**Project Proposal On**

**Bike Rental System**

****

**Summited To**

**Sudeep Bajimaya**

**Summited by**

**Binay Thapa**

**NCC ID: 00175903**

**Batch: 23**

**Softwarica Collage of IT and E-commerce**

Table of Contents

[Table of figure and table 4](#_Toc23495180)

[Abstract 8](#_Toc23495181)

[Acknowledgements 8](#_Toc23495182)

[1 Introduction 8](#_Toc23495183)

[1.1 Introduction 8](#_Toc23495184)

[1.2 System background 8](#_Toc23495185)

[1.3 Overview of the project 8](#_Toc23495186)

[1.4 Justification 9](#_Toc23495187)

[1.5 Aims and Objectives 9](#_Toc23495188)

[2 Introduction of analysis 9](#_Toc23495189)

[2.1 Analysis methodologies 9](#_Toc23495190)

[2.2 Information gathering 13](#_Toc23495191)

[2.3 Feasibility study 13](#_Toc23495192)

[2.4 Software Requirements Specification 14](#_Toc23495193)

[2.4.1 Hardware and software 14](#_Toc23495194)

[2.4.2 Functional requirement 14](#_Toc23495195)

[2.4.3 Non-Functional requirements 16](#_Toc23495196)

[2.4.4 MoSCoW 17](#_Toc23495197)

[2.5 Use Case Diagram 19](#_Toc23495198)

[2.5 System architecture 21](#_Toc23495199)

[3 tier architecture 25](#_Toc23495200)

[Chapter 3- Design 26](#_Toc23495201)

[Introduction 26](#_Toc23495202)

[3.1 Structural modeling 26](#_Toc23495203)

[3.1.1 Class diagram 26](#_Toc23495204)

[3.1.2 Flow chart 27](#_Toc23495205)

[3.2 Behavioral Modeling 30](#_Toc23495206)

[3.2.1 Activity diagram 30](#_Toc23495207)

[3.2.2 Sequence diagram 34](#_Toc23495208)

[3.3.1 Data Dictionary 37](#_Toc23495209)

[3.3.2 ER diagram 40](#_Toc23495210)

[3.4 UI Mode 42](#_Toc23495211)

[Chapter 5 Testing 49](#_Toc23495212)

[5.1 Black box testing 49](#_Toc23495213)

[6 Other project issues 64](#_Toc23495214)

[6.1 Limitation 64](#_Toc23495215)

[6.2 Future Work 65](#_Toc23495216)

[6.3 Risk management 65](#_Toc23495217)

[6.4 Configuration management 65](#_Toc23495218)

[6.5 Other project issue 67](#_Toc23495219)

[6.6 User manual 67](#_Toc23495220)

[7 Conclusion 69](#_Toc23495221)

[References 70](#_Toc23495222)

[Appendix 71](#_Toc23495223)

# Table of figure and table

Figure1………………………………………………………………………………………………………………………13

Figure2………………………………………………………………………………………………………………………13

Figure3………………………………………………………………………………………………………………………21

Figure4………………………………………………………………………………………………………………………21

Figure5………………………………………………………………………………………………………………………25

Figure6………………………………………………………………………………………………………………………26

Figure7………………………………………………………………………………………………………………………28

Figure8………………………………………………………………………………………………………………………30

Figure9………………………………………………………………………………………………………………………32

Figure10………………………………………………………………………………………………………………………33

Figure11………………………………………………………………………………………………………………………34

Figure12………………………………………………………………………………………………………………………36

Figure13………………………………………………………………………………………………………………………36

Figure14………………………………………………………………………………………………………………………38

Figure15………………………………………………………………………………………………………………………39

Figure16………………………………………………………………………………………………………………………39

Figure17………………………………………………………………………………………………………………………39

Figure18………………………………………………………………………………………………………………………40

Figure19………………………………………………………………………………………………………………………40

Figure20………………………………………………………………………………………………………………………40

Figure21………………………………………………………………………………………………………………………42

Figure22………………………………………………………………………………………………………………………43

Figure23………………………………………………………………………………………………………………………43

Figure24………………………………………………………………………………………………………………………44

Figure25………………………………………………………………………………………………………………………44

Figure26………………………………………………………………………………………………………………………45

Figure27………………………………………………………………………………………………………………………45

Figure28………………………………………………………………………………………………………………………46

Figure29………………………………………………………………………………………………………………………46

Figure30………………………………………………………………………………………………………………………47

Figure31………………………………………………………………………………………………………………………47

Figure32………………………………………………………………………………………………………………………48

Figure33………………………………………………………………………………………………………………………48

Figure34………………………………………………………………………………………………………………………49

Figure35………………………………………………………………………………………………………………………49

Figure36………………………………………………………………………………………………………………………51

Figure37………………………………………………………………………………………………………………………52

Figure38………………………………………………………………………………………………………………………52

Figure39………………………………………………………………………………………………………………………53

Figure40………………………………………………………………………………………………………………………56

Figure41………………………………………………………………………………………………………………………56

Figure42………………………………………………………………………………………………………………………55

Figure43………………………………………………………………………………………………………………………55

Figure44………………………………………………………………………………………………………………………55

Figure45………………………………………………………………………………………………………………………55

Figure46………………………………………………………………………………………………………………………56

Figure47………………………………………………………………………………………………………………………56

Figure48………………………………………………………………………………………………………………………57

Figure49……………………………………………………………………………………………………………………58

Figure50……………………………………………………………………………………………………………………58

Figure51………………………………………………………………………………………………………………………59

Figure52………………………………………………………………………………………………………………………60

Figure53………………………………………………………………………………………………………………………60

Figure54………………………………………………………………………………………………………………………61

Figure55………………………………………………………………………………………………………………………61

Figure56………………………………………………………………………………………………………………………62

Figure57………………………………………………………………………………………………………………………62

Figure58………………………………………………………………………………………………………………………62

Figure59………………………………………………………………………………………………………………………64

Figure60………………………………………………………………………………………………………………………64

Figure61………………………………………………………………………………………………………………………65

Figure62………………………………………………………………………………………………………………………67

Figure63………………………………………………………………………………………………………………………68

Figure64………………………………………………………………………………………………………………………68

Figure65………………………………………………………………………………………………………………………69

Figure66………………………………………………………………………………………………………………………69

Figure67………………………………………………………………………………………………………………………70

Figure68………………………………………………………………………………………………………………………70

Figure69………………………………………………………………………………………………………………………72

Figure70………………………………………………………………………………………………………………………73

Figure71………………………………………………………………………………………………………………………73

Figure72………………………………………………………………………………………………………………………74

Figure73………………………………………………………………………………………………………………………75

Figure74………………………………………………………………………………………………………………………76

Figure75………………………………………………………………………………………………………………………78

Figure76………………………………………………………………………………………………………………………79

Figure77………………………………………………………………………………………………………………………79

Figure78………………………………………………………………………………………………………………………80

Figure79………………………………………………………………………………………………………………………82

Figure80………………………………………………………………………………………………………………………82

Figure81………………………………………………………………………………………………………………………84

Figure82………………………………………………………………………………………………………………………85

Figure83………………………………………………………………………………………………………………………86

Figure84………………………………………………………………………………………………………………………86

Table1………………………………………………………………………………………………………………………….17

Table2………………………………………………………………………………………………………………………….17

Table3………………………………………………………………………………………………………………………….19

Table4………………………………………………………………………………………………………………………….24

Table5………………………………………………………………………………………………………………………….24

Table6………………………………………………………………………………………………………………………….49

Table7………………………………………………………………………………………………………………………….49

Table8………………………………………………………………………………………………………………………….49

# Abstract

Bike rental system is web based system where user can book bike and to book the bike user most be register. After register user can login and book the bike. Use also can give feedback to admin and to give feedback user does not have login. In this document task are divided into seven parts. Each task have its own importance. The problem I face when I developing this system is in testing. All the other feature of this system is running smoothly. In analysis I gather the requirement of the user and made user case diagram and data flow diagram. After the analysis their come design part in this part it includes activity diagram, sequence diagram, proto type. After that there is codding and testing. In last there are other project issues and conclusion of this document.

# Acknowledgements

I would to thank Sudeep Bajimaya sir for his expect advice and encouragement for this project, as well as my classmate for their supported me greatly and were always willing to help me.

# 1 Introduction

## 1.1 Introduction

My system is bike rental. It is web base system. In here can hire a bike for the rent. This system is made for **Everest bike hire**. This system allow user to book bike for the rent. User also and send the feedback to admin. Admin handle all back end. Admin can add, delete, update information about the bike.

## 1.2 System background

**Everest bike hire** is establish in 2014 A.D. Company use paper base at first now they want to new system where their customer can book bike. They believe it will increase their customer. In system customer have register to book the bike. If they are register they can login and book bike. User also can cancel book if they want to.

## 1.3 Overview of the project

Main function of this system is customer can book bike. If they want they can cancel bike. Admin can add, delete, update information about bike. Admin can view feedback from the user. To give feedback user have not to be login. In website user can see location of the company.

## 1.4 Justification

This system objective to user can book bike from anywhere. User don’t have to visit company only for booking or to know the bike information. All information and booking can done easily all data are store in the database it will make work more efficient.

## 1.5 Aims and Objectives

* To provide the customer easy access to book bikes.
* To view details of the bike.
* To know the price to hired the bike.
* To communicate between user and company.

# 2 Introduction of analysis

Analysis is the method of breaking down a something into its parts to learn what they do and how they relate to one another. Analysis we can know about the problem and helps to make the decision.

