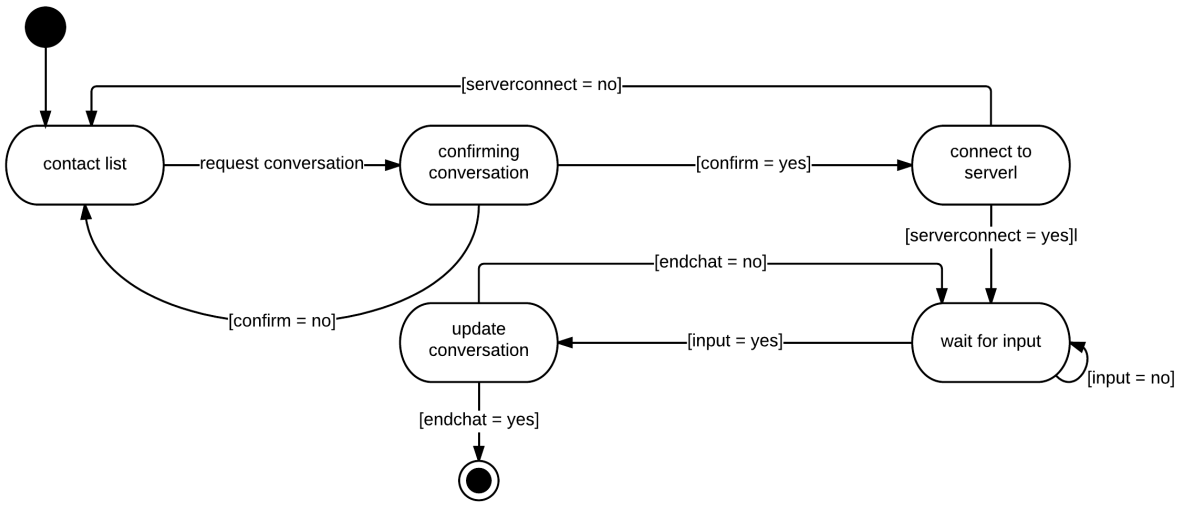

Figure 10: **State Diagram of TalkController** When the user wants to talk to the other people within the cabpool they are joining or when someone in the cabpool is communicating with someone they're going to pick up.

