SOFTWARE ENGINEERING FOR GEOINFORMATICS PROJECT

RASD Version 1.0

Project name: BIKOD

Prepared by: AbdelGafar, Omar

Ruken, Dilara Zaf

Fakherifard Katayoun

Submission Date: 29.04.2019

Table of Contents

Revision History

Name	Date	Reason for Changes	Version

List of Tables

List of Figures

1.   Introduction

1.1   Purpose

The purpose of the Requirement Analysis and Specification Document (RASD) is to present a detailed description of a bike sharing system to monitor the bikes flow both in space and time and the capability to assess the system.

The document fulfils the following purposes:

- The RASD specifies all system capabilities that are needed to produce the desired interfaces (i.e. the location of the Bike stations) using the available data for data users.
- The RASD forms the baseline for the construction of the system, where construction means the (technical) process performed by the Developers, from the design, to the implementation and testing of the System.

1.2   Product Scope

- The RASD specifies the functionality of the web application (BIKOD) which are as following:
  1- Allowing the user to visualize the location of bike stations by maps
  2- Allowing the user to query about each station by interactive graphs
  3- Allowing the system admin to access the database for both the users and bikes and visualize the statistical data into interactive graphs.
- The RASD does not include requirements on the procurement, management and maintenance activities needed for the system, nor does it include financial ones. These are expected to be covered in management and financial documents elsewhere.

1.3   Appreciations

Appreciations	Meaning
RASD	Requirement Analysis and Specification Document
OSM	Open Street Maps
FR00	Functional Requirement
NFR	Non-Functional Requirement
TR	Technical Requirement
UC	Use Case

1.4   References

ISO/IEC/IEEE International Standard - Systems and software engineering -- Life cycle processes -- Requirements engineering," in ISO/IEC/IEEE 29148:2011(E), vol., no., pp.1-94, 1 Dec. 2011
doi: 10.1109/IEEESTD.2011.6146379

1.5   Overview

BIKOD is developed for everyone who interests to use bike sharing systems as transportation. This service can show the users number of available bikes and free stalls in each station in real time and also has the capability to show the stations separated around Milano city on the map. From an admin point of view, it can show the bike flow in graph format and visualize the statistical data. Additionally, the admin can access the user dataset and query it or change it. This web application was developed to run on every operating system.

In the RASD file, the overall description of the product, the requirements (functional, nonfunctional and technical) to reach the goals, constraints and use case models will be explained in order to clarify the development process.