In my project with helps of analysis it will helps which analysis methodologies is best for my project. It also show the feasibility of my project. With the help of the analysis I can gather information about the requirements by interview, groups focus, questionnaires to user. After gathering the requirements I can make decisions which features is must have in the project and other features will be should have, could have and won’t have. In analysis I can know about which hardware and software is gone use in the project. Through analysis I can separated functional and non-functional of the project. In analysis it show use case diagram to show how system will work.

By this way analysis play vital role in my project. With analysis I can make decision for my project by choosing the methodologies gathering requirements which requirements is most importance for the project so I can give more time on that requirement.

# 2.1 Analysis methodologies

A methodology offers a structure to follow when working on a project and makes the analysis and design more manageable. For this I choose hard systems methodology. In hard system methodology focuses on technical part of the system. In hard system methodology I did Structured Systems Analysis and Design Method. The reason are:

* This is waterfall method for analysis of the information system.
* It use Logical data modeling
* It use Data Flow Modeling
* It use Entity Event Modeling
* It ensures thorough planning and scheduling

**Data flow diagram**

A data flow diagram (DFD) illustrates how data is processed by a system in terms of inputs and outputs. As its name indicates its focus is on the flow of information, where data comes from, where it goes and how it gets stored. (Anon., 2011)

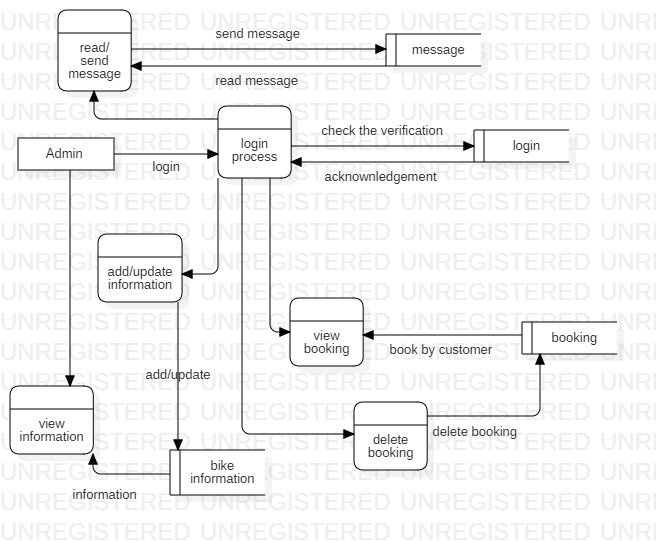


Figure 1: admin data flow diagram

This is admin data flow diagram. In here we can see firstly admin login then it check it verified or not then after it verified admin can add/update the information, view the booking and delete the booking, read and send message to customer. To view the information about bike admin does need login.

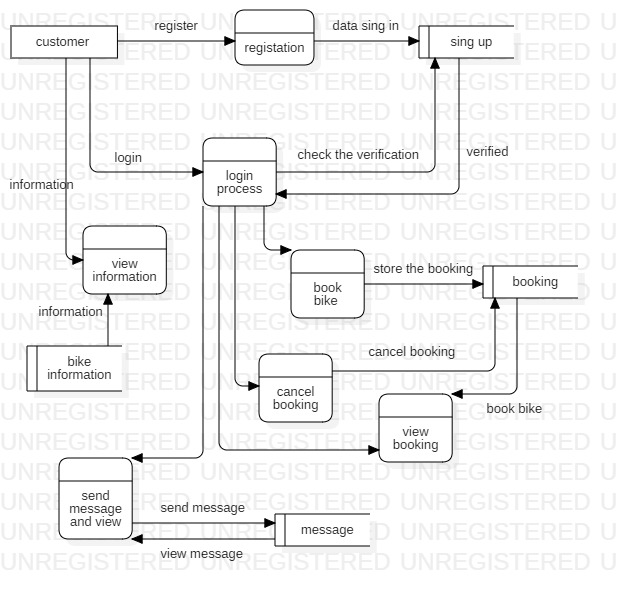


Figure 2: customer data flow diagram

In this figure firstly customer have to register of the customer is ready register then customer can login. In here also customer does not have to login to view the information about bike. After login customer can book the bike, view booking, cancel the booking. Customer can send message to admin and view message to admin.

In figure 1 and 2 we can see how data is process and store in database.

# 2.2 Information gathering

User are the main source of the information in the project. Through the user we can know the problem and requirements of the project. By taking information we make solution of the problem. For the information gathering I have interview the staff of the organization. After interviewing them I find out that about eighty percent of staff is not happy with this manual system in organization. Main problem is the customer does not know what are the paper is need for booking, and also to find the location of the organization. Customer does not know what type of the bike available in the organization. About ten percent does not have the problem and other ten percent is neutral. Through the interview I have gather information which will to make successful project. To make successful project I have make the website which in where all information about bike. And what kind paper is need to book the bike and it should also have the location of the organization.

# 2.3 Feasibility study

Feasibility study is an evaluation and analysis of the potential impact of a proposed project. With help of feasibility study I can find out my project is sufficient for development by technical, economical, time, legal and social. The reason are.

* **Technical:** This small project all hardware and software need for this project are easily available in market. This website can run in mobile also from any place world booking can be done. In technical this project is feasibility.
* **Economical:** For this project all the software is used are open source. Hardware is also available. And collage project. So economical this project can be done without any problem.
* **Time:** To complete this project we have time of six months. In six months we can do analysis, designs, coding and testing, maintenance. Within time I can complete the task.
* **Legal:** This collage project so there will no copyright issue in the project. This is legal project.
* **Social:** This website can used by everyone. Anyone can book the bike and view the information.

# 2.4 Software Requirements Specification

Software requirements specification show the functional and non-functional requirements. It also show the interaction between customer and the system through the use care diagram. So it is importance in analysis cause it description of the functional and non-functional requirements of the software.

## 2.4.1 Hardware and software

Hardware and software give the information about the hardware and software used to develop the system

**Hardware**

* This system is develop in Dell inspiron 15 3000 series
* This system can run any hardware if change are require.

**Software**

* Software need to develop is Xampp because programming language written in PHP
* Xampp is cross platform it can run in windows, Linux and Mac.

## 2.4.2 Functional requirement

Functional requirement is that requirement which is need to run the system without this requirement system can’t be complete. So this are the functional requirements of the system

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Title** | **Description** | **Rational** | **Dependencies** |
| F001 | Registration | User can register and login for booking | Without registration user can’t book. | NA |
| F002 | Login | In login user can book and send message and admin can read message | Without login other feature can’t be use to use the other feature login is needed | F001 |
| F003 | View bike information | User can view details of the bike available in company | After looking detail of the bike customer can book bike as their wish | NA |
| F004 | Book bike | Customer can book the bike available is company | After booking customer can rent the bike | F001, F003 |
| F005 | Add and update bike information | Admin can add and update information of the bike | if bike information is wrong enter then admin and update book and add new information about bike | F001 |
| F006 | Cancel booking | User can cancel the booking | If user has book wrong bike and user can cancel the booking | F001, F004 |
| F007 | Delete booking | Admin can delete the booking of the user | If user cancel the booking them admin can delete the booking | F001, F004, F006 |
| F008 | Send message | User and admin can communicate each other | If user find any problem or does not understand requirement user can message the admin | F001 |
| F009 | Read message | User and admin can read message each other | After message is send user and admin can read message then replay. | F008 |
| F0010 | Online payment | After booking user can send the payment through bank | To the money | NA |
| F0011 | Refund | After payment has done customer can get refund | If customer cancel the booking payment has done then they will gets their refund | F009 |
| F0012 | Delivery bike | To delivery the bike. | After bike is booked bike can delivery in user location | F004 |
| F0013 | Exchange bike | To exchange the bike rental | User can exchange the bike | F004 |
| F0014 | Location | It have the location of the company | User can easily find the place the location is provided | NA |
| F0015 | User can delete | User can delete the account | If user does not want to use this system they can delete their account | F001 |
| F0016 | Add admin | Admin can add other admin as well | Alone admin may not handle the detail of the system so they can add other admin | F002 |
| F0017 | delete account | Admin can delete the account | Admin can delete account of the add new admin | F0016 |

*Table1: functional table*

# 

## 2.4.3 Non-Functional requirements

Non-functional requirement means that system can run without it but if this functional is add then it will good.