2.11 UserProfileController

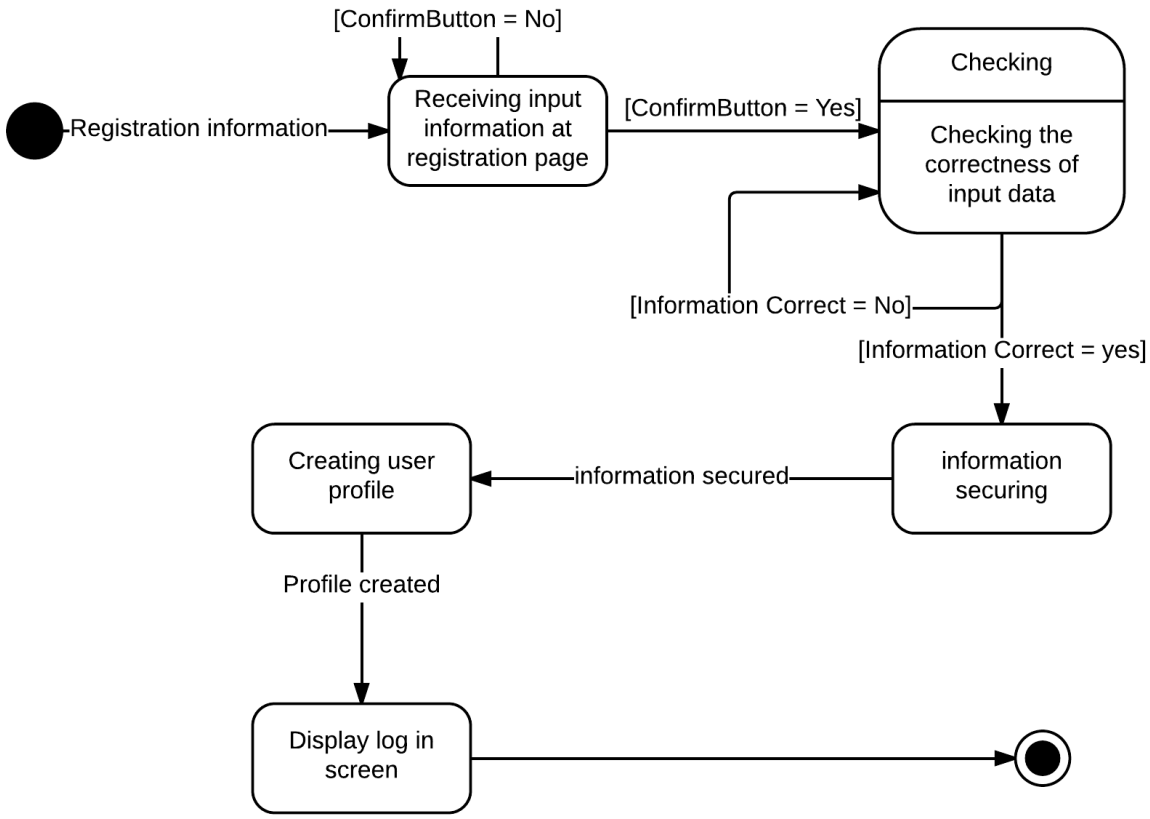

Figure 11: **State Diagram of UserProfileController** When the user first creates a profile.

2.12 UserRatingController

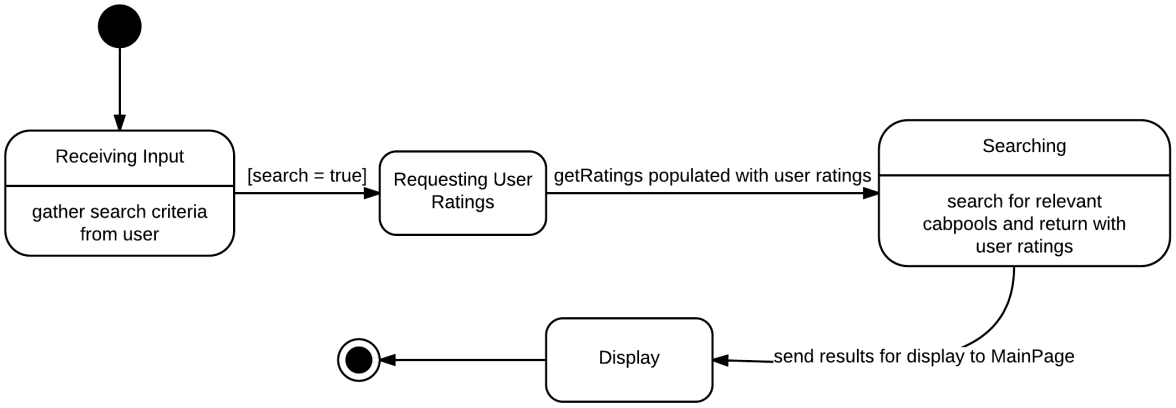

Figure 12: **State Diagram of UserRatingController** When a user wants to see the ratings of the user they're going to share the cabpool with.

