Bicycle Rental Accounting System

*Explanatory Notes*

SENER Kaan B.

HEISKANEN Joonatan P.

NESTEROV Roman A.

**Contents**

[1. Task Statement 3](#_Toc437900356)

[2. Responsibility Matrix for Group Members 4](#_Toc437900357)

[3. Application Functions and Features 5](#_Toc437900358)

[3.1. Use cases 5](#_Toc437900359)

[3.2. User Interface 5](#_Toc437900360)

[4. Application Structural Description 6](#_Toc437900361)

[4.1. Data Structure Description 6](#_Toc437900362)

[4.2. Application Modules Description 6](#_Toc437900363)

[4.3. CRUD Queries Implementation 6](#_Toc437900364)

[4.4. Statistical Queries Implementation 6](#_Toc437900365)

[5. Overall Result Characteristics 7](#_Toc437900366)

[References 8](#_Toc437900367)

[Appendix A. Something useful code extracts (SQL or PHP)… 9](#_Toc437900368)

[6. Conclusion 10](#_Toc437900369)

# Development Problem Statement

This report contains a thorough description to the process of development process of the application “Bicycle Rental Accounting System”.

As for the main development objective, in concerns the automation of the routine tasks implemented by bicycle rental outlet shop assistants. There is a network of bicycle rental outlets offering the bicycles to the citizens. The application is to exploit the typical client-server architecture.

The accounting system is aimed at providing the following functionalities:

1. inserting reference information concerning bicycle features like brand, type and so on;
2. registering bicycles in a rental outlet (as well as updating, deleting this information);
3. renting and returning bicycles from and to the network of outlets (people is to return the bicycle to the outlet where they rented it);
4. gathering the statistics on the rental outlets operation including information on the average rental times for bicycles of specified types and brands as well as the revenue generated.

The “Bicycle Rental Accounting System” is to be implemented in the form of client web application employing the following instruments:

1. PHP for web-application development and markup;
2. Javascript and Bootsrap libraries are to be exploited to design and to add dynamics to the web application pages;
3. XAMPP is used to deploy the application on client machine.

The database for the accounting system is to be implemented using the tools and instruments provided by MySQL technologies and phpMyAdmin services to develop and deploy databases.

In addition, Trello instruments were used for the organization of the development process together with Github versioning control system.

# Responsibility Matrix for Group Members

This section indicates the main generic activities taken place while developing the accounting system and the contribution areas of each group member (see Table 1).

Table 1. Development Group Responsibility Distribution

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Database | | Application | | | Report | | Management |
| **Initial Design** | **Deployment** | **GUI and modules development** | **Necessary query specifying** | **Testing** | **Preparation** | **Presentation** | **Work coordination and load balancing** |
| **KAAN** | X | X | X | X | X | X |  |  |
| **JOONATAN** | X |  | X | X | X | X |  |  |
| **ROMAN** | X | X |  | X | X | X | X | X |

From this table we can observe that each group member was involved in the process of data processing connected both with the database itself development and with the specification of queries needed for the implementation of functions briefly discussed in the previous secrtion (they will be described below in the form of use cases).

# Application Functions and Features

This section provides a description to a functions implemented in the form of use cases and an insight to the inner organization of the user interface

## Use cases

There was identified two main roles of the system end-users as follows:

1. administrator (or admin) who is able to do many data-sensitive operations without restrictions;
2. ordinary user (bicycle outlet shop assistant) able to do the operation concerning the bicycle outlet disposal, rent-return operations and statistics viewing.

The following use cases were identified for the Admin:

***Use Case 1: Add bicycle to database***

*Primary Actor:* Admin

*Scope:* Admin dashboard

*Brief:* The admin adds a new bicycle by clicking the Add New Bicycle –button, inputting the required information into the appropriate fields on the form and clicking “Save”.

***Use Case 2: View all bicycles in database***

*Primary Actor:* Admin

*Scope:* Admin dashboard

*Brief:* Admin presses the View All Bicycles button and gets a table with all the data on all saved bicycles.

***Use Case 3: Add a new bicycle brand***

*Primary Actor:* Admin

*Scope:* Admin dashboard

*Brief:* The admin adds a new bicycle brand by clicking the Add New Bicycle Brand -button, inputting the brand name into the form and clicking “Save”.

***Use Case 4: View all bicycle brands***

*Primary Actor:* Admin

*Scope:* Admin dashboard

*Brief:* Admin presses the View All Bicycle Brands -button and gets a table with the names and ids of saved bicycle brands.