2.   Overall Description
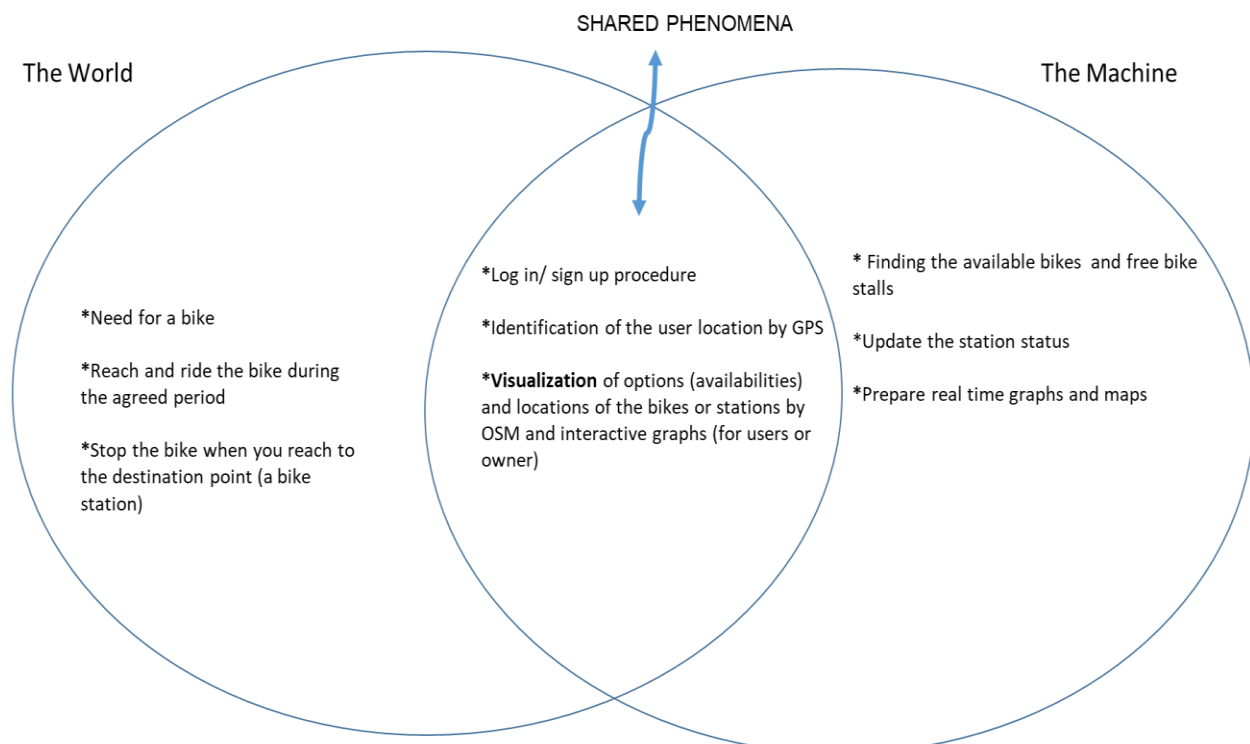
2.1   Product Perspective

Figure 1: Analysis of the world, the machine and the shared phenomena for BIKOD

The interaction between the world and the machine will be supplied by the three shared phenomena showed as in Figure 1.

The log in or sign up procedure is shared by the two worlds however it is controlled by the real world. The user has the control on this step.

Then the second interaction between the worlds will be during the identification of user's location by an external system which is GPS. This step will be controlled by the machine world.

Finally, the third step which is the main goal of the product is visualization. For this interaction, the machine world will control all the process, but the output will be shared with the real world. This visualization includes location of the stations and availabilities of the bikes which may be used by the users and the bike flow monitoring and statistical graphs for the admin.

2.2 Product Functions

For user:

- **Sign Up:** Users need to sign up to use the web site. The users should have a username and password. After filling their name, surname, they register the system.
- **Sign In:** If a user is signed up, s/he can sign in the system by filling username and password boxes.
- **Sign Out:** User may need to sign out the system. S/he can do it by clicking the sign out button which is placed in every page.
- **GPS Permission:** BIKOD asks the user for enabling his/her GPS.
- **Station selection:** The user selects the bike station s/he interested in.
- **Popup:** When the users click on the station, BIKOD will show up the status of the station, number available bikes, number of free stalls.
- **Query Station:** Users need to know the number of bikes in a previous hour. The users enter the requires hour, BIKOD give the number of bikes for this time.
- **Request interactive graph:** User request to see the number of bikes per station during any hour of the day till the queried time.

For Admin:

- **Sign In:** Admins can sign in the system by filling username and password boxes.
- **Sign Out:** Admins may need to sign out the system. S/he can do it by clicking the sign out button which is placed in every page.
- **Station selection:** The Admin selects the bike station s/he interested in.
- **Popup:** When the Admins click on the station, BIKOD will show up the status of the station, number available bikes, number of free stalls.
- **Query Station:** Admins need to know the number of bikes in a previous hour. The Admins enter the requires hour, BIKOD give the number of bikes for this time.
- **Data base access:** Admins have the authority to access the databases for both bikes and users.
- **Request interactive graph:** Admin request to see interactive statistical graphs through these graphs they can monitor the system.

2.3   User Classes and Characteristics

Typical users, anyone who wants to use BIKOD for transportation.
The admin (owner) of the system, who invested on this service (shall visualize the real-time graphs or maps of bike flow in each zone they want).

2.4   Design and Implementation Constraints

The system requires OSM which enables the system functionality. Because of this situation, when the server crashes the system will not be able to its operations until the server become available to respond system requests. In addition to this, sensors provided by the customer should be working properly otherwise users and admins may have difficulties to reach the correct information about the situation of stations. To sum up, BIKOD system has constraints in terms of reliability and functionality.

