

# **IskUber**

## **Use Case Specification**

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  
1<sup>st</sup> Semester, AY 2017-2018

**Unique Reference:**

The documents are stored in the IskUber GitHub Repository Link.

<https://github.com/CrumbleThorn/IskUber>

**Document Purpose:**

This document serves as the official Use Case Specification document for IskUber. This document expounds on the internal details of the use case.

**Target Audience:**

This document is targeted towards software engineers who wish to learn about the specific flow of events for each use case of the system.

**Revision Control***History Revision:*

<b>Revision Date</b>	<b>Person Responsible</b>	<b>Version Number</b>	<b>Modification</b>
10/08/17	Luis Gabriel Q. del Rosario	1.0	Initial Document.
10/11/17	Juan Gabriel C. Tamayo	1.1	Filled out empty fields and tables, and added the Activity Diagram; Version number should match the one found in the footer.

**Use-Case Name:** 5.1 Update Search Criteria

**Description:** The user may opt to update their search criteria if they made a mistake, or if they want to change their . An updated list of available drivers that may accommodate their request will be displayed.

**Preconditions:** The user must have already initiated a search (Use Case 5.0) in order to arrive at this use case.

**Flow of Events:**

<b>Scenario Name</b>	<b>Description</b>
Scenario 1 (Basic Flow) User updates the search parameters, initiates the search, and finds a matching driver.	1. The search menu with the previous search parameters is displayed. 2. The user updates the search fields. 3. After pressing search, the driver database is checked for matching search criteria. 4. The results are displayed for the user to view. 5. The user picks a driver and views his/her credentials and contact details.
Scenario 2 User updates the search parameters, initiates the search, and views the results.	1. The search menu with the previous search parameters is displayed. 2. The user updates the search fields. 3. After pressing search, the driver database is checked for matching search criteria. 4. The results are displayed for the user to view.
Scenario 3 User fills in the search box, initiates the search, and wishes to update their search criteria.	1. The search menu with the previous search parameters is displayed. 2. The user updates the search fields. 3. After pressing search, the driver database is checked for matching search criteria. 4. The results are displayed for the user to view. 5. The user clicks a button to return to the search menu and update their search parameters.
Scenario 4 The search yielded no matching results.	1. The search menu with the previous search parameters is displayed. 2. The user updates the search fields. 3. After pressing search, the driver database is checked for matching search criteria. 4. The user is informed that the search yielded no results, and is returned to the search menu.

*Activity Diagram of the Flow of Events:*

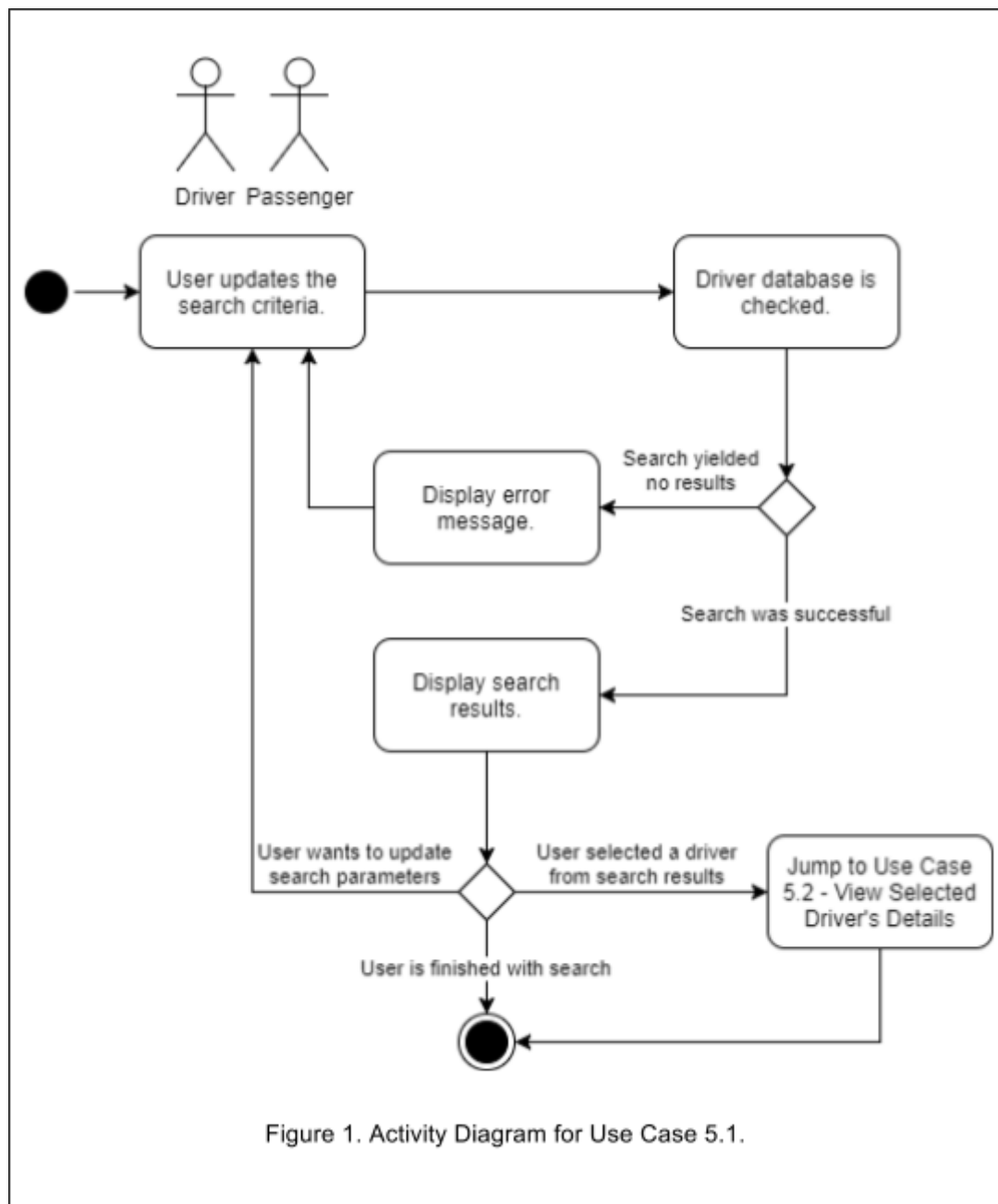


Figure 1. Activity Diagram for Use Case 5.1.

*Postcondition:* NONE

*Relationships:* This Use Case is related to Use Case 5.0.

*Special Requirements:*  
NONE