***Use Case 5: Add a new bicycle type***

*Primary Actor:* Admin

*Scope:* Admin dashboard

*Brief:* The admin adds a new bicycle type by pressing the Add New Bicycle Type -button, inputting the type into the form and clicking “Save”.

***Use Case 6: View all bicycle types***

*Primary Actor:* Admin

*Scope:* Admin dashboard

*Brief:* Admin presses the View All Bicycle Types -button and gets a table with the names and ids of saved bicycle types.

***Use Case 7 Add a new outlet***

*Primary Actor:* Admin

*Scope:* Admin dashboard

*Brief:* The admin adds a new outlet by inputting the name and address of the outlet into the form and clicking “Save”.

***Use Case 8: View all outlets***

*Primary Actor:* Admin

*Scope:* Admin dashboard

*Brief:* Admin presses the View All Bicycle Outlets -button and gets a table with the ids, names and addresses of saved bicycle outlets.

The following use cases were identified for the ordinary user (outlet shop assistant):

***Use Case 9: Register and sign in to the application***

*Primary Actor:* Shop employee

*Scope:* User Dashboard

*Brief:* User presses the Register or Sign up -button, inputs their name, email address and password and presses Sign up. Afterwards the login screen is shown, to where the user inputs their email address and password and presses log in.

***Use Case 10:* Edit information of outlet**

*Primary Actor:* Shop employee

*Scope:* User Dashboard

*Brief:* User clicks the Edit Shop -button, and can edit the name and address of the outlet he works in.

***Use Case 11:* Add new staff of outlet**

*Primary Actor:* Shop employee

*Scope:* User Dashboard

*Brief:* User clicks the Manage Staff -button, andthen fills in the name, email address, password and role of the new staff member. Finished by pression Add Staff –button.

***Use Case 12:* View or remove staff of outlet**

*Primary Actor:* Shop employee

*Scope:* User Dashboard

*Brief:* User clicks the Manage Staff -button, and a table with the name, email address and role of each staff member is shown. The user can click the Remove -button next to the staff member’s information to remove that particular staff member.

***Use Case 13:* View statistics**

*Primary Actor:* Shop employee

*Scope:* User Dashboard

*Brief:* User can click the Statistics -button to view different statistics about rental of bikes from his outlet.

***Use Case 14:*  Rent a bike to a customer**

*Primary Actor:* Shop employee

*Scope:* User Dashboard

*Brief:* When the employee rents a bike, he creates a new order by pressing the Manage Orders -button. If the customer is new, his name, address and phone number need to be filled in. If the customer has rented a bike before, the employee only has to input the phone number. All rentals where the bike has not been returned are shown in the Unfinalized Orders -tab.

***Use Case 15:* Add new bicycle to outlet**

*Primary Actor:* Shop employee

*Scope:* User Dashboard

*Brief:* User can click the Manage Bicycle -button, and can add a new bicycle to the selection of his outlet. Only the name, price of rental per hour, after how many hours the customer gets a discount, and how big the discount is, can be chosen by the employee. Other information is chosen from a dropdown menu. This data comes from the main database.

***Use Case 16: View or remove bicycles of the outlet***

*Primary Actor:* Shop employee

*Scope:* User Dashboard

*Brief:* User can click the Manage Bicycle -button to get a table of all the info of all the bicycles that this outlet has to offer. A bicycle can be removed by pressing the Remove –button next to the information of the bicycle to remove this bicycle from the list.

## User Interface

The organisation of modules and relations between them, instruments used for UI design and development, screenshot examples illustrating user interface structure….

….

# Application Structural Description

This section provides a description to the process of development the interaction between the client application and the database: how the database organized, what instruments are used to develop the database, the queries..

## Data Structure Description

The database diagram including brief description of each table of a database as well as the semantics of the relationships between them

…

## Application Modules Description

Brief Hierarchical modules description and how the connections are provided. What are the puproses of each module briefly.

## CRUD Queries Implementation

This section should briefly discover how the basic operation with data are implemented.

…

## Statistical Queries Implementation

This section briefly discuss how these queries are done and what sort of stat is provided…

…

# Overall Result Characteristics

Advantages and possible bottlenecks description as well possible ways to eliminate them in future, for example.

…

# Conclusion

# References

# Appendix A. Something useful code extracts (SQL or PHP)…