There are some non-functional requirements.

|  |  |  |
| --- | --- | --- |
| **ID** | **Title** | **Description** |
| NF001 | Security | In security only valid user can use the system. To do that user authenticated is needed |
| NF002 | Usability | The system should be ease learn and satisfaction content of the use. |
| NF003 | Reliability | In here we look how to protect the failure to make trustworthy system. |
| NF004 | Maintainability | If in system need in any modification then modification should be done easily. |
| NF005 | Performance | System should response in time. It should perform in real time speed. |
| NF006 | Implementation | System must run in any browser from customer/client. No other things need to installation is needed |
| NF007 | Interface | It is part of interact. So inter face should be in web based. |
| NF008 | Legal | User data must be private. |

*Table2: non- functional table*

## 2.4.4 MoSCoW

MoSCOW stand for

M=Must haves in system

S= Should haves in the system

C=Could haves in the system

W= Won't haves in the system

With the help of the MoSCow I can complete the task in time. By using I will know which function have to done first and focus on the function. I should focus in must haves. Here are the list of the function and which are MoSCoW

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Requirement** | **MoSCoW** | **Rational** |
| F001 | Registration | M | To use the system |
| F002 | Login | M | To use feature of the system |
| F003 | View bike information | M | To provide the detail of the bike |
| F004 | Book bike | M | Fundamental function of system |
| F005 | Add and update bike information | M | Provide system add new details |
| F006 | Cancel booking | M | Provide the cancel booking for user |
| F007 | Delete booking | M | Provide to delete  the booking to admin |
| F008 | Send message | M | Provide the communication |
| F009 | Read message | M | after message is send it must be to read |
| F0010 | Online payment | W | User can pay after receiving the bike |
| F0011 | Refund | W | Payment is in won’t have so there will be no refund |
| F0012 | Delivery bike | W | Customer have to come themselves to receive |
| F0013 | Exchange bike | W | User can exchanges the bike |
| F0014 | Location | M | Provide the location of the company |
| F0015 | User can delete | C | To use the feature of the system |
| F0016 | Add admin | S | Function of the admin |
| F0017 | Delete account | S | Function of the admin |
| NF001 | Security | M | In system it is foundation function |
| NF002 | Usability | M | In system it is foundation function |
| NF003 | Reliability | M | In system it is foundation function |
| NF004 | Maintainability | M | In system it is foundation function |
| NF005 | Performance | M | In system it is foundation function |
| NF006 | Implementation | M | In system it is foundation function |
| NF007 | Interface | M | In system it is foundation function |
| NF008 | Legal | M | In system it is foundation function |

*Table3: MoSCoW table*

# 2.5 Use Case Diagram

Use case diagram show interact of system and user.

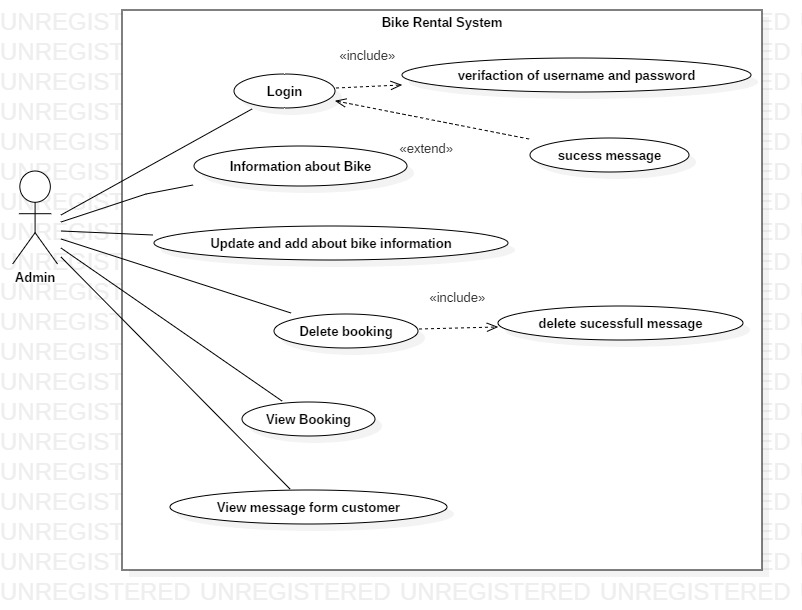


Figure 3: admin user case diagram

This is use case diagram of admin is the primary actor. In here we can see how admin interact with the system. Admin have login in the system to add and update the bike information, delete the booking, view the booking and view the message for customer.

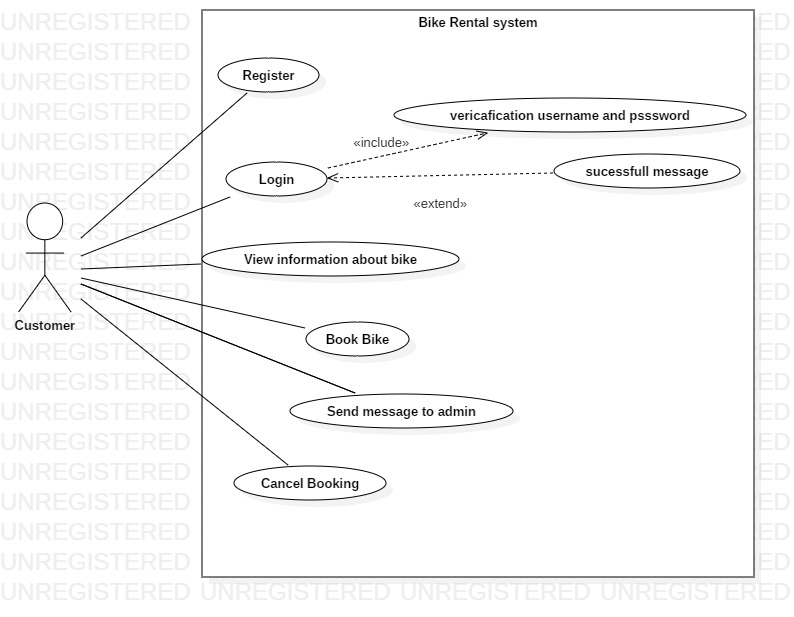


Figure 4: customer user case diagram

In this figure customer is the primary actor. As we ac seen in this figure first register have then customer can login to book the bike the send the message to admin and cancel the booking.

# 2.5 System architecture

The systems architect is a professional figure in information and communications technology. In system architecture I have done class diagram to class I have to follow the NLA (Natural Language Analysis) steps to do NLA I have create the scenario.

**Scenario**

Everest bike hire the company based in Kathmandu, Nepal. The company is established in 2014. The company provide the service to rent the bike.

So the company want to create their own website for customer where they can book the bike the view details of the bike, can message to the admin and have location of the company.

Customer can register and login allow then to book the bike, cancel the booking view the booking send and read message.

And admin can add the other user and add/update the bike information view the booking view the message form the customer and delete the booking.

**Natural Language Analysis (NLA)**

Natural Language Analysis (NLA) is to find the nouns, verb and adjectives from the descriptive text. With helps of NLA we can *find* candidate class, attribute and their relationship.

**The step of the NLA**

**List of the nouns**

Everest, bike, hire, company, information, company, rent, bike, company, customer, booking, bike, details, bike, message, location, company, bike, message, bike, booking, customer, message, booking, admin

**List of the verbs**

Create, add, update, book, register, login, book, add, update, cancel, delete, add, delete.

**Steps 1: Removing in repetition in nouns and verbs**

**Nouns**

Bike, company, customer, booking, details, message

**Verbs**

Add, delete, book, update

**Steps 2: Removing synonyms nouns / verbs**

**Nouns**

Rent, information

**Verbs**

Non available

**Steps 3: removing the nouns which does not have specific meaning**

**Nouns**

Everest

**Verbs**

Non available

**Steps 4 Remove out of scope**

**Nouns**

Company

**Verbs**

Non available

|  |  |  |  |
| --- | --- | --- | --- |
| **SN** | **Noun identification for the class** | **Selected as class** | **Justification of selection and rejection for the class** |
| 1 | Bike | Yes | They will many bike |
| 2 | Hire | Yes | It is fundament class |
| 3 | Book | Yes | It is fundament class |
| 4 | Location | No | Does not give any meaning |
| 5 | Details | No | Does not give any meaning |
| 6 | Customer | Yes | It is fundament class |
| 7 | Admin | NO | Does not give any meaning |

*Table 4: selection class table*

|  |  |  |  |
| --- | --- | --- | --- |
| **SN** | **Identification of verb** | **Use as method** | **description** |
| 1 | Add | NO |  |
| 2 | Create | Yes | Allow to create user |
| 3 | Delete | Yes | Allow to delete |
| 4 | Login | No | There is create |
| 5 | Register | No | There is create |
| 6 | Update | Yes | Allow to update data |
| 7 | Cancel | Yes | Allow to cancel |

*Table5: verbs table*

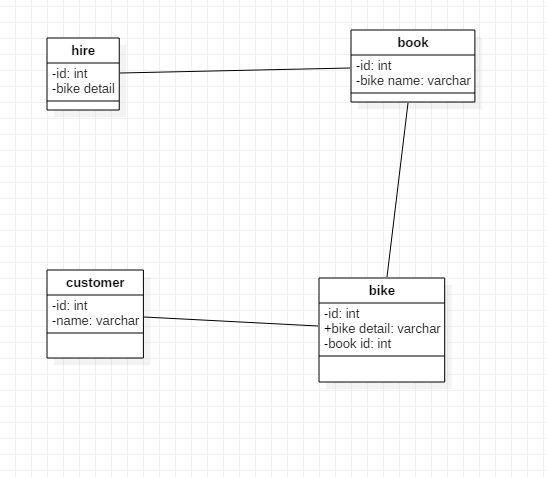
****

Figure 5: initial class diagram

This is class diagram. In this figure classes are only association with each other and few attribute. While doing task class relation can be change so they only associated now. In my project this kind classes.

## 3 tier architecture



Figure 6: three tier architecture

For the system architecture I have choose three tire architecture. The reason the why I choose three tier architecture:

* There are unlimited client and they access from anytime and anywhere from the world. So there will no problem in scalability.
* It more secure because client will not does not interact directly with database. All data from will process in application server first then it will go to database.

# Chapter 3- Design

# Introduction

The logical and physical planning of the project is called design (Anon., 2015). In design we show the database design of the system, logical of the program and how they interact with the system. With the help of design we find the problem and their solution. So, it is a general repeatable solution to a commonly occurring problem in software development.