3 Sequence Diagrams

3.1 CancelSequence

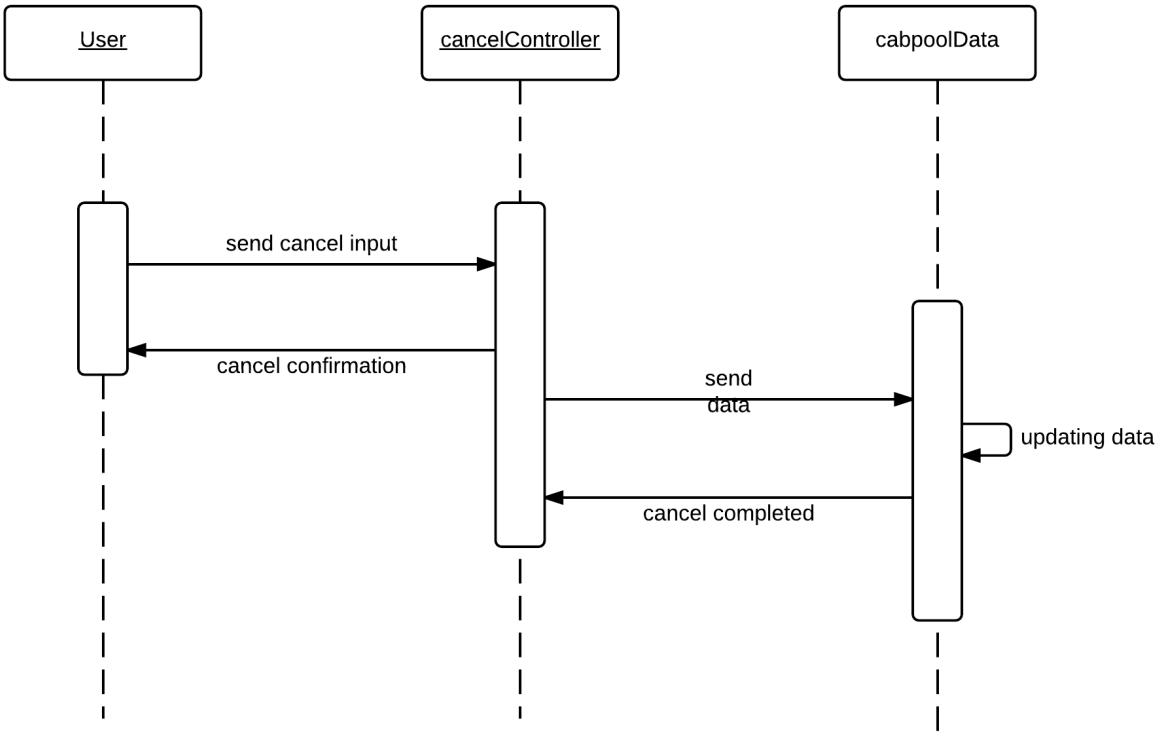

Figure 13: **Sequence Diagram of CancelSequence** How a user cancels a request for pickup.

3.2 LoginSequence

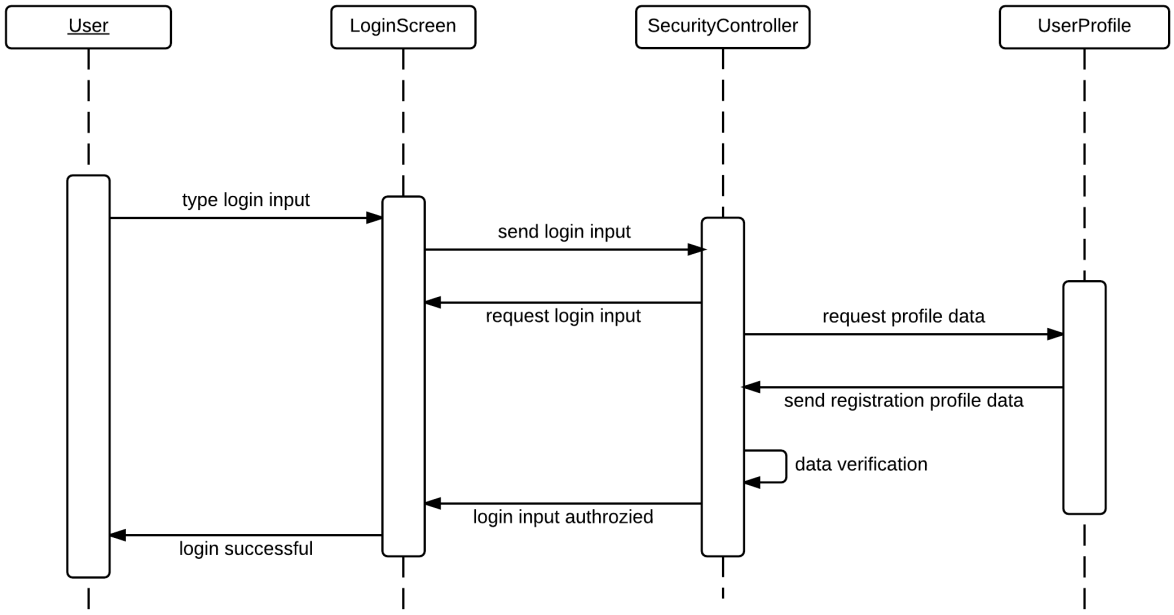

Figure 14: **Sequence Diagram of Login** The progression of the user logging in.

