IskUber
Analysis Model

Submitted to:

Asst. Prof. Ma. Rowena C. Solamo
Faculty Member
Department of Computer Science
College of Engineering
University of the Philippines, Diliman

Submitted by:
Bilaw, Nicole
del Rosario, Luis Gabriel
Tamayo, Juan Gabriel

In partial fulfillment of Academic Requirements
for the course
CS 191 Software Engineering I
of the
1st Semester, AY 2017-2018

Revision Control

History Revision:

Revision Date	***Person Responsible***	***Version Number***	***Modification***
10/26/17	Juan Gabriel Tamayo	1.0	Initial document.
10/26/17	Nicole Bilaw	1.1	Added system description; Added description for entities.
10/27/17	Juan Gabriel Tamayo	1.2	Added boundary classes.
10/27/17	Nicole Bilaw	1.3	Added controller descriptions and started on class diagram.
10/29/17	Juan Gabriel Tamayo	1.31	Added missing classes, worked on class diagram.
10/29/17	Luis Gabriel del Rosario	1.32	Worked on class diagram.
10/29/17	Juan Gabriel Tamayo	1.4	Finalized class diagram; added to document.

Purpose:

This document serves as the official Analysis Model Document for IskUber. It will provide an overview of the system, it's modules and their dependencies via the Analysis Model.

Audience:

This document is targeted towards software engineers who wish to learn about or extend the functionalities of the system.

System Name: IskUber

Description: Uber is a transport mobile application that allows UP Diliman students to book rides inside the UP Diliman campus. It aids the students for easy transportation inside the campus. This web application provides a service that is an alternative to the Ikot and Toki jeeps which are the staple transportation of UPD students.

Class Diagram:

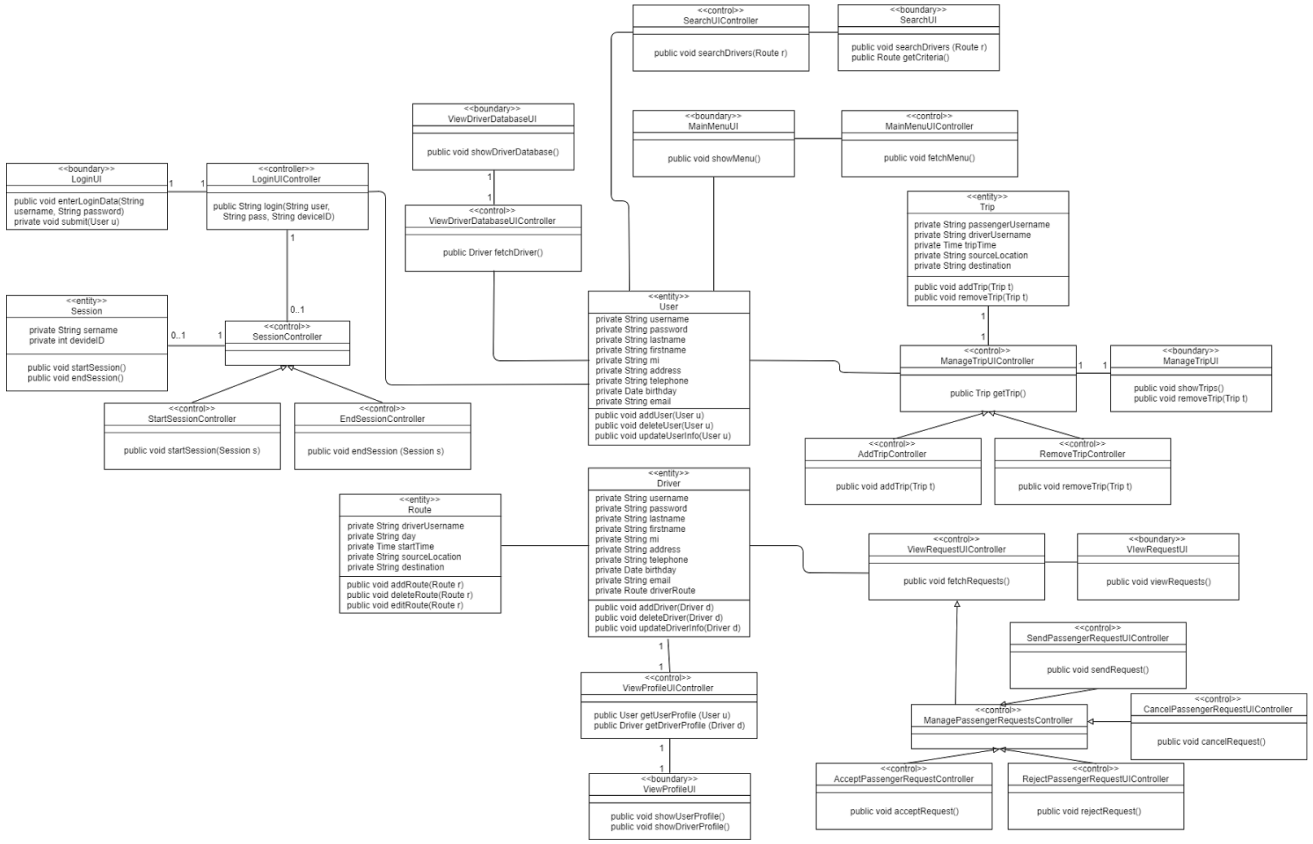

Figure 1: Class Diagram

Boundary Classes:

Class Name	Description
LoginUI	Contains the elements and resources found and used for the Login User Interface of the application. This interface is shown upon initially running the application, and upon logging out of the app. Here, the user may input their login credentials, or sign up for a new account.
MainMenuUI	Contains the elements and resources found and used for the Main Menu Interface of the application. This interface is shown upon logging in, and during subsequent runs of the application while logged in. Here, the user is shown a summarized status of their account (current trips, outstanding requests). Buttons that lead to the SearchUI, ViewRequestUI and ManageTripUI can also be found here.
ViewRequestUI	Contains the elements and resources found and used for the View Request Interface of the application. This interface is shown upon clicking the appropriate button from the MainMenuUI. The user's outstanding requests are shown here. If the user is a driver, then their passenger requests are also displayed.
ViewProfileUI	Contains the elements and resources found and used for the View Profile Interface of the application. This interface is shown upon inspecting a Driver from the SearchUI, ViewDriverDatabaseUI, or the ManageTripUI. The specified driver's credentials and contact details are displayed here. If the user is not in the specified driver's trip list, they may also send a passenger request from here.
SearchUI	Contains the elements and resources found and used for the Driver Search Interface of the application. This interface is shown upon clicking the appropriate button from the MainMenuUI. The user may input search parameters to look for a driver that can accomodate their trip. A button that will display the ViewDriverDatabaseUI is can also be seen here.
ViewDriverDatabaseUI	Contains the elements and resources found and used for the View Driver Database Interface of the application. This interface is shown upon clicking the appropriate button from the SearchUI. This interface displays a list of all drivers currently registered within the system, in alphabetical order.
ManageTripUI	Contains the elements and resources found and used for the Manage Trips Interface of the application. This interface is shown upon clicking the appropriate button from the MainMenuUI. The user's current trips are shown here. If the user is a driver, then their current passengers are also displayed.

Control Classes:

Class Name	Description
LoginUIController	This control class can be performed by the user to log-in to the system.
SessionController	This control class handles the user login sessions.
StartSessionController	This control class activates during the login process.
EndSessionController	This control class activates during the logout process.
MainMenuUIController	This control class can be performed by the user to navigate the main menu options of the system.
ViewRequestUIController	This control class can be performed by the user to view the current requests.
ViewProfileUIController	This control class can be performed by the user to view his/her profile.
SearchUIController	This control class can be performed by the user to search for drivers.
ViewDriverDatabaseUIController	This control class can be performed by the user to view the list of all registered drivers in the database.
ManageTripUIController	This control class can be performed by the user to manage his/her current trips.
AddTripController	This control class is automatically activated upon accepting a passenger request.
RemoveTripController	This control class can be performed by the user to remove any of their current trips.
ManagePassengerRequestsController	This control class handles the driver and passenger requests.
AcceptPassengerRequestController	This control class can be performed by the driver to accept a passenger request.
RejectPassengerRequestController	This control class can be performed by the driver to reject a passenger request.
SendPassengerRequestController	This control class can be performed by the user to send a passenger request to drivers.
CancelPassengerRequestController	This control class can be performed by the user to cancel a passenger request.

Entity Classes:

Class Name	Description
User	This entity contains relevant information about the user of the mobile application. These relevant information are as follows: username, full name, address, telephone, email address, birthday.
Driver	This entity also contains relevant information about the driver. These relevant information are as follows: username, full name, address, telephone, email address, birthday, and the route the driver is taking.
Route	This entity contains details of a route a driver is taking. It contains relevant information as follows: the username of the driver, the day of the week, the starting time, the source location, and the destination.
Trip	This entity contains details of the user's/driver's trip. Variables such as time, user involved, driver involved, and path information are stored here.
Session	This entity contains details of the user's login session. The username and deviceID used to log in will be stored here.