In design there is four phase they are:

* Structural modeling
* Behavior modeling
* Database modeling
* UI (User Interface) modeling

# 3.1 Structural modeling

Structural model is the framework for the system. Class diagram is the example of the structural modeling. (Anon., 2015)

## 3.1.1 Class diagram

Class diagram is the static design of the system. It address the static view of an application. Class diagram shows a set of classes, interface, collaboration and their relationships. (Anon., 2009)

**Justification for this approach**

The reason I choose this are

* It describe the static view of system
* It show the collaboration among the basics of the static view.
* It also describe the functionalities of the system.
* Construction of the software application using object oriented language

**Diagram**

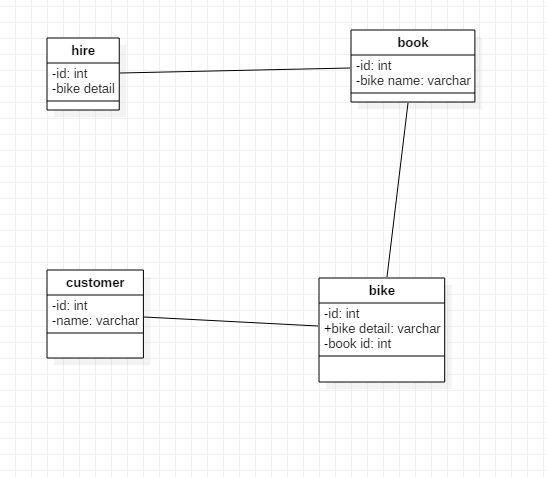
****

Figure 7: class diagram

This is not finally class diagram. Finally class diagram will be in final documentation.

## 3.1.2 Flow chart

Flow chart is graphical representation of our workflow. This often used in design phase of programming to work out the logical flow of the program (Anon., 2009). It also a step by step method to solving the problem. That’s why I used flow to my project.

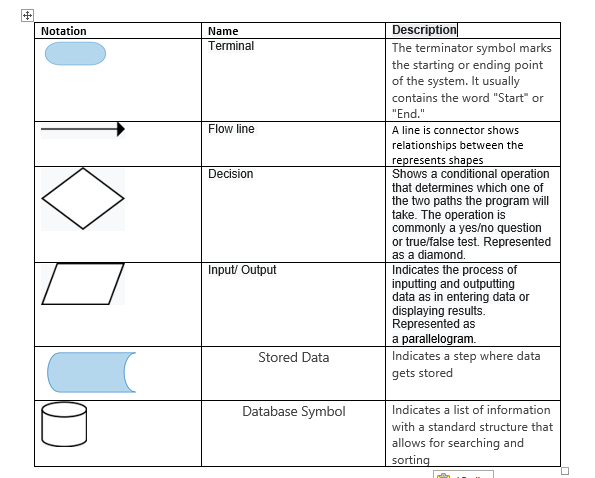
**Justification for this approach**

Here are the some reason how flow chart used in my project.

* It make our logic more clear.
* It helps during writing of programing
* This flow chart is very helpful to make testing and debugging easy
* It is best way of representing sequence of steps in an algorithm

**Notation used**

This are notation use in this flow chart

****

**Diagram**

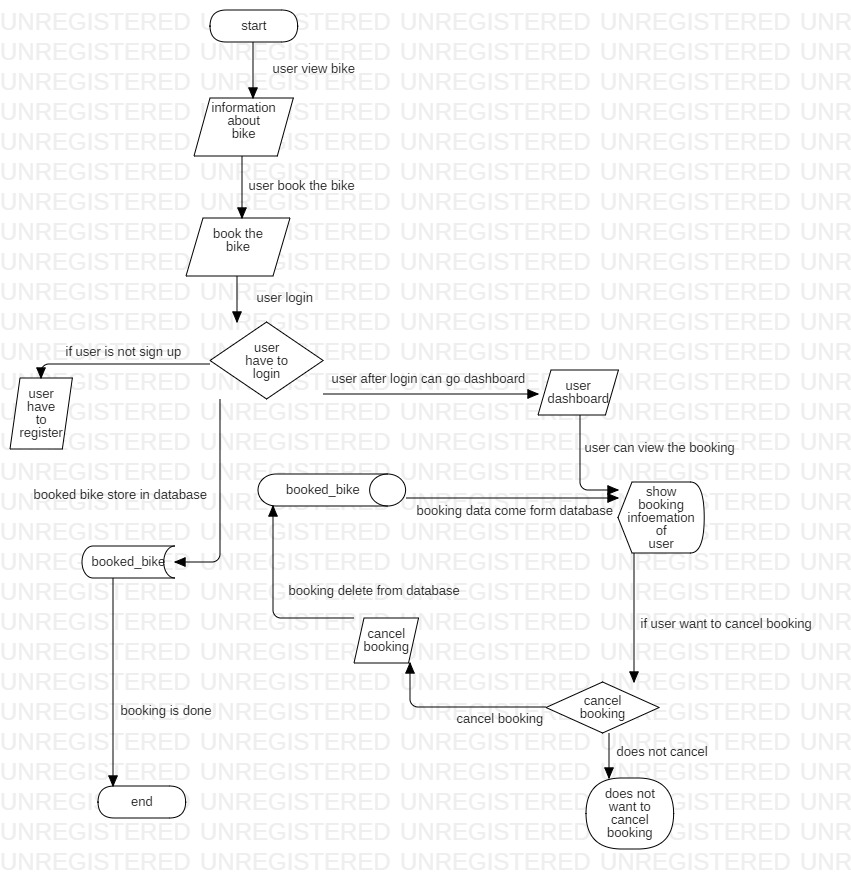
****

Figure 8: flow chart

In this flow show the how user book bike. Firstly user view the information about bike then booked the bike for rent. To book the bike user have to login if user have to register to do login. After login user can book the bike. After login user can go to dashbord and view bookinf if user want then cancel the booking aslo. Booked bike data will store in database. If user cancel the booking then booking data also remove from database.

# 3.2 Behavioral Modeling

It is dynamic behavior of the system. It show what happens or what is supposed to happen when a system responds to a stimulus from its environment.

## 3.2.1 Activity diagram

Activity diagram describe the dynamic aspects of the system. It is basically a flowchart to represent the flow from one activity to another activity.

**Justification for this approach**

Here are some benefits of activity diagram in my project.

* Reveal the logic of an algorithm
* It describe the step completed in UML use case
* It show the show process of business between user and system
* Define the parallel, branched and concurrent flow of the system
* Examining business requirement at future period.

**Notation used in diagram**

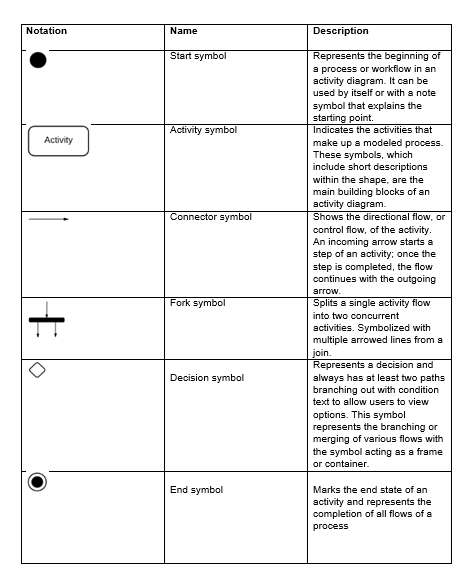
****

Figure 9: notation

**Diagram**

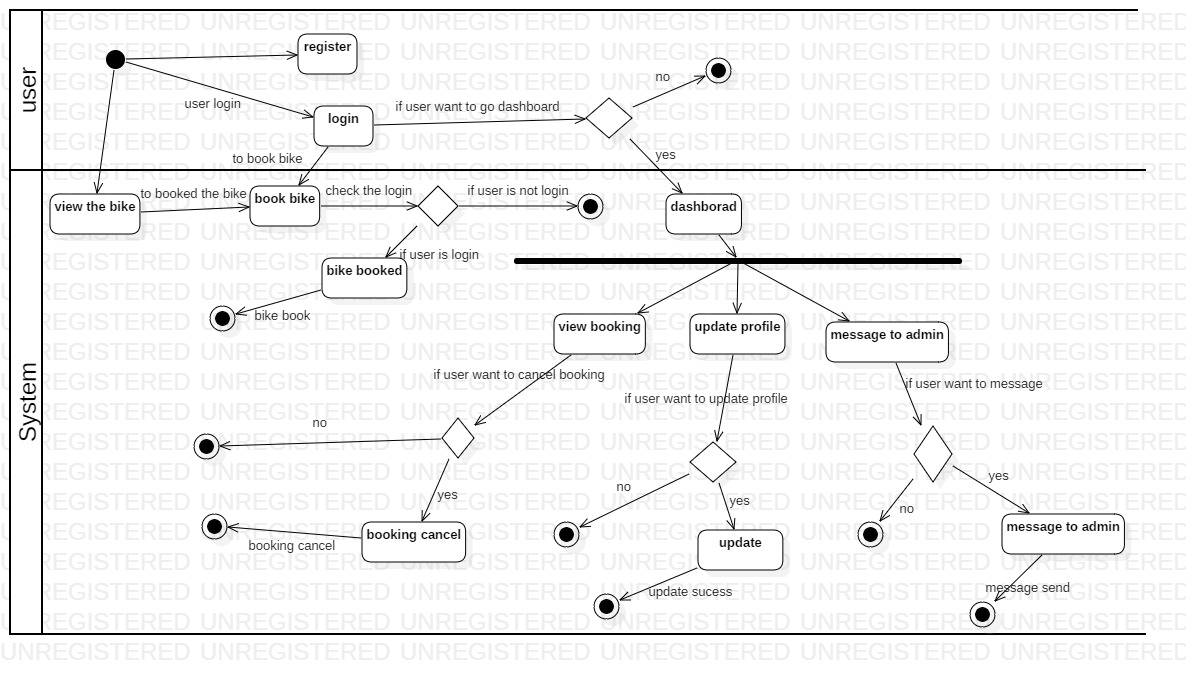
****

Figure 10: user activity diagram

Here are explanation about figure 2

* User view bike information first
* To book the bike user have login and to login user have to register first
* After login user can book the bike
* If user want the she/he can go to dashboard to view the booking
* User also can cancel booking if he/she wants then
* In dashboard user can update and message to admin

