UP Bike Share – Android App

Use Case Specification

Submitted to:

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

Submitted by:

Barozzo, Steven
Mamac, Mark Anton
San Gabriel, Jaypee Renz

In partial fulfillment of Academic Requirements
for the course
CS 191 Software Engineering I
of the
1st Semester, AY 2015-2016

Unique Reference:

The documents are stored in:
https://github.com/devsofup/UPBike-Share-Android/tree/master/01-Project-Documents

Document Purpose:

This document serves to detail the structure of the Bike Share project's main functionalities or use cases and how they work, without too much detail on the software implementation. This will serve as the agreement between the client and the developers as to how each of the functionalities will work. This will also guide the developers as to what are the most important aspects of the application.

Target Audience:

This document is mainly for the viewing of the client and the development team. It will also be viewed by the guiding faculty.

Revision Control

History Revision:

Revision Date	***Person Responsible***	***Version Number***	***Modification***
09/17/15	Jaypee San Gabriel	0.5	Placed description, preconditions, postconditions, relationships and special requirements for each use case.
09/18/15	Jaypee San Gabriel Steven Barrozo	1.0	Complete rough draft. Created activity flow and activity diagrams for each use case.

Use-Case Name: Use-Case 1.0 View Bike Share Information

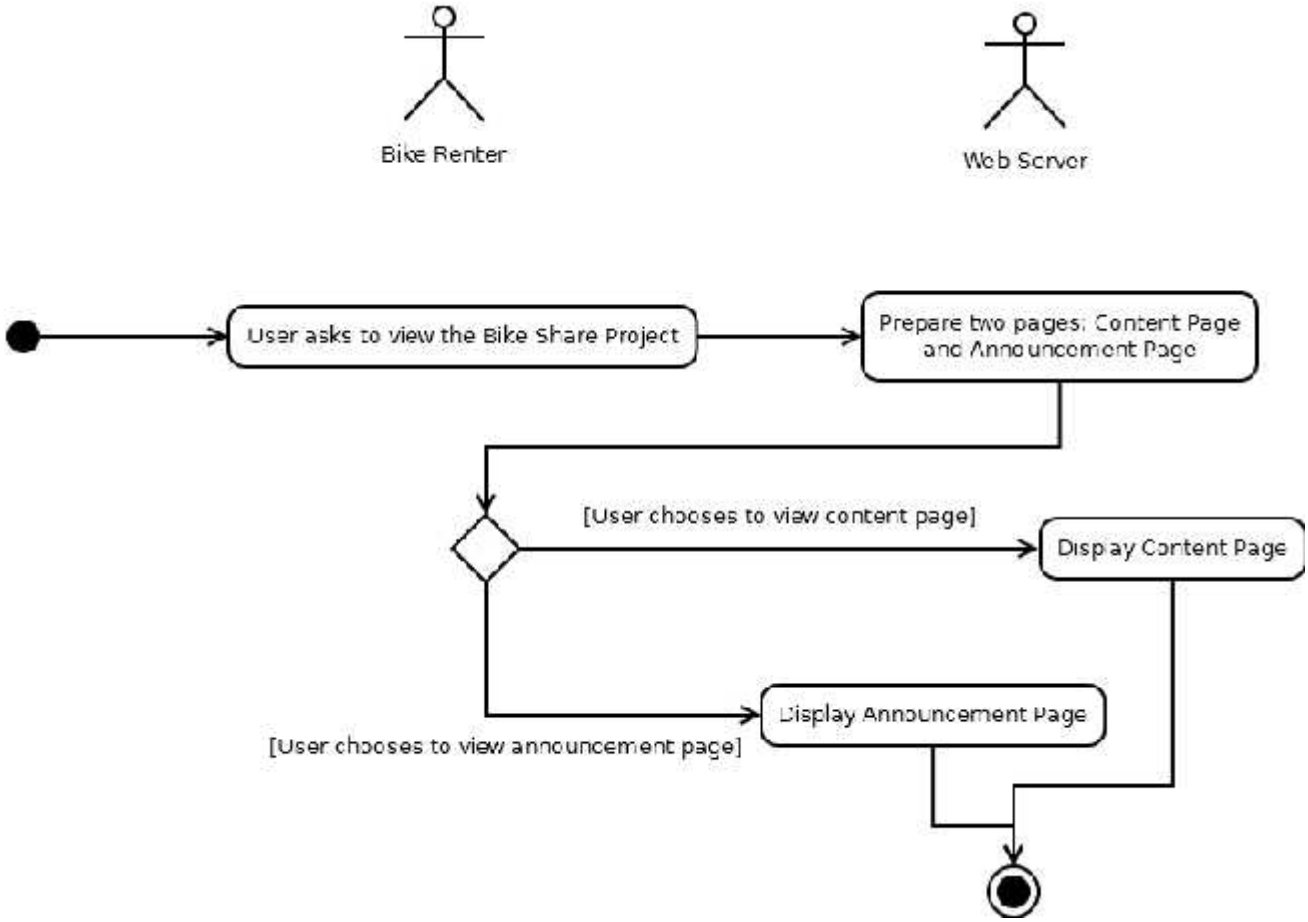
Description: Users of the Bike Share module, whether they have an account or not, can view details and news regarding the Bike Share Project. This includes a Content Page featuring all information about the Bike Share Project in general. There is also an Announcement Page, featuring recent happenings for the Bike Share Project including short term, long term and recurring news.

Preconditions: There is a working connection with the Web Server.