2.5   Assumptions and Dependencies

In order to use BIKOD service, the users must use a web browser and have an internet connection. OSM server should provide the map including the stations' locations in Milan city.

3.   Specific Requirements

3.1   Requirements

3.1.1   External Interface Requirements

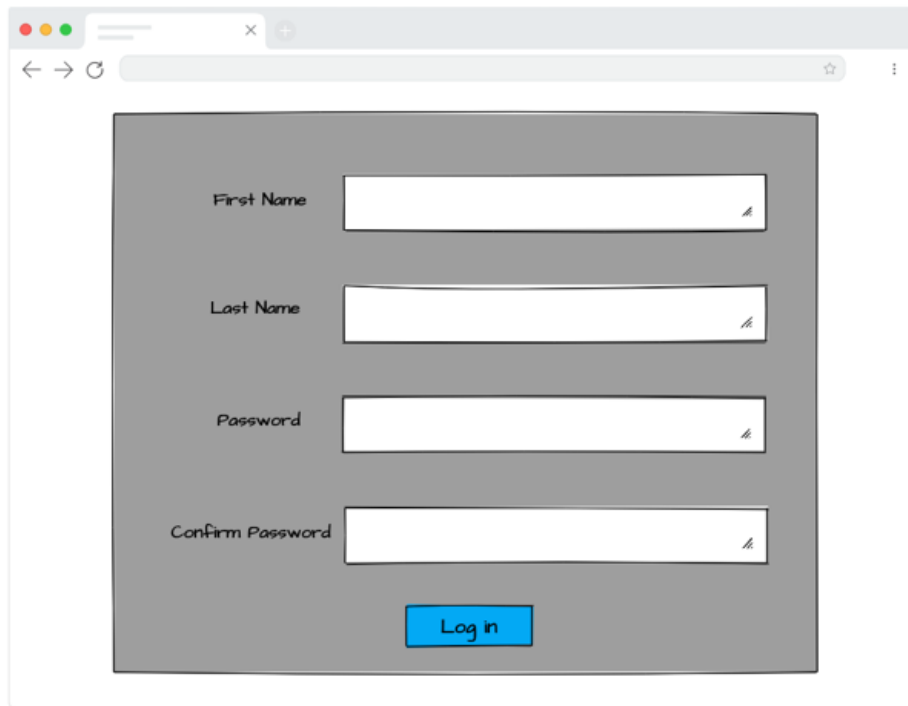#### 3.1.1.1   User Interfaces
The general interface for BIKOD is the sign up or directly Log in for both users and admins.

The user interface will have a sign up and log in page where the user can fill it with his or her/her username and password. The second page will include a map provided by Open Street Maps (OSM). On the map will have markers denoting the Bike stations. As the user scrolls over a marker, the number of available bikes, the number of free stalls and the status of the station (i.e. if the station is usable or under maintenance) will be showed up. The user will also be able to view an interactive graph shows number of bikes per station for the current day until the queried time.