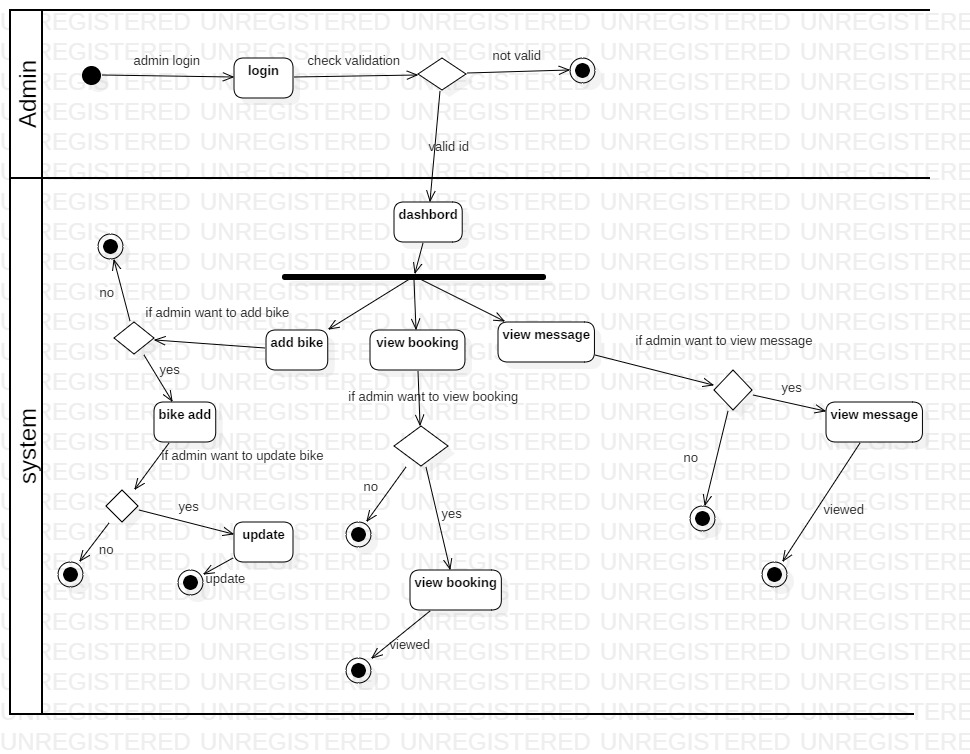


Figure 11: admin activity diagram

In figure 3 show how user interact with system

* Admin have to login to go dashboard
* After login admin can go to dashboard to view booking, view message ,add bike and update bike

## 3.2.2 Sequence diagram

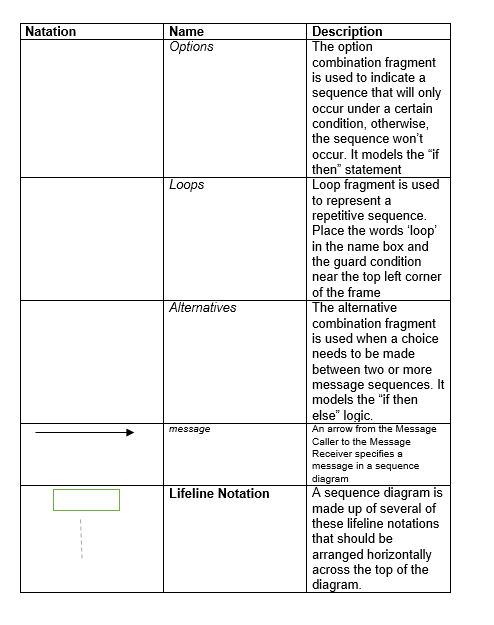
Sequence diagram is an interaction diagram that highlight the time ordering the messages. It also known as timing diagram, event diagram and event scenario. (Anon., 2018)

**Justification for this approach**

Here are some reason why sequence diagram is benefits for my project.

* It helps makes model the logic of a knowledgeable method, function, or operation
* We can know about how objects and components interact with each other to complete a process
* If you consider a service to be a high-level method used by different clients, a sequence diagram is an ideal way to map that out
* Sequence diagram helps to make our system more logically.

**Notational used**

****

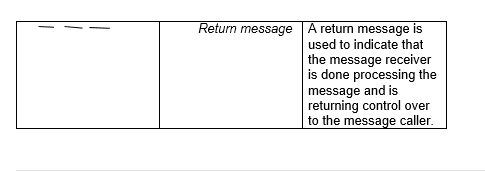
****

Figure 12: notational used

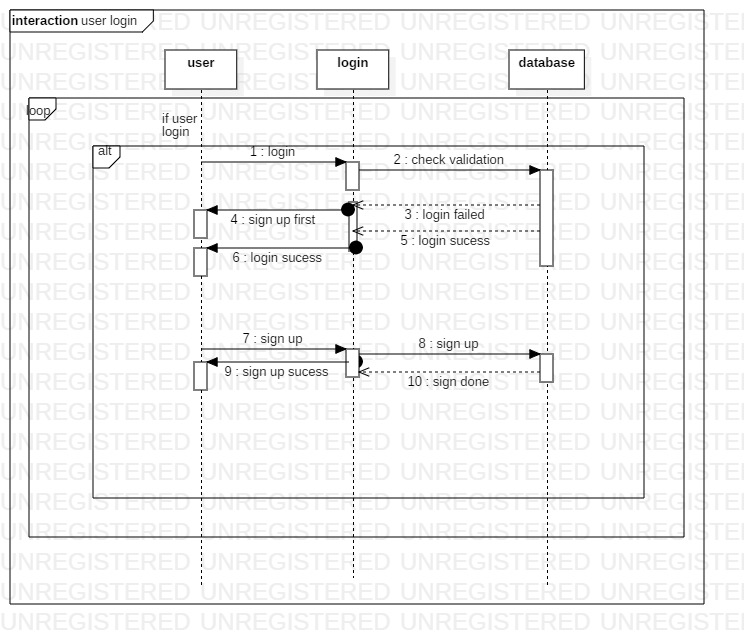


Figure 13: sequence diagram

In this sequence diagram

* User login to system then system check the database
* If login is valid then login success and else not then user have to sign up
* After sign up user can login

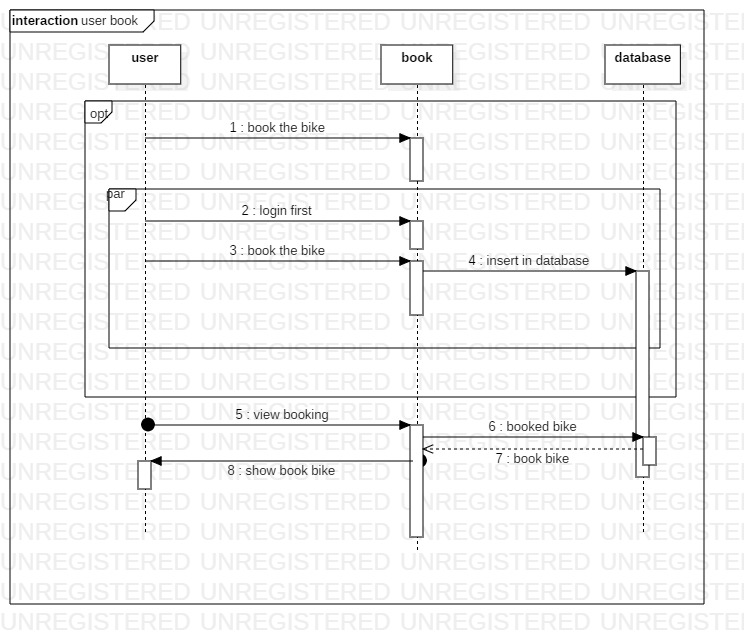


Figure 11: sequence diagram2

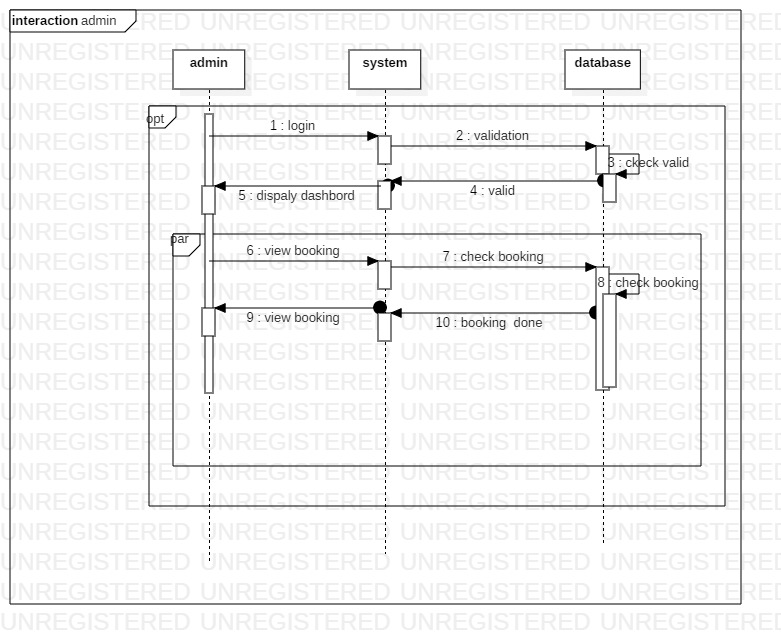


Figure 14: sequence diagram3

Figure 7

* Admin have to login.
* After login admin can view the booking

## 3.3.1 Data Dictionary

Data Dictionary is structured place to keep details of contents of data flow process and data store. (Anon., 2019)