Flow of Events:

Scenario Name	***Description***
Scenario 1(Basic Flow: Content Page) Possible and actual Bike Renters view the Content Page	1. Through the Bike Share module, the user asks the Web Server about the Bike Share Project. 2. The Web Server provides two pages: the Content Page and the Announcement Page. 3. The user views the Content Page. 4. The system displays the contents of the Content Page regarding the Bike Share Project.
Scenario 2 (Basic Flow: Announcement Page) Possible and actual Bike Renters view the Announcement Page.	1. Through the Bike Share module, the user asks the Web Server about the Bike Share Project. 2. The Web Server provides two pages: the Content Page and the Announcement Page. 3. The user views the Announcement Page. 4. The system displays the most recent announcements from the Web Server.
Scenario 3 (Connection Error)	1. If at any time in the basic flow, the connection with the Web Server is lost, the system will display a Connection Error.

Activity Diagram of the Flow of Events:



Postcondition: NONE

Relationships: NONE

Special Requirements:
NONE

Use-Case Name: Use-Case 2.0 Register Account

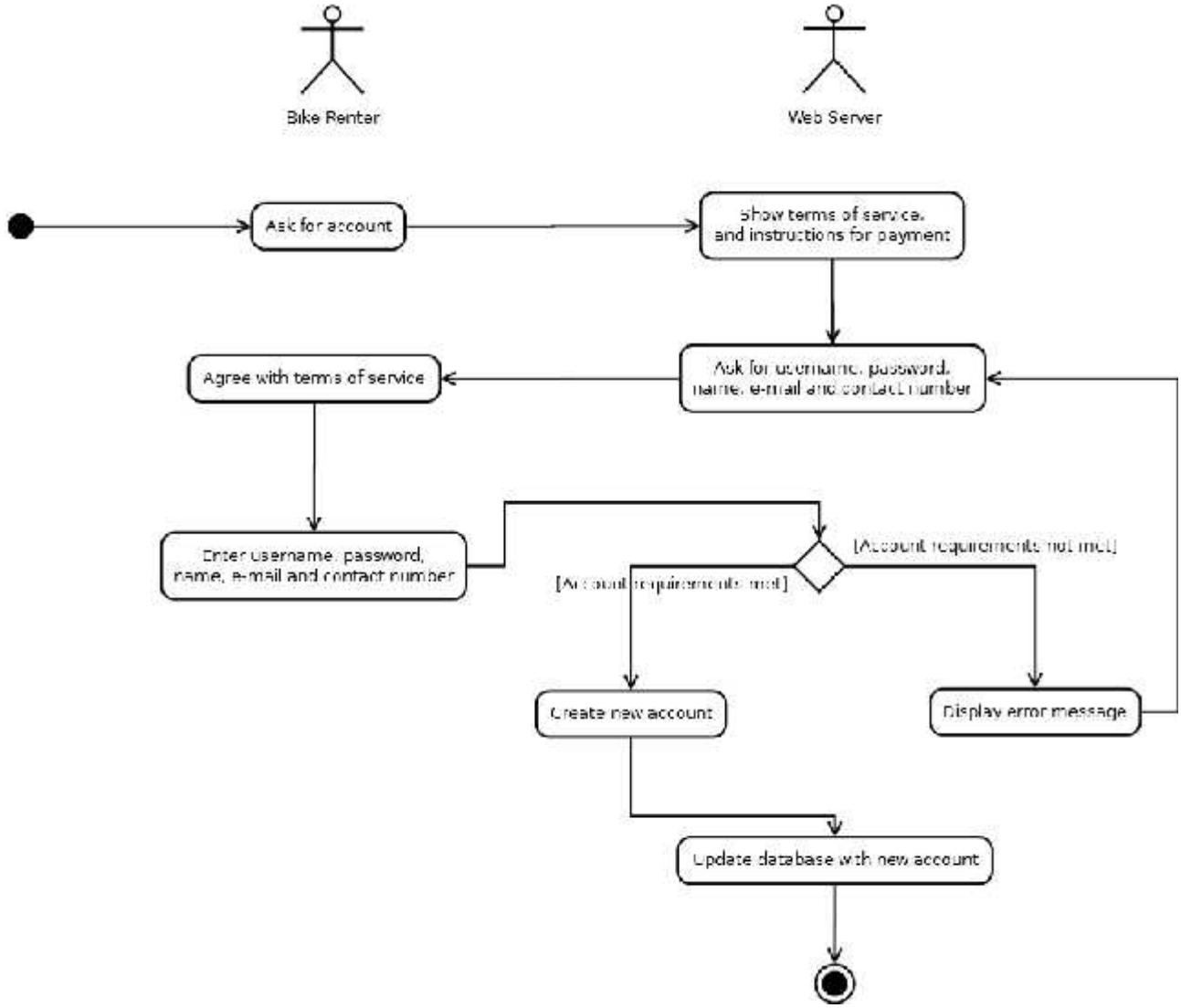
Description: Anyone can register for an account and become a Bike Renter. They can also register for bike lessons. They will be given instructions on how to pay for the service of Bike Share.

Preconditions: NONE

Flow of Events:

Scenario Name	***Description***
Scenario 1 (Basic Flow) A Bike Renter registers for an account.	1. The Bike Renter asks the system for an account. 2. The system presents the terms of service, along with instructions for payment. 3. The Bike Renter agrees with said terms and regulations. 4. The system prompts the Bike Renter to enter a username and password for their account, along with their name, e-mail, and contact number. 5. The Bike Renter enters a username and password for their account, along with their name, e-mail and contact number. 6. If all the requirements of the username and password are met, the Bike Renter's new account is made. 7. The Web Server updates the user database with the new account.
Scenario 2 (Registration Error) A Bike Renter registers for an account, but requirements for the username and password are not met.	1. The Bike Renter asks the system for an account. 2. The system presents the terms of service, along with instructions for payment. 3. The Bike Renter agrees with said terms and regulations. 4. The system prompts the Bike Renter to enter a username and password for their account. 5. The Bike Renter enters a username and password for their account, along with their name, e-mail and contact number. 6. If all the requirements of the username and password are not met, the system presents an error message, and the flow trails back to step 4 of the scenario basic flow.
Scenario 3 (Connection Error)	1. If at any time in the basic flow, there is no connection with the Web Server, the system will display a Connection Error.