The admin interface has two main pages. The first page will include the interface with the map which has the same functionalities with the user interface. The second page will allow the admin to reach various statistical graphs (i.e. the number of active users hourly/daily/weekly).
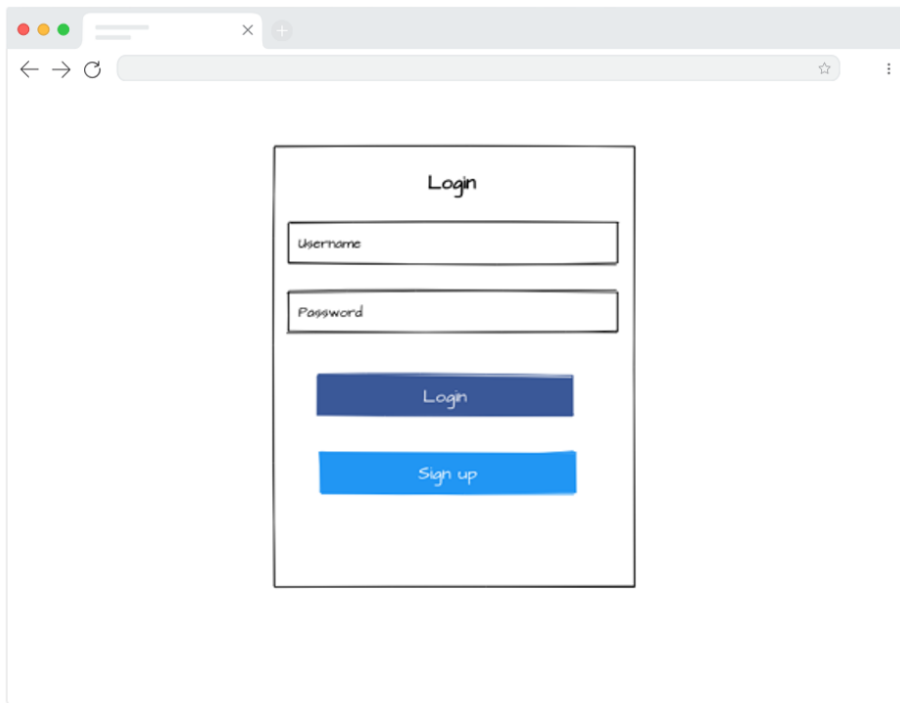
3.1.1.2   Software Interfaces
BIKOD requires web Browser (i.e. Google chrome, Firefox, etc.) to be installed on the system. The map is provided by Open street maps (OSM). BIKOD will be connected with PostgreSQL database to import a graph edge list.