3.3 ModifyProfileSequence

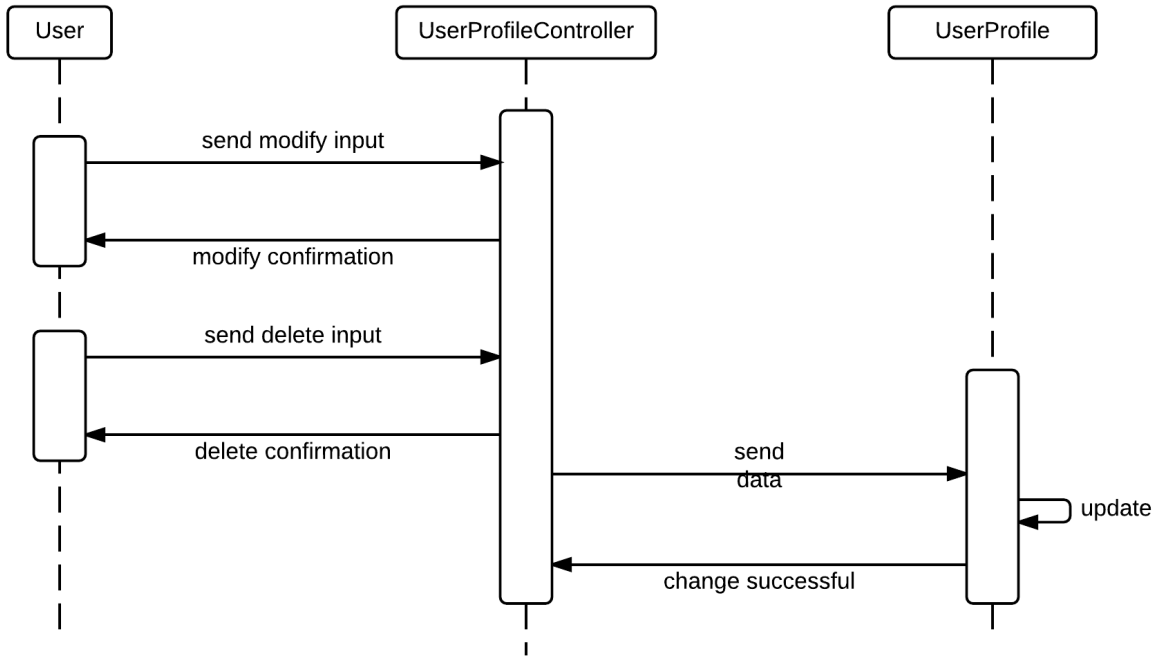

Figure 15: **Sequence Diagram of ModifyProfile** When the user modifies their profile details.

3.4 OfferSequence

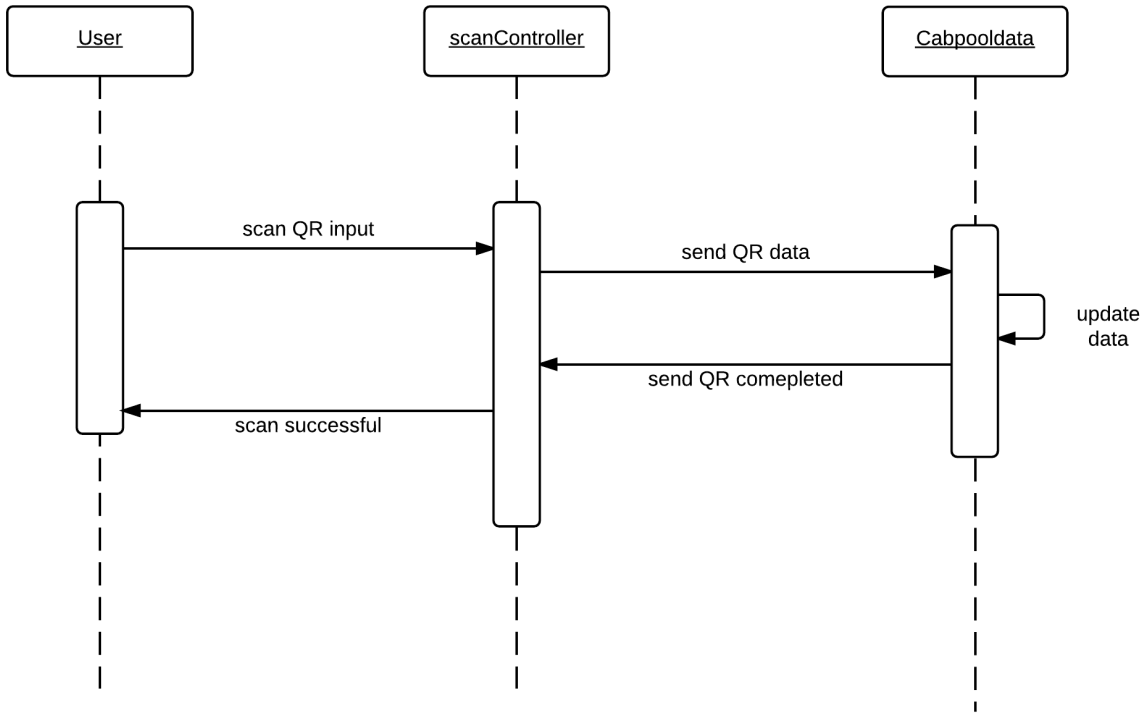

Figure 16: **Sequence Diagram of OfferSequence** When someone accepts an offer by scanning the QR code.