Activity Diagram of the Flow of Events:



Postcondition: NONE

Relationships: For a Bike Renter to be able to use the Log-In Account use case successfully, he/she must have registered an account through this use case. The Web Server which holds the database will then add the newly-registered Bike Renter to the list of users with accounts.

Special Requirements:

NONE

Use-Case Name: Use-Case 3.0 Log-In Account

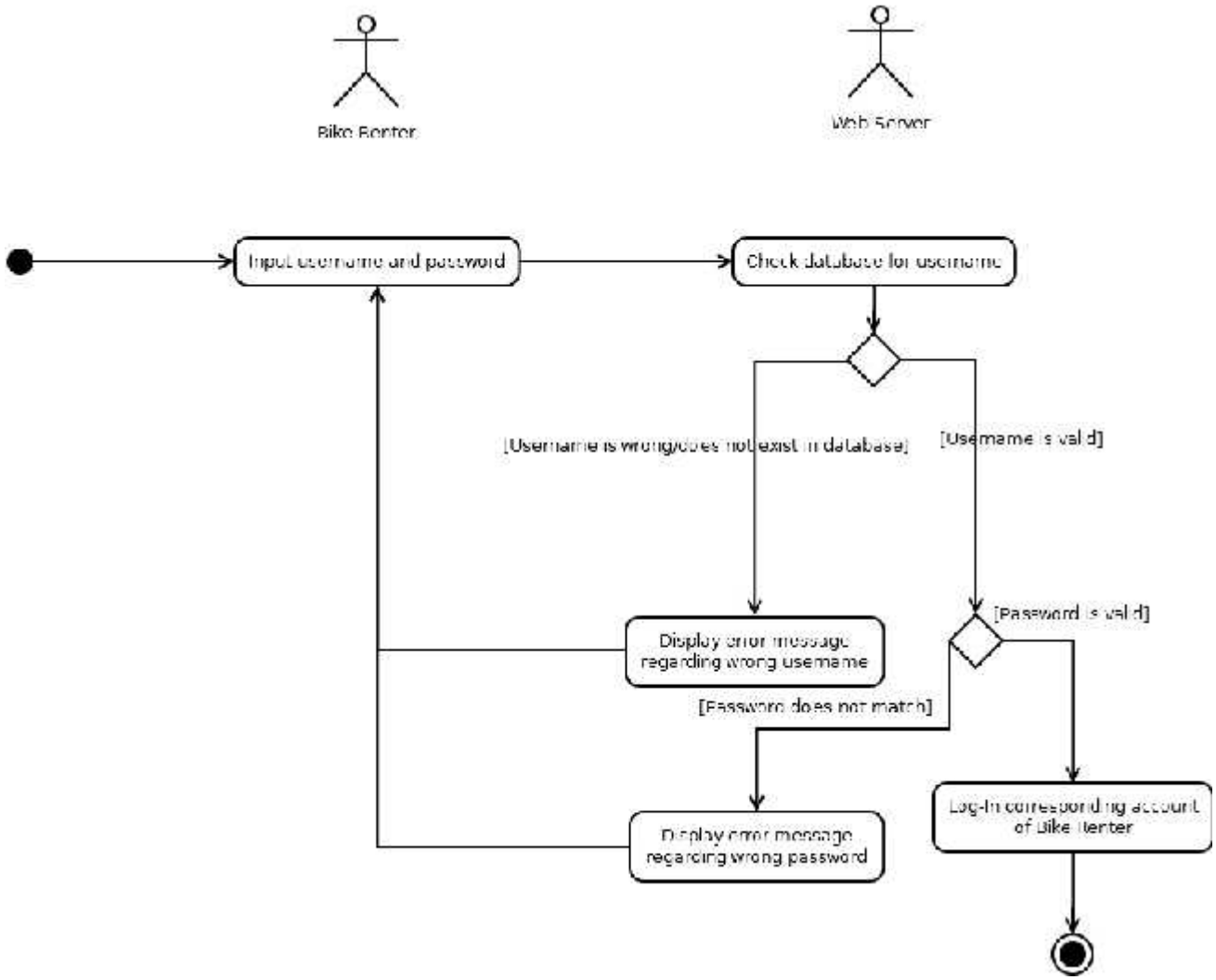
Description: A Bike Renter who has registered an account can then log in the system using their username and password. Most other functionalities of the Bike Share module would require users to be logged in.

Preconditions: NONE

Flow of Events:

Scenario Name	***Description***
Scenario 1 (Basic Flow) The Bike Renter logs in his/her account.	1. The Bike Renter inputs their username and password. 2. The Web Server checks the user database for a matching username and password. 3. If there is a match, the corresponding user is logged in.
Scenario 2 (Wrong Username) The Bike Renter attempts to log in his/her account, but inputs a wrong/non-existing username.	1. The Bike Renter inputs their username and password. 2. The Web Server checks the user database for a matching username and password. 3. If the username is not found in the database, the system displays an error stating that the username is wrong, and the flow trails back to step 1 of the scenario basic flow.
Scenario 3 (Wrong Password) The Bike Renter attempts to log in his/her account, but inputs an incorrect password for the username.	1. The Bike Renter inputs their username and password. 2. The Web Server checks the user database for a matching username and password. 3. If the username is in the database but the inputted password is wrong, the system displays an error stating that the password is wrong, and the flow trails back to step 1 of the scenario basic flow.
Scenario 4 (Connection Error)	1. If at any time in the basic flow, the connection with the Web Server is lost, the system will display a Connection Error.