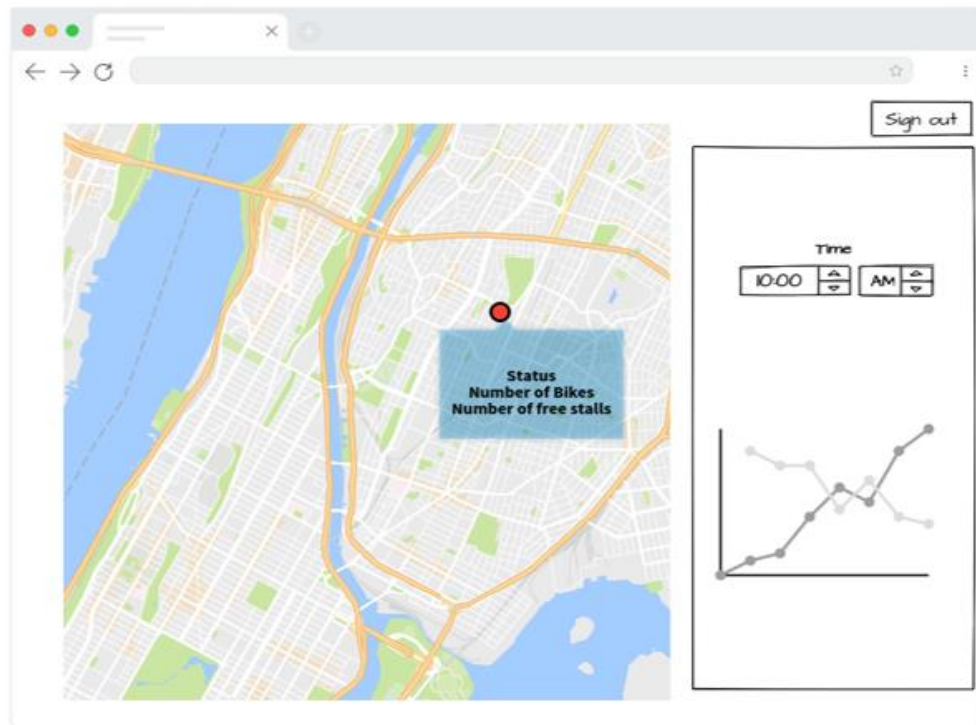*Figure 2: Sign Up Page*

Figure 3:Login Page
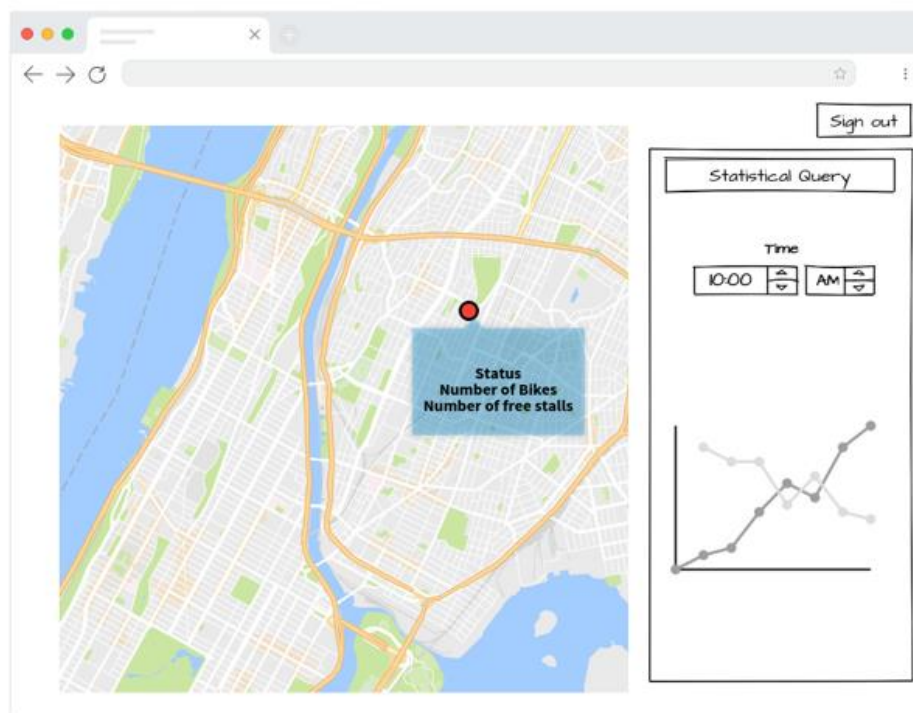
Figure 4: User Main Page
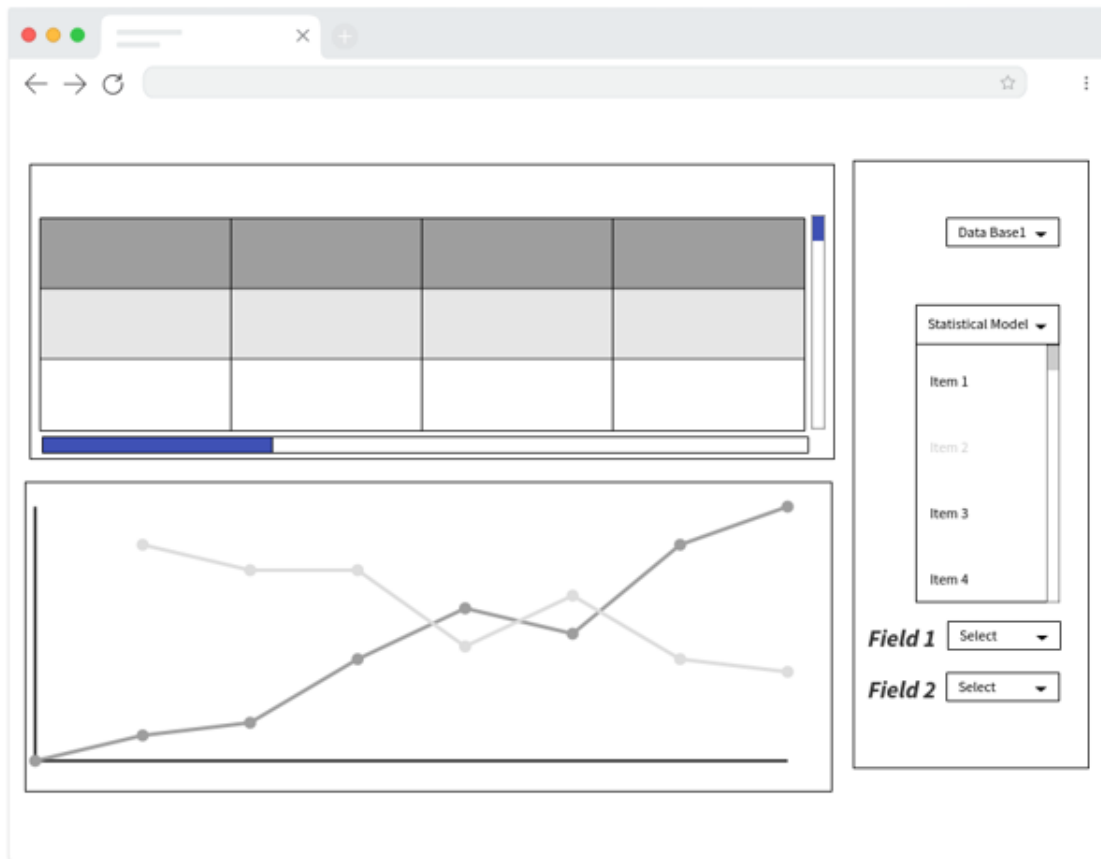