3.5 PaymentSequence

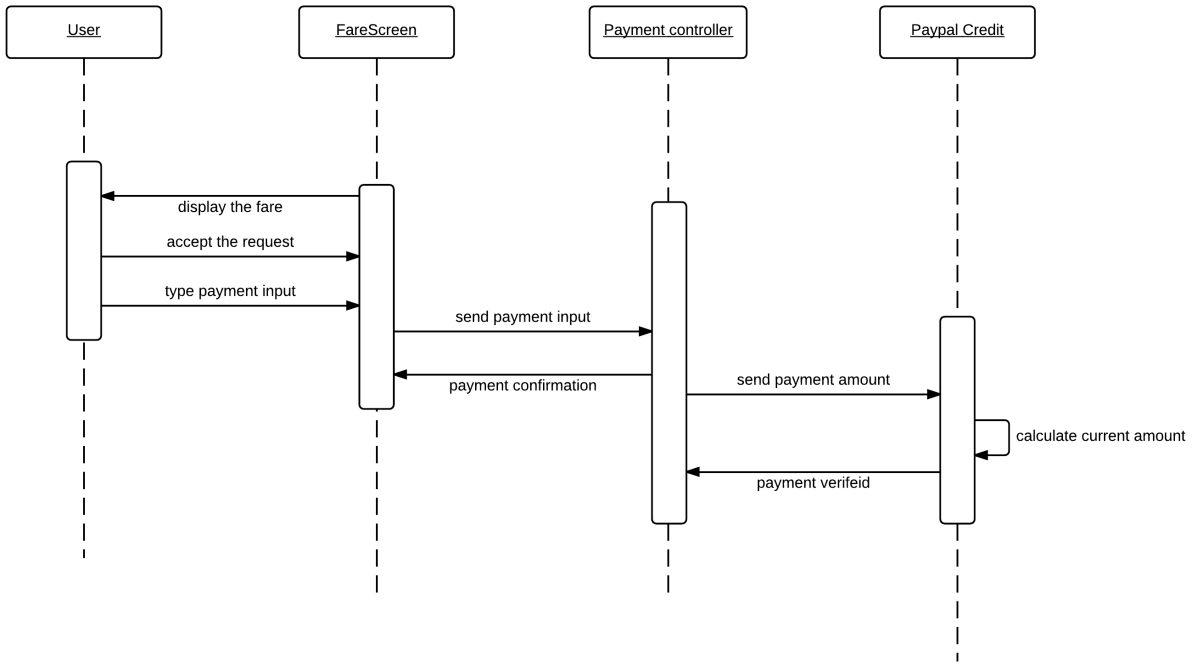

Figure 17: **Sequence Diagram of PaymentSequence** When someone has finished their cab ride and it's time to pay, using PayPal.