Activity Diagram of the Flow of Events:



Postcondition: The Bike Renter's account status is 'Logged In.'

Relationships: The Web Server will communicate with the Bike Share module that the Bike Renter is using for this use case in order to validate account credentials and place the Bike Renter's account status as 'Logged In.' The View Account, Search Available Bikes, Reserve Bike, Lock Bike, and Report Bike Issue use cases can only be used once the Bike Renter is logged in.

Special Requirements:

NONE

Use-Case Name: Use-Case 4.0 View Account

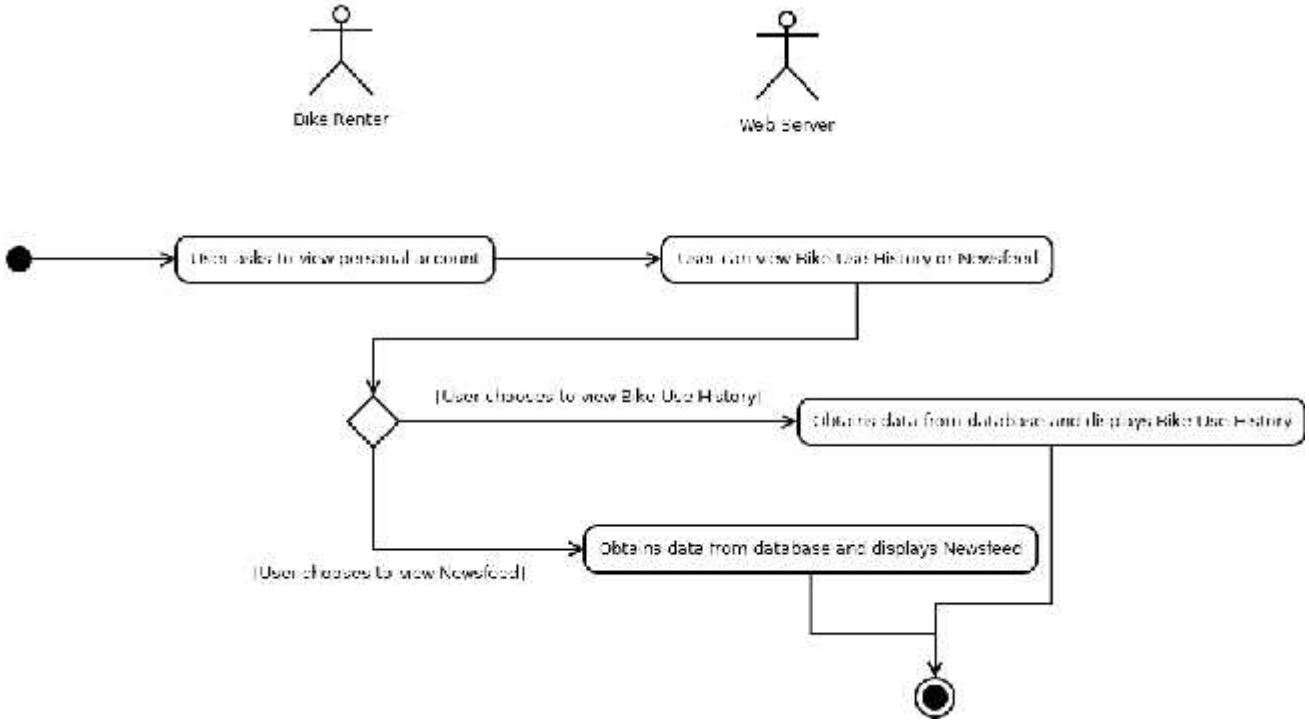
Description: While logged in, a user can view his/her current account details so that he/she can take note of the current status of his/her account. This includes a Bike Use History, which shows the previous bikes used, when they were used, and all relevant transactions. There is also a dynamic Newsfeed showing which bikes have been reported as having issues.

Preconditions: The Bike Renter is 'Logged In.'
There is a working connection with the Web Server.

Flow of Events:

Scenario Name	***Description***
Scenario 1(Basic Flow: Bike Use History) A logged-in Bike Renter views the Bike Use History of his/her account.	1. Through the Bike Share module, the user prompts the system to view his/her account. 2. The system provides two pages: the Bike Use History and the Newsfeed. 3. The user chooses to view the Bike Use History. 4. The system displays the Bike Use History for the user's account using data from the Web Server.
Scenario 2 (Basic Flow: Newsfeed) A logged-in Bike Renter views the Newsfeed.	1. Through the Bike Share module, the user prompts the system to view his/her account. 2. The system provides two pages: the Bike Use History and the Newsfeed. 3. The user chooses to view the Newsfeed. 4. The system displays the current Newsfeed using data from the Web Server.
Scenario 3 (Connection Error)	1. If at any time in the basic flow, the connection with the Web Server is lost, the system will display a Connection Error, and log out the current user.