Figure 5: Admin First Page

Figure 6: Admin Second Page

3.1.2 Functional Requirements

3.1.2.1 User Class 1: BIKOD user

[FR001]: The system should allow the users to insert all the data required for the registration.
[FR002]: The system should ask for permission to access the user's location through GPS.
[FR003]: The system should show the user's and stations' location at the same time.
[FR004]: The system shall allow the user to select a certain station to see the bike and free stall situation there.
[FR005]: The system should provide an opportunity to user to query for a certain time if and only if it is before the queried time in the same day.
[FR006]: The system shall prepare station specific graphs for the current day.

3.1.2.2 User Class 2: BIKOD admin
[FR007]: The system shall allow the admin to select a certain station to see the bike and free stall situation there.
[FR008]: The system should allow the admin to access the database.

[FR009]: The system shall allow the admin to perform statistical operation on the databases (i.e. numbers of users, bikes, free stalls per week for all stations).
[FR010]: The system shall provide statistical graphs to the admins.
[FR011]: The system should allow admins to select their preferred mode for graphs.

3.1.3 Other Non-Functional Requirements

3.1.3.1 Performance Requirements
[NFR001]: The system shall be able to respond all the registered users simultaneously.
[NFR002]: The system shall be able to keep all the registered user information.

3.1.3.2. Design Constraints
[NFR003]: The system shall run on every browser (Chrome, Safari, Mozilla Firefox etc.) and operating system (Windows, Linux, Mac Mavericks etc.).

3.1.4 Technical Requirements

[TR001]: The sensors in the stations should be working properly.
[TR002]: The system should use Phyton as the programming language.

3.2 Use case Models

3.2.1. Registration

Table 1: Registration Use case UC01

Use Case ID	UC01
Actors	BIKOD User
Entry condition	The user opens his/her browser and serve for the Website and goes to sign up page
Flow of events	• The user opens the browser and types the address of the system. • User presses the sig up button. • User enters her or his user name, surname and password.
Exit Condition	• If all the required fields are properly filled, the user will be redirected to the main page of the system.
Exceptions	If one of the required fields (user name, user surname and password) in sign up page are not filled properly, the warning message will be shown by the system.
Special Requirements	None

3.2.2. Signing in

Table 2: Sign in Use Case UC02

Use Case ID	UC02
Actors	• BIKOD User • BIKOD Admin
Entry condition	User will be able to use the system
Flow of events	• The user/admin opens the browser and types the address of the system. • User/admin presses the log in button. • User/admin enters her or his user name and password. • User/admin enters her or his user name, surname and password.
Exit Condition	• If the user enters his or her user name and password information correctly, user will be redirected to the application relevant page of the system.
Exceptions	If one of the required fields (user name, password) in log in are not right, the warning message will be shown by the system.
Special Requirements	None

3.2.3. Sign Out

Table 3: Sign out Use Case UC03

Use Case ID	UC03
Actors	• BIKOD User • BIKOD Admin
Entry condition	User/admin will be able to use the system

Flow of events	• The user/admin presses the log out button. • User/admin leaves the system. • The system's main page will be loaded.
Exit Condition	User will be able to leave the system.
Exceptions	None
Special Requirements	None