Here are data dictionary of my project

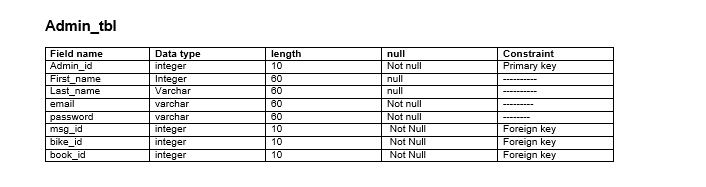


Figure 15: admin

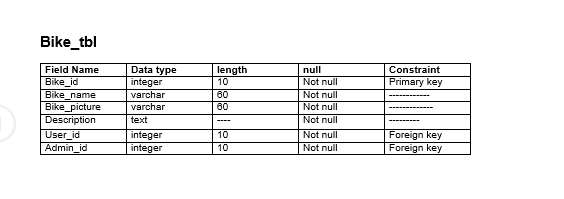


Figure 16: bike table

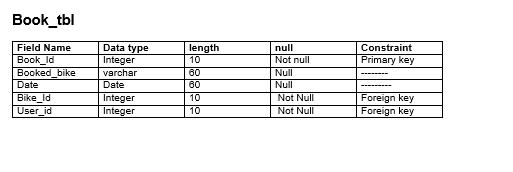


Figure 17: book table

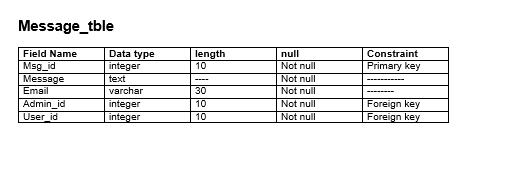


Figure18: message table

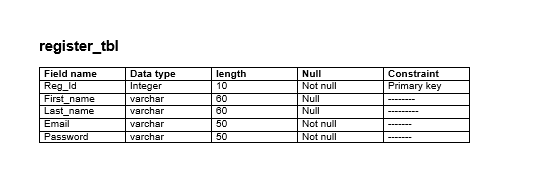


Figure 19: register table

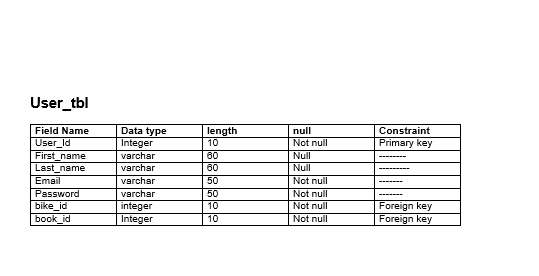


Figure 20: user table

## 3.3.2 ER diagram

Entity Relationship Diagram (ER diagram) is consists of collection of basic object called entities and of relationships among these objects and attributes which define their properties. (Anon., 2018)

**Justification for this approach**

Here are some reason ER diagram is important for my project.

* ER diagram helps to design the database. By looking the diagram we can find the problem and make correction before performing in database
* It help in database debugging by making ER diagram we can easily know the entities, their attributes and relationship with each other. By analyze this we find out the problem easier

**Diagram**

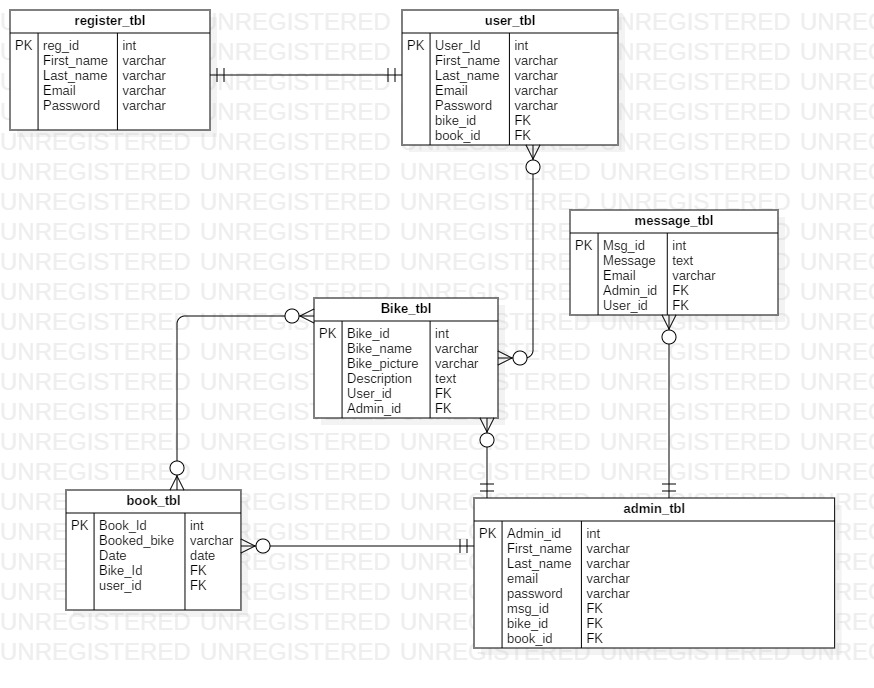
****

Figure 21: ER diagram

**ER Diagram explanation**

* One user can one register
* Register User can book the bike
* Register can message to admin
* Admin view the book form user
* Admin can view the message
* Admin can add information about bike

# 3.4 UI Mode

User interface (UI) is model of the system how it perform which is show to user for the feedback.

**Justification for this approach**

The reason why UI is design

* It show the mode of the system how it work if user does not like it then it will easier to change the model then system.
* It helps to find out what user want to in system.
* We can find out how the system fit in user daily activity.

**Diagram**

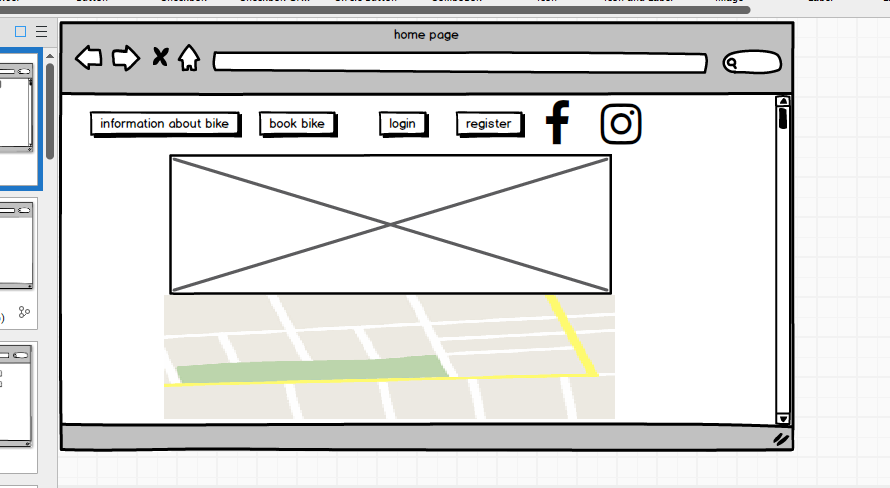
****

Figure 22: homepage

This homepage of bike rental system in here user can view information about bike, book bike and register and location of the organization.