Activity Diagram of the Flow of Events:



Postcondition: NONE

Relationships: NONE

Special Requirements:
NONE

Use-Case Name: Use-Case 5.0 Search Available Bikes

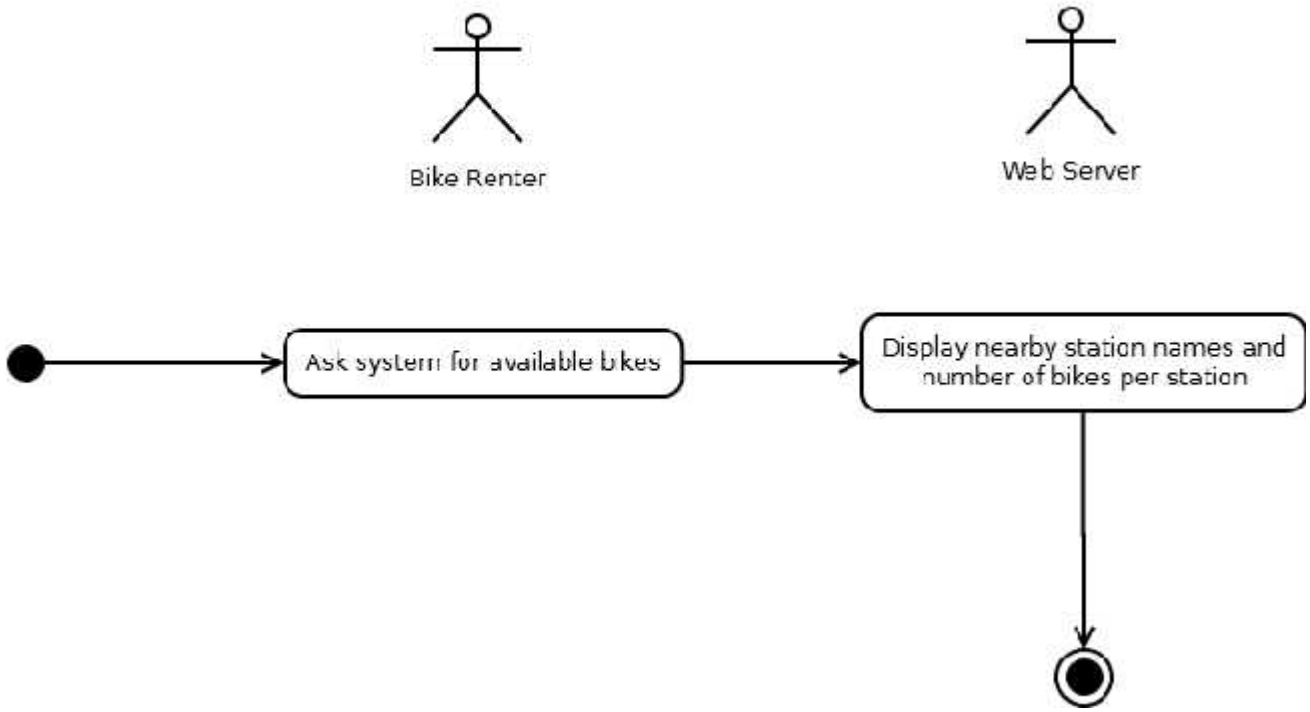
Description: A Bike Renter can request a list of available bikes for use. Using data from the Web Server, the system can produce a list of nearby bikes and their location. The number of available bikes per station will be shown.

Preconditions: The Bike Renter is 'Logged In.'
There is a working connection with the Web Server.

Flow of Events:

Scenario Name	***Description***
Scenario 1 (Basic Flow) The Bike Renter searches for available bikes.	1. The Bike Renter prompts the system to search for available bikes. 2. Using data from the Web Server, the system displays nearby station names and produces the number of available bikes per station.
Scenario 2 (Connection Error)	1. If at any time in the basic flow, the connection with the Web Server is lost, the system will display a Connection Error, and log out the current user.

Activity Diagram of the Flow of Events:



Postcondition: NONE

Relationships: NONE

Special Requirements:
Physical devices for tracking associated with the bikes function properly.

Use-Case Name: Use-Case 6.0 Reserve Bike

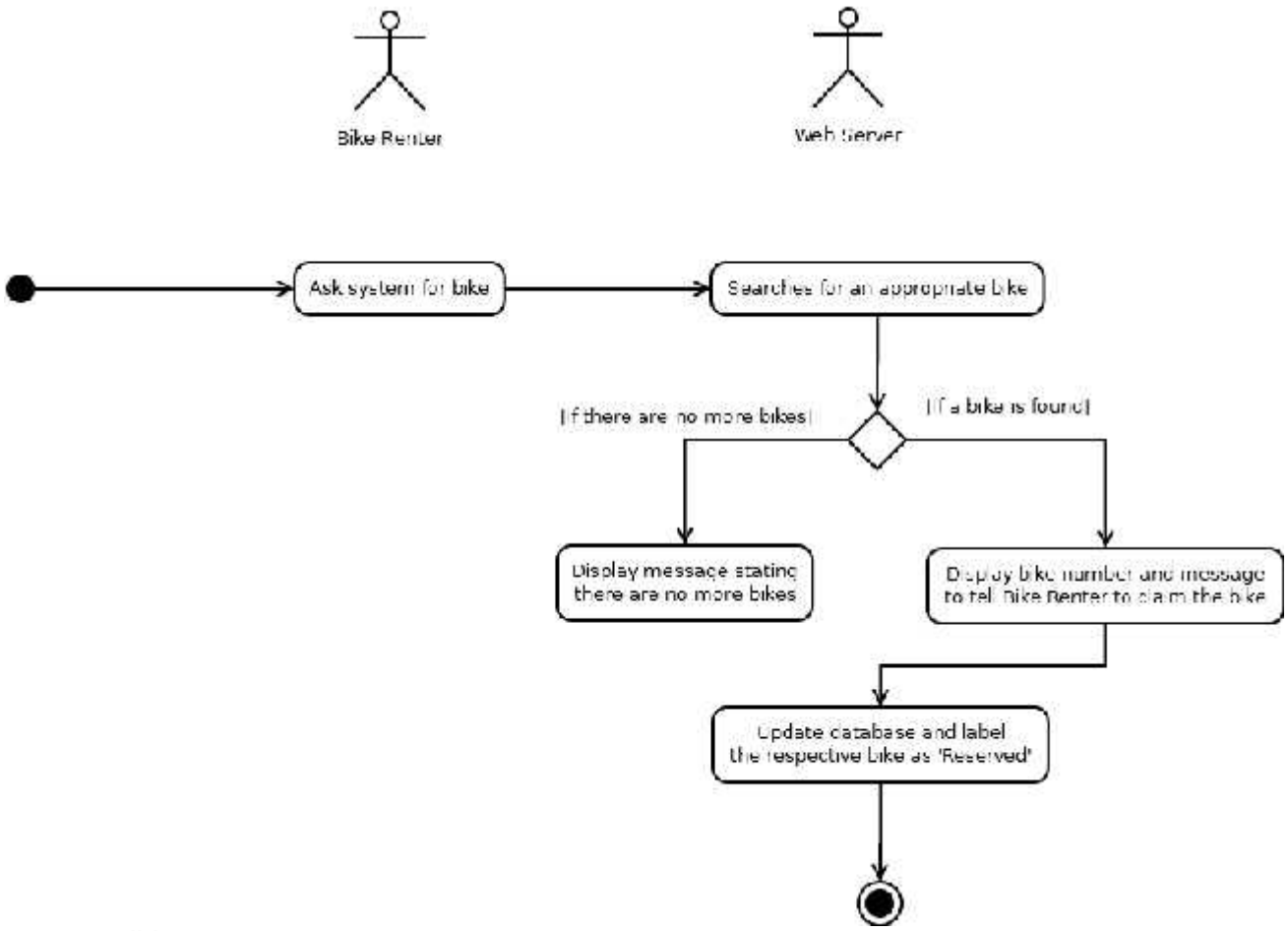
Description: If the Bike Renter has chosen a bike it can use, it can send data to the Web Server telling it that it is renting the said bike for use. The bike is to be claimed within 30 minutes. The designated bike number will be shown.