3.6 RateSequence

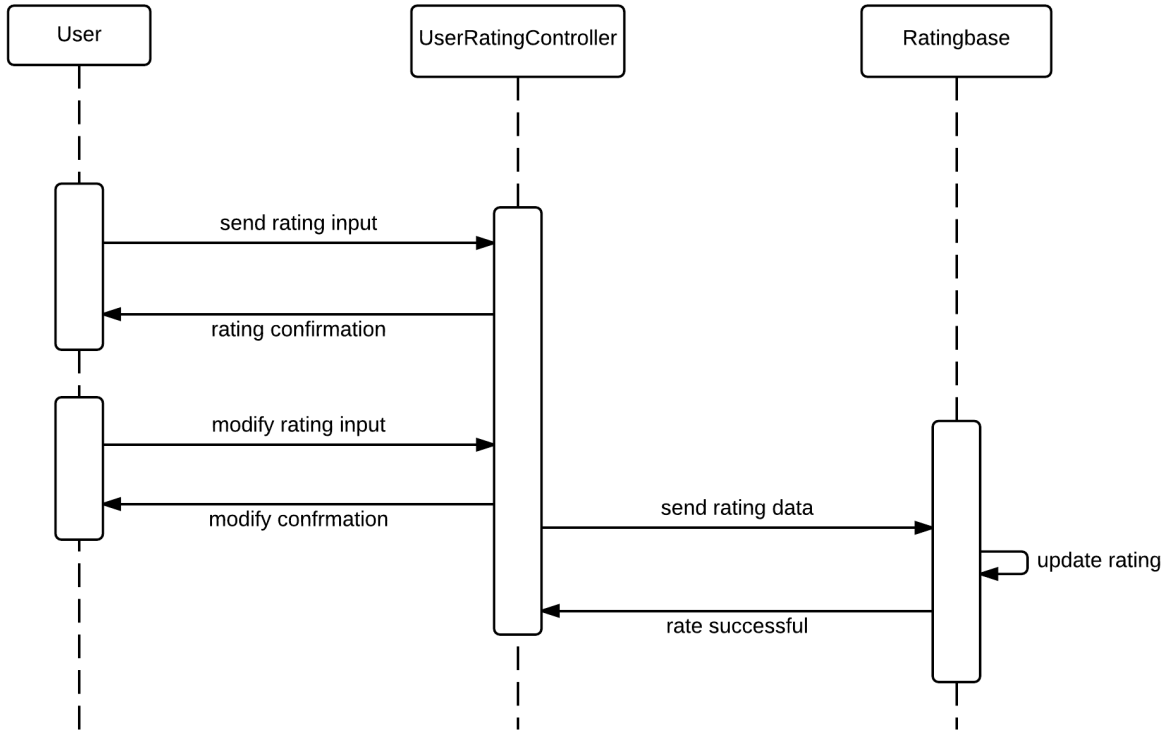

Figure 18: **Sequence Diagram of RateSequence** When someone rates a fellow cab rider.

3.7 ReceiveSequence

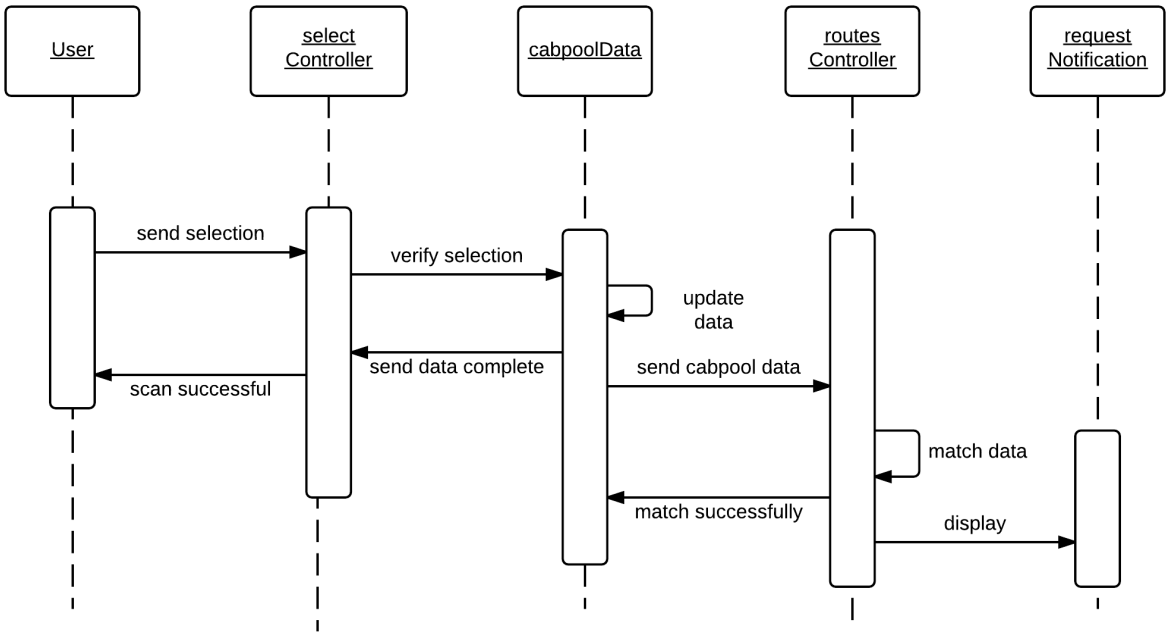

Figure 19: **Sequence Diagram of ReceiveSequence** When a user chooses a cabpool to join.