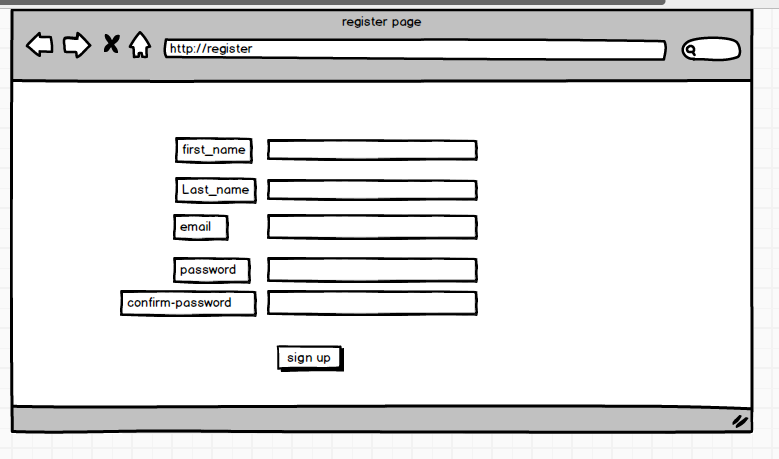


Figure 24: register page

User can sign up then book bike and message the bike

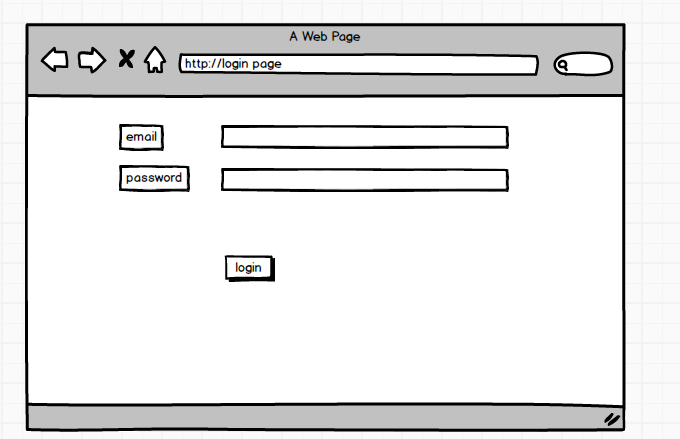


Figure 25: login page for user

This login page to book the bike user have to login

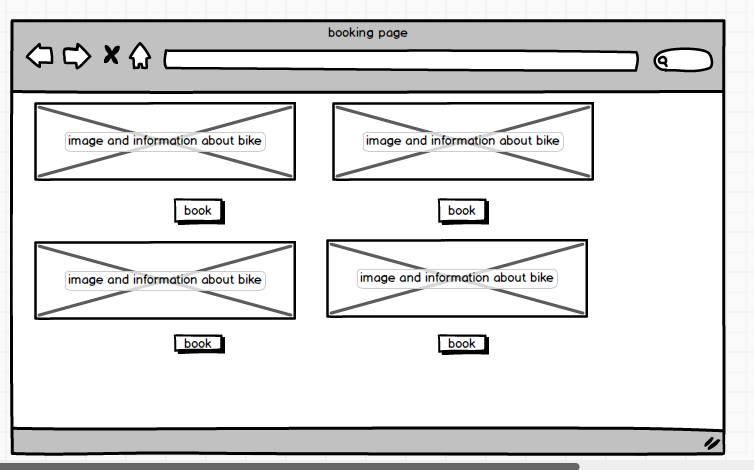


Figure 26: booking page

User can book the bike in this page.

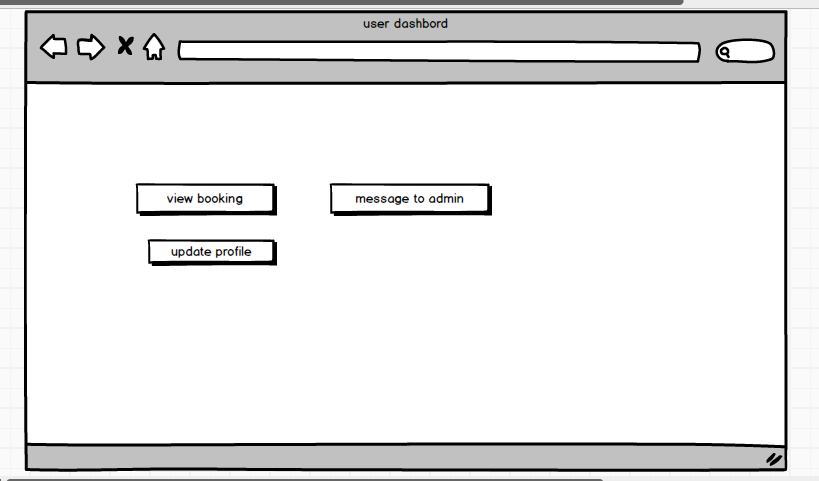


Figure 27: user dashboard

This is user dashboard. User can view the book, message to admin and update profile form this page

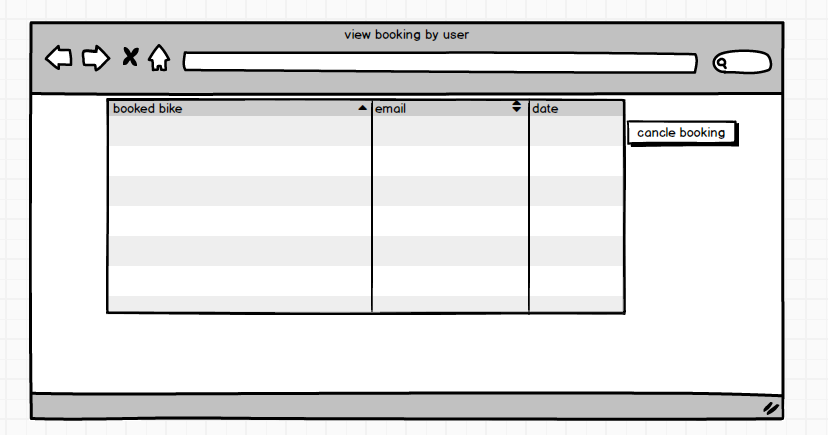


Figure 28: view booking

For user to view the booking. User can cancel the booking if they want

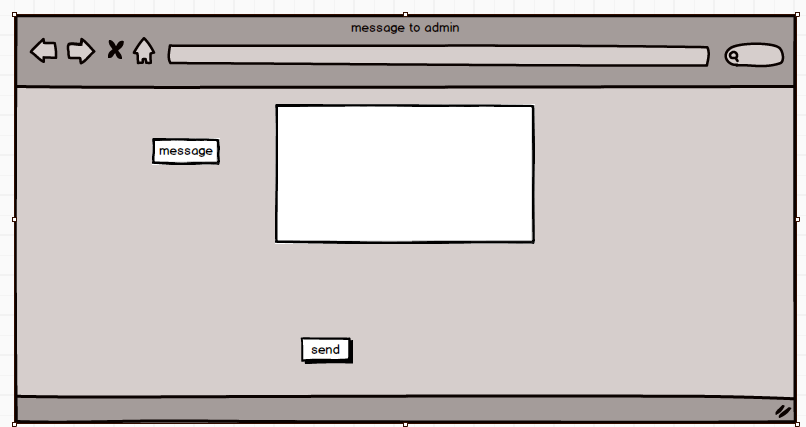


Figure 29: message

User can message the admin

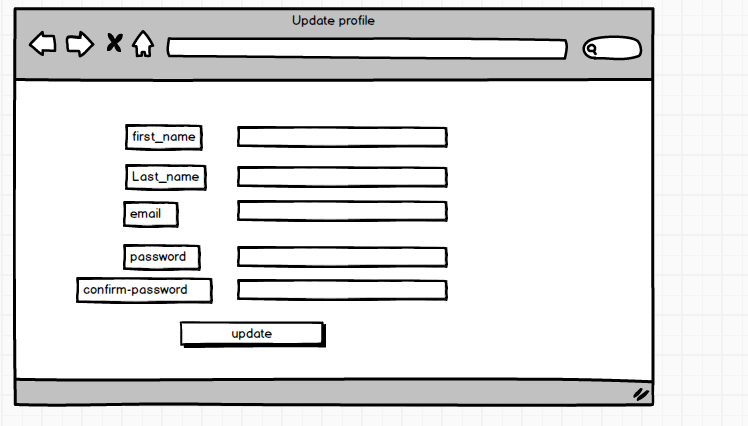
****

Figure 30: update profile

If user want then they can update their profile from this page

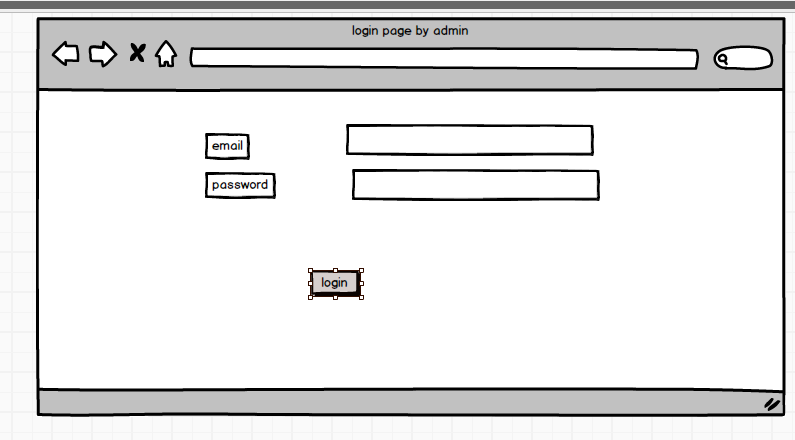
****

Figure 31: login page

This login page is for admin.

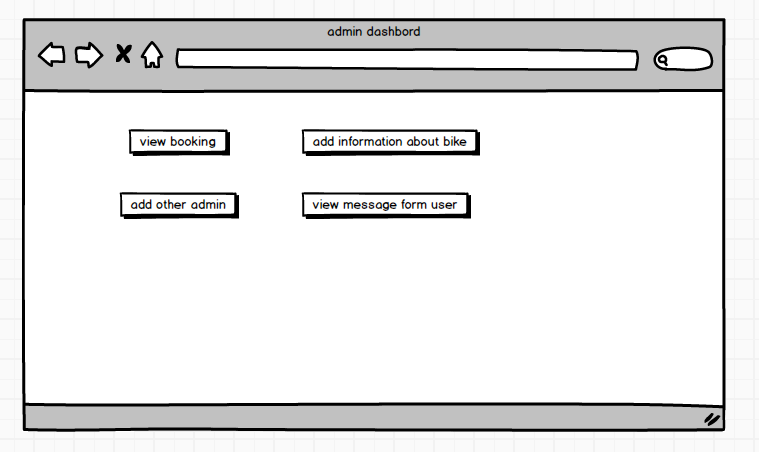


Figure 32: admin dashboard

Admin can add information about the bike, view the booking from the user, view the message form the user and add other admin also.