Preconditions: The Bike Renter is 'Logged In.'
There is a working connection with the Web Server.

Flow of Events:

Scenario Name	***Description***
Scenario 1 (Basic Flow) The Bike Renter reserves a bike.	1. The Bike Renter asks the system for a bike. 2. The Web Server chooses an appropriate bike based on location. 3. The system displays the bike number and a message telling the Bike Renter to claim the said bike within 30 minutes. 4. The Web Server updates the database, placing the respective bike as 'Reserved.'
Scenario 2 (No Bikes Available) The Bike Renter reserves a bike, but there are no more nearby bikes available.	1. The Bike Renter asks the system for a bike. 2. The Web Server chooses an appropriate bike based on location. 3. If there are no longer any bikes available, the system displays a message stating that there are no more bikes.
Scenario 3 (Connection Error)	1. If at any time in the basic flow, the connection with the Web Server is lost, the system will display a Connection Error, and log out the current user.

Activity Diagram of the Flow of Events:



Postcondition: The reserved bike will be marked as 'Reserved.'

Relationships: Upon using this use case, the Update Bike Status use case will also be used by the Web Server in order for the bike data to still be accurate.

Special Requirements:

Physical devices for tracking associated with the bikes function properly.

Use-Case Name: Use-Case 7.0 Lock Bike

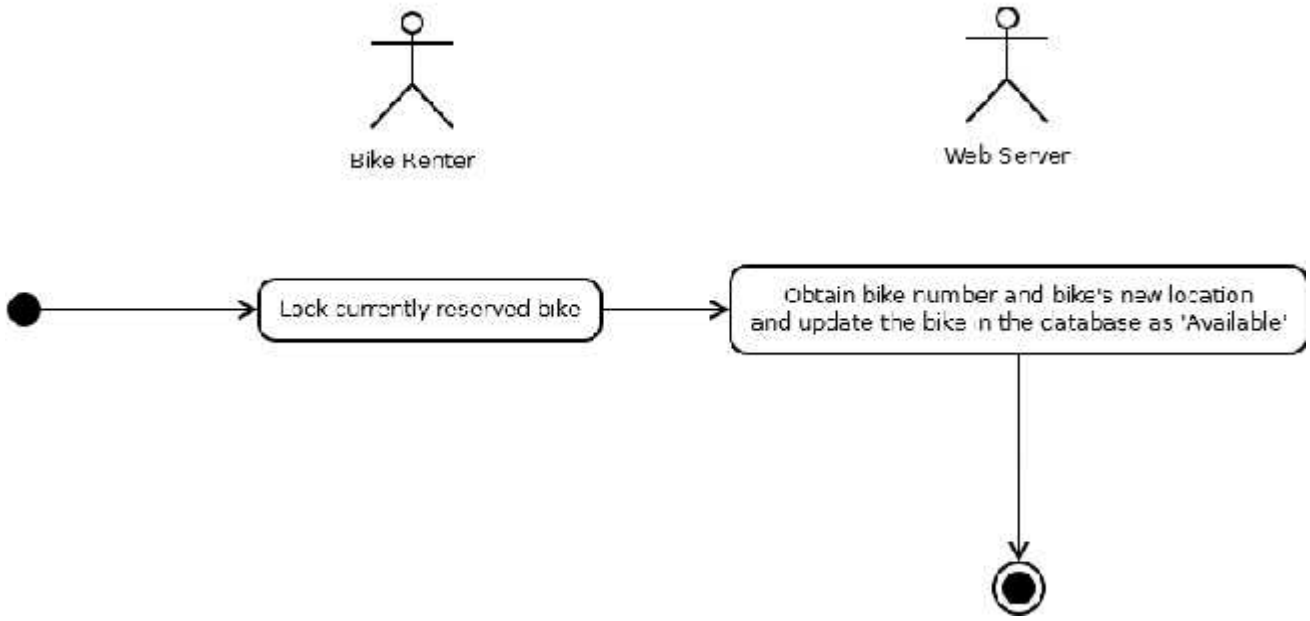
Description: After a Bike Renter has rented a bike and reached a destination, he/she can lock the bike at a safe location. This will be done through the Bike Share module, allowing the Web Server to update its data.