3.2.4. Popup

Table 4: Popup Use case UC04

Use Case ID	UC04
Actors	• BIKOD User • BIKOD Admin
Entry condition	User/admin shall sign in to the system
Flow of events	• The user/admin selects the station by one click on it. • The web shows the status, number of available bikes and number of free stalls per station.
Exit Condition	User/admin can see information about the bike station.
Exceptions	The admin/ user must click on the station to make the popup disappear again
Special Requirements	None

3.2.5. Query

Table 5: Query Use Case UC05

Use Case ID	UC05

Actors	• BIKOD User • BIKOD Admin
Entry condition	User/admin shall sign in to the system
Flow of events	• The user/admin double click the required station. • The web shows a window at which the user/admin can enter the quarried time. • The web shows an interactive graph showing the number of bikes per time at the station till the quarried time. • The user/admin can press the exit button to close the window
Exit Condition	The user /admin sees the number of bikes for each station in an interactive graph.
Exceptions	None
Special Requirements	None

3.2.6. Statistical Query

Table 6: Statistical Query UC06

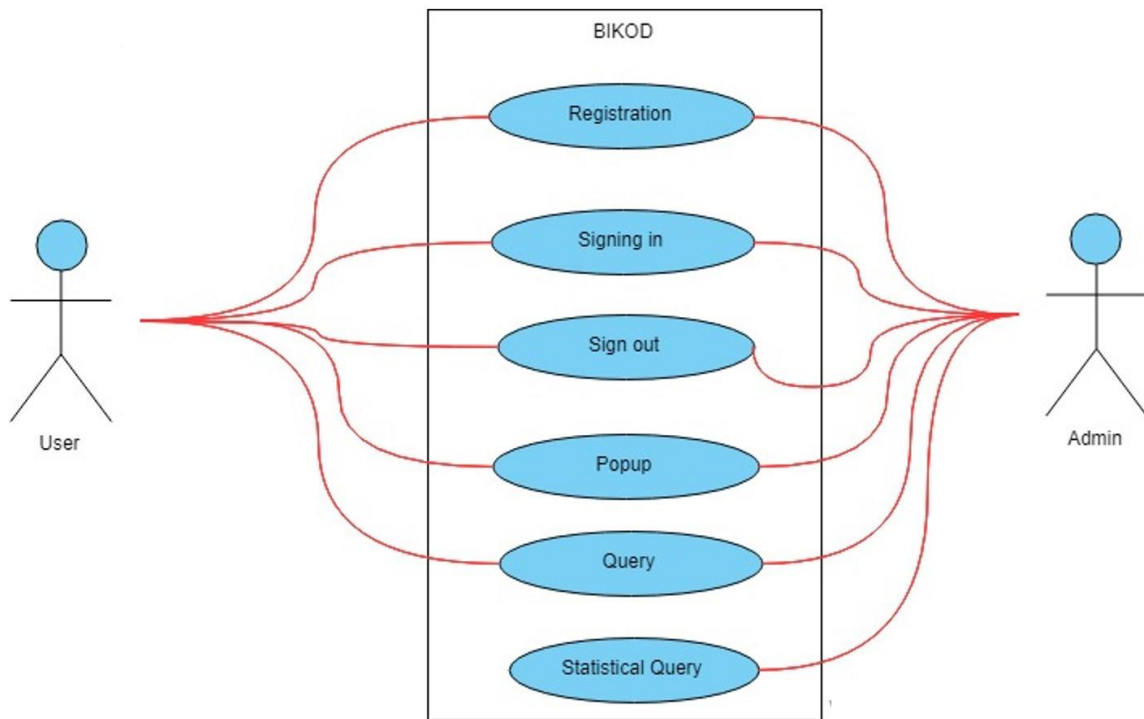| Use Case ID	UC06
Actors	• BIKOD Admin
Entry condition	Admin shall sign in to the system
Flow of events	• The admin clicks on Statistical query button. • The web opens a new window showing the data bases for users and bikes. • The admin performs statistical operations. • The admin requests interactive graphs for each station and for all stations.
Exit Condition	The admin monitors the bike sharing system through the statistical graph which allows him to take a decision
Exceptions	None

Special Requirements	None

Figure 7: ***Use Case Diagram for BIKOD Website***