3.8 RegistrationSequence

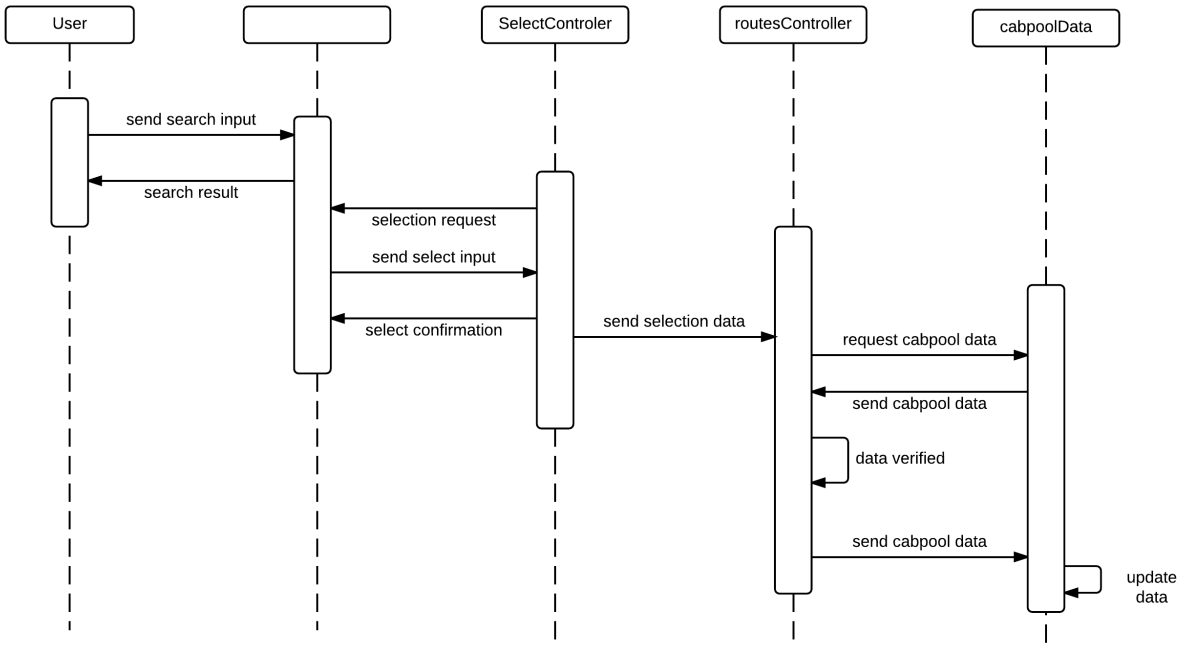

Figure 20: **Sequence Diagram of RegestrationSequence** When the user is first creating a profile.

3.9 RequestSequence

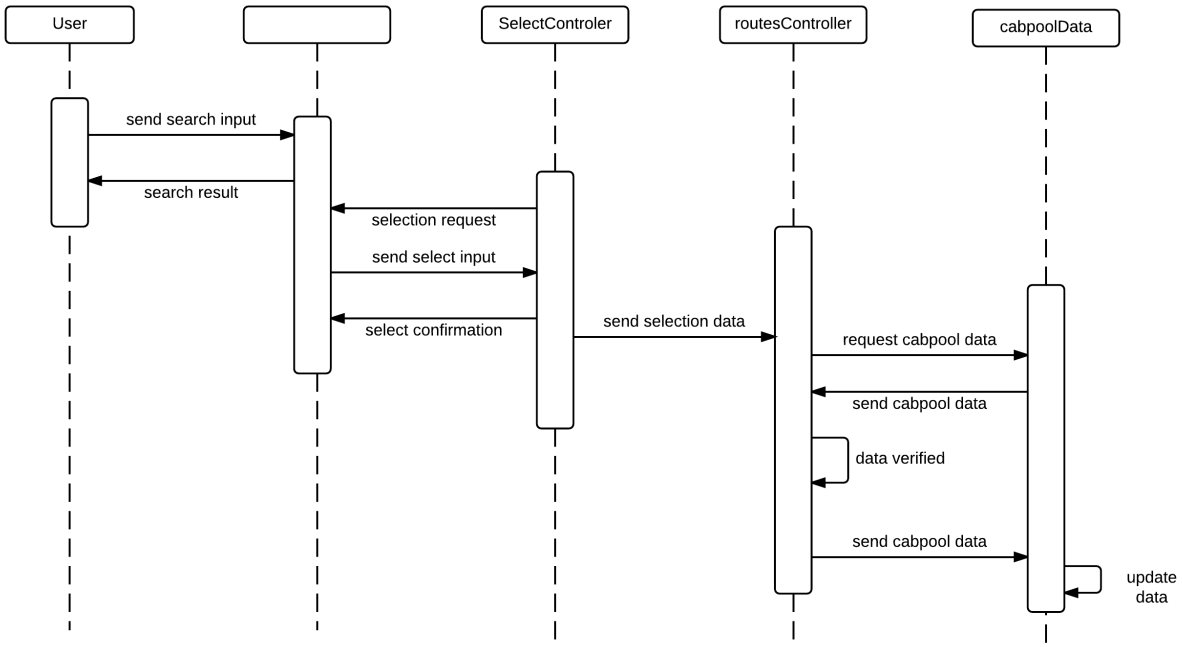

Figure 21: **Sequence Diagram of RequestSequence** When a user requests a cabpool.