Preconditions: The Bike Renter reserved a bike through the Reserve Bike use case.
The Bike Renter is 'Logged In.'
There is a working connection with the Web Server.

Flow of Events:

Scenario Name	***Description***
Scenario 1 (Basic Flow) The Bike Renter locks his reserved bike.	1. The Bike Renter prompts the system that he/she will be locking his/her reserved bike at a station. 2. The Web Server obtains the bike number and the bike's new location then updates the database, placing the respective bike as 'Available.'
Scenario 2 (Connection Error)	1. If at any time in the basic flow, the connection with the Web Server is lost, the system will display a Connection Error, and log out the current user.

Activity Diagram of the Flow of Events:



Postcondition: The respective bike will be marked as 'Available.'

Relationships: Upon using this use case, the Update Bike Status use case will also be used by the Web Server in order for the bike data to still be accurate.

Special Requirements:
Physical devices for tracking associated with the bikes function properly.

Use-Case Name: Use-Case 8.0 Report Bike Issue

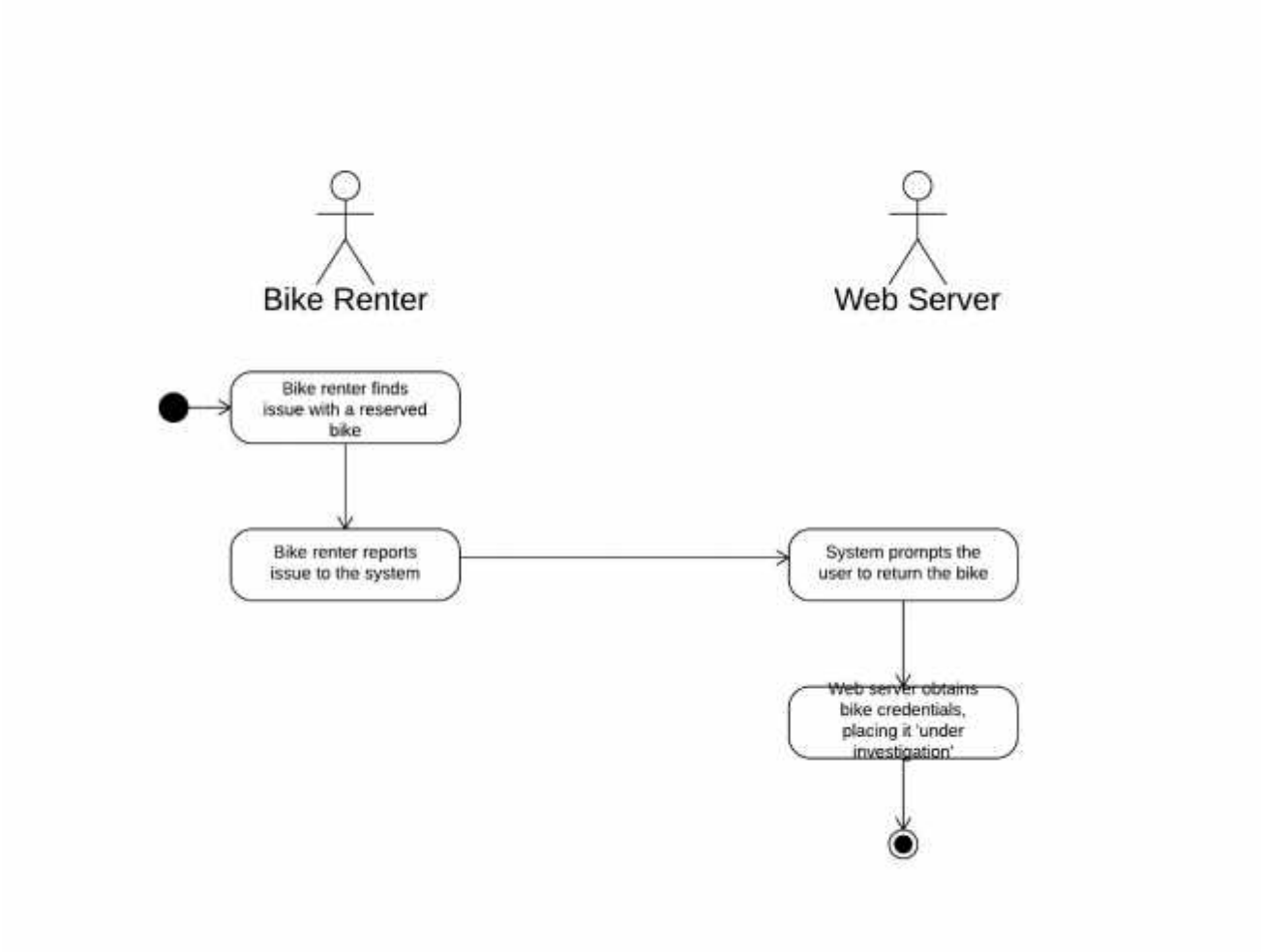
Description: While logged in, a Bike Renter can report any issues encountered with a reserved bike. Bikes reported such will be automatically locked once returned to a station. These bikes will be marked ‘Under Investigation’ while not yet checked, then be labeled ‘Invalid’ if a problem is confirmed.

Preconditions: The Bike Renter reserved a bike through the Reserve Bike use case.
The Bike Renter is ‘Logged In.’
There is a working connection with the Web Server.

Flow of Events:

Scenario Name	***Description***
Scenario 1 (Basic Flow) The Bike Renter reports an issue with a bike, and returns it to a station.	1. The Bike Renter finds an issue with a reserved bike. 2. The Bike renter reports the issue to the system, detailing the bike number, and the problem with the bike. 3. The system prompts the user to return the bike to a station. 4. The Web Server obtains the bike number and the bike’s new location then updates the database, placing the respective bike as ‘Under Investigation.’
Scenario 2 (Connection Error)	1. If at any time in the basic flow, the connection with the Web Server is lost, the system will display a Connection Error, and log out the current user.

Activity Diagram of the Flow of Events:



Postcondition: The respective bike is automatically locked upon returning it to a location.

Relationships: Upon using this use case, the Update Bike Status use case will also be used by the Web Server in order for the bike data to still be accurate.

Special Requirements:

Physical devices for tracking associated with the bikes function properly.

Use-Case Name: Use-Case 9.0 Update Bike Status

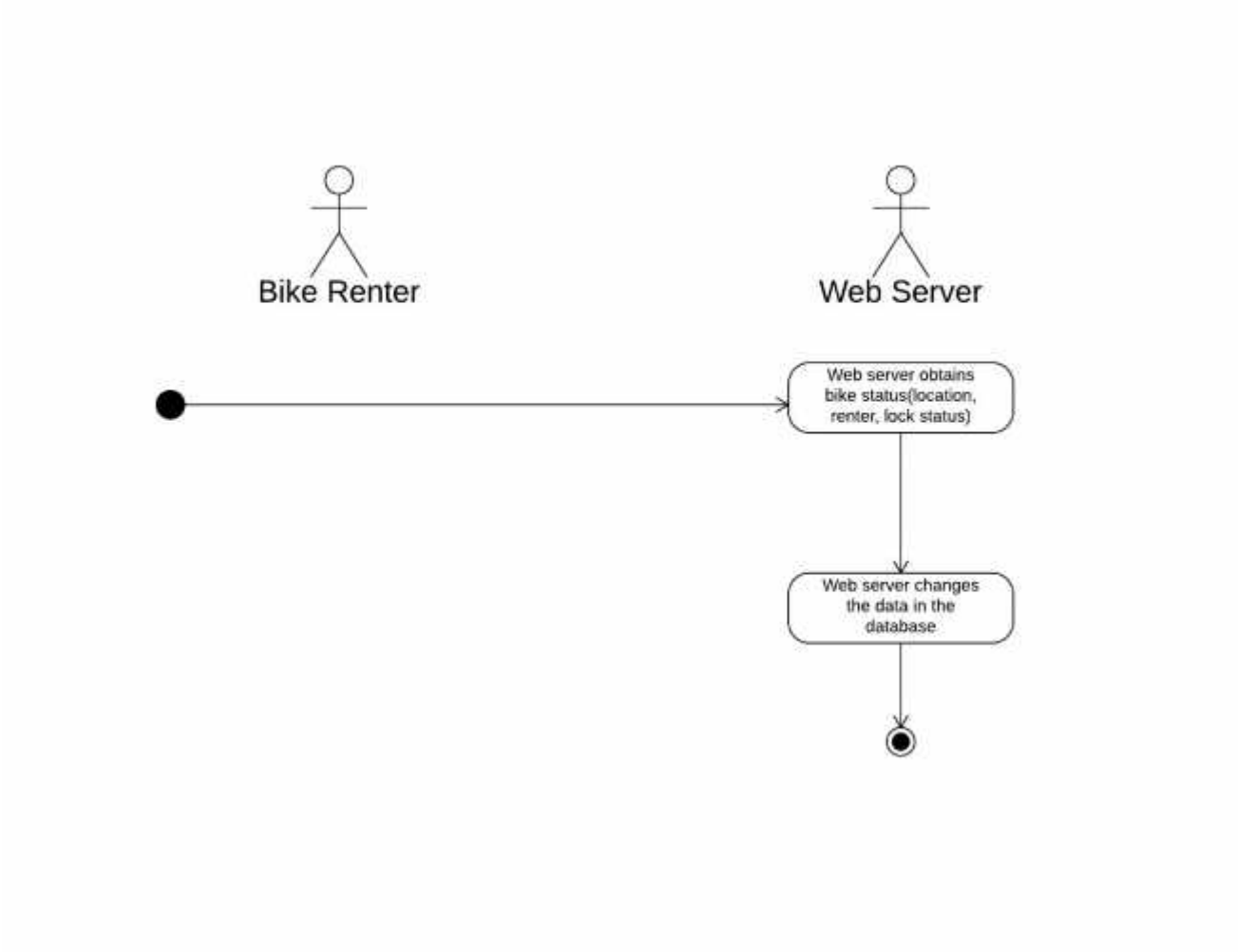
Description: The Web Server is tasked with updating the statuses of bikes such as their location, whether they are locked or not, the user that is renting it (if applicable), their ID and more. The Web Server updates the statuses of bikes whenever it processes a request from a bike renter such as locating bikes, renting bikes and locking bikes.

Preconditions: NONE

Flow of Events:

Scenario Name	***Description***
Scenario 1 (Basic Flow) The Web Server updates the bikes' statuses.	1. The Web Server obtains the necessary information such as location and state to change from another corresponding use case (Reserve Bike, Lock Bike, Report Bike Issue) and the id of the bike that will change status. 2. The Web Server changes the respective data in the database.

Activity Diagram of the Flow of Events:



Postcondition: NONE

Relationships: Used to present accurate data in the Search Available Bikes and Reserve Bike use cases.

Special Requirements:

Physical devices for tracking associated with the bikes function properly.