4 Detailed Class Diagram

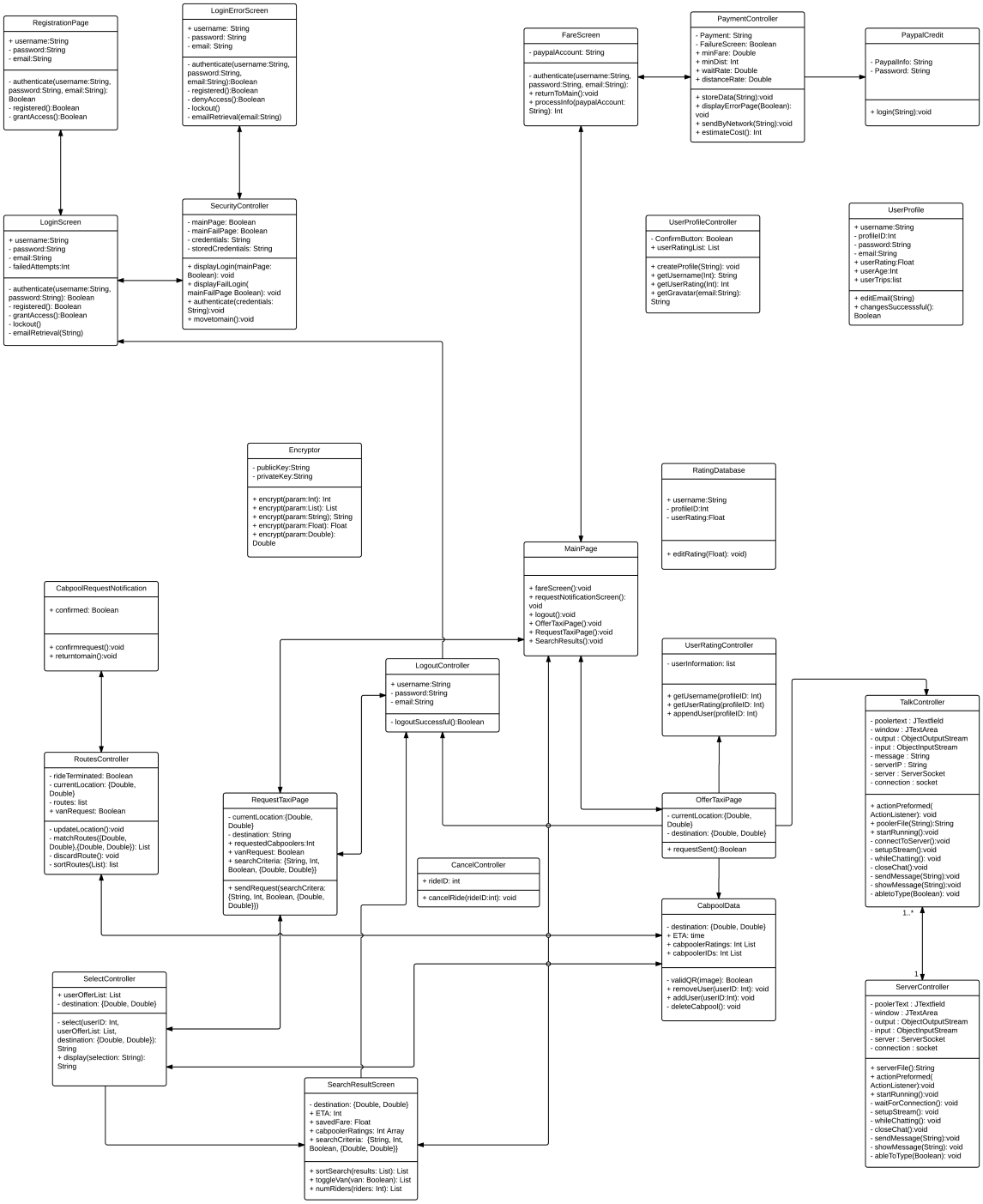

Figure 22: **Detailed Class Diagram** Classes and their relationships between each other are outlined in this document.

5 Cake

Figure 23: **Cake** Red velvet cake for your time.

A Division of Labour

Include a Division of Labour sheet which indicates the contributions of each team member. This sheet must be signed by all team members.

- Kemal Ahmed:
  - Formatted this LaTeX document
  - Reviewed all controller state diagrams before putting the picture in the LaTeX document
  - LogoutController State diagram
  - Learned and taught encryption protocols to group
  - Fed the group at meetings
  - Class Diagram
- Mitchell Spector:
  - Chat functionality
  - Class Diagram
- Sahajmeet Bhutta:
  - Introduction
  - Helped all other members with corrections in diagrams formatting and correctness of information
  - Class Diagram
- Xue Lin
  - Sequence Diagrams
- Yuchen Xiao
  - Encryptor
  - Cancel Sequence Diagram
  - UserProfileController