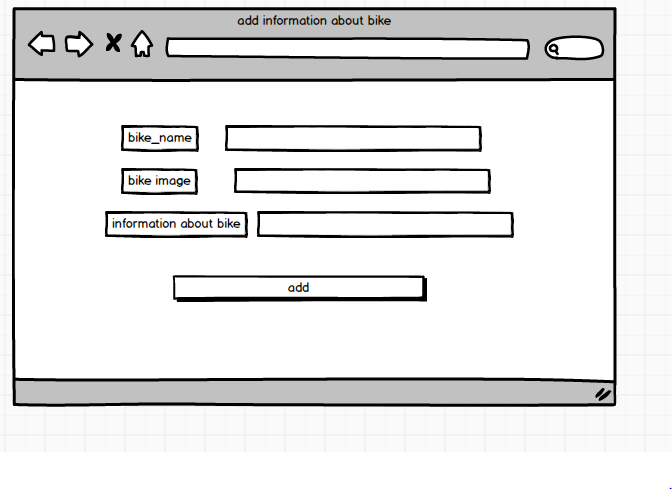


Figure 33: add information

Admin add bike name, image and description about then bike.

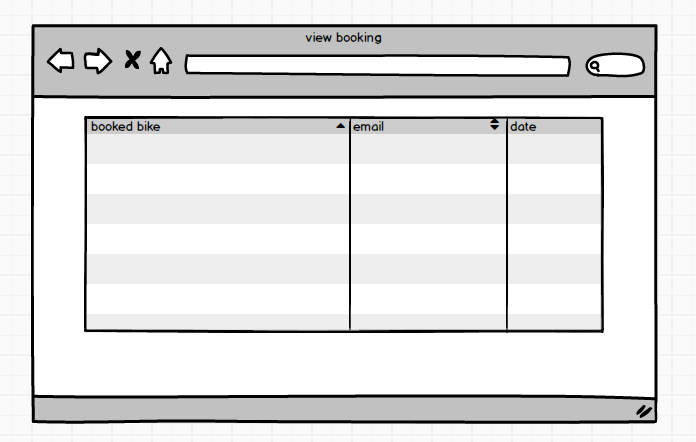


Figure 34: view booking

Admin can view booking from the user

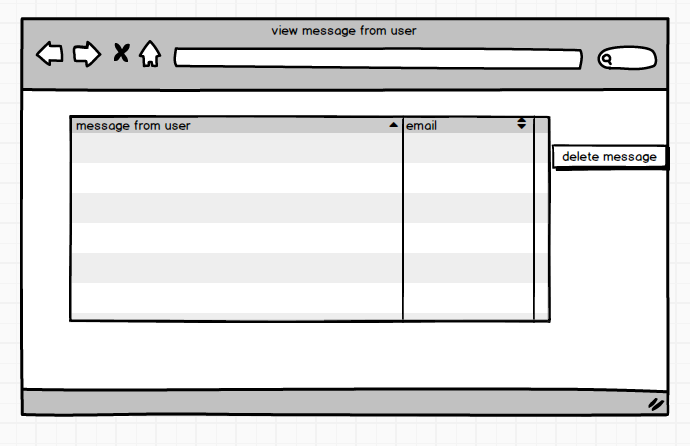


Figure 35: view the message

Messages send by the user view by the user

**Prototype explanation**

So this is prototype of the system. But system will not be exactly like when this prototype can be change if requirement will changes.

# Chapter 5 Testing

Testing is the process of evaluating a system or its component(s) with the intent to find whether it satisfies the specified requirements or not (Anon., 2019). There are different type of testing I have choose black box and unit testing for my project.

## 5.1 Black box testing

Black-box testing is a method of software testing that examines the functionality of an application without peering into its internal structures or workings

Reason why I choose this test

* To do black testing it does not require to write code it is done by end user
* It only show what program does not how program is running
* In here it show either program is pass or fail.

**Login testing**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test case**  **ID** | **Test scenario** | **Test description** | **Test date** | **Expected** | **Actual** | **Status**  **(Fail/pass)** |
| 1 | Login | Valid email and password | 9/28/2019 | User dashboard should be open | User dashboard has been open | Pass |
| 2 | Login | Invalid email and password | 9/28/2019 | Message should be show | Message has been shown invalid password /email | Fail |
| 3 | Login | Incorrect password | 9/28/2019 | Should bock for 3 minute | Block for 3 minute | Pass |

Table4: login testing

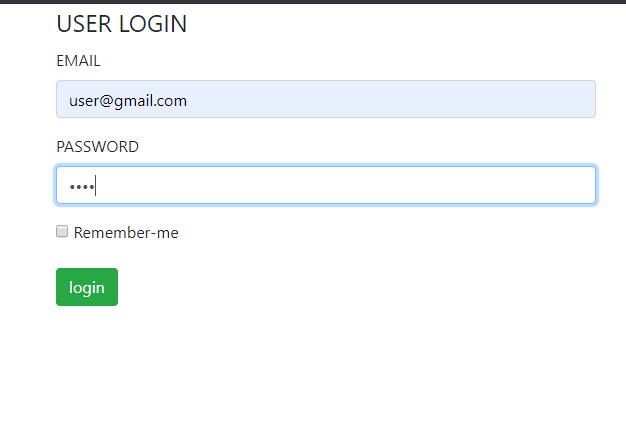


Figure 36: test 1

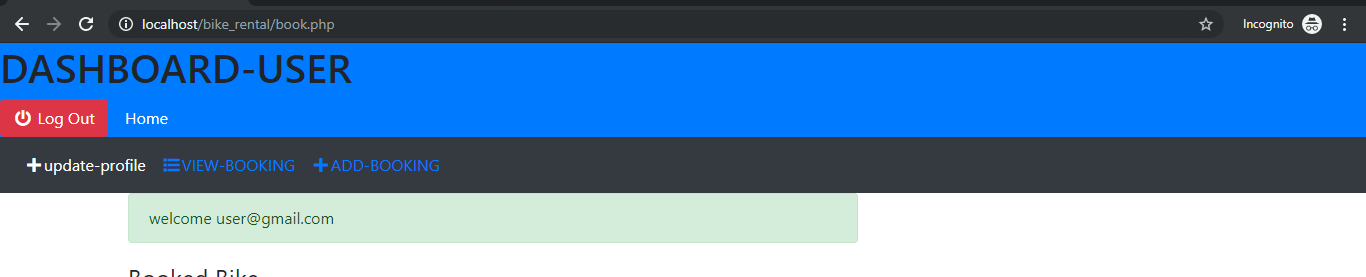


Figure 37: test 1 result

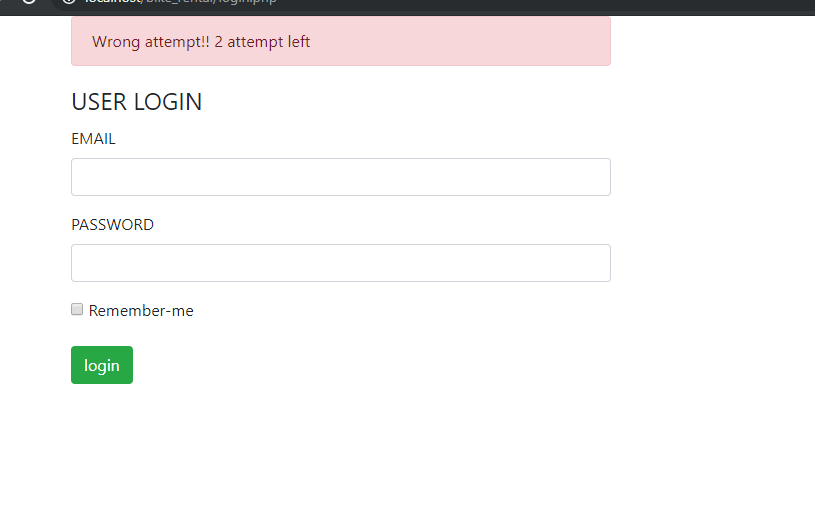


Figure 38: test 2

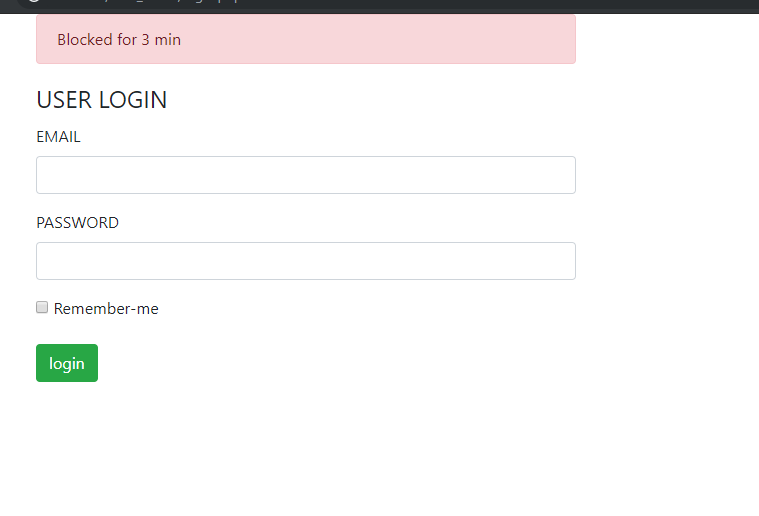


Figure 39: test 4 for invalid password

**Registration testing**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test case**  **ID** | **Test scenario** | **Test description** | **Test date** | **Expected** | **Actual** | **Status**  **(Fail/pass)** |
| 4 | Register for user | User can register | 9/28/2019 | User can register | User is Register | Pass |
| 5 | Register | User can not register same email that is already register | 9/28/2019 | Message should be came | Message came email has already register | Pass |
| 6 | Register | User have to fill all detail to register | 9/28/2019 | Message should be shown | Message is shown | Pass |

Table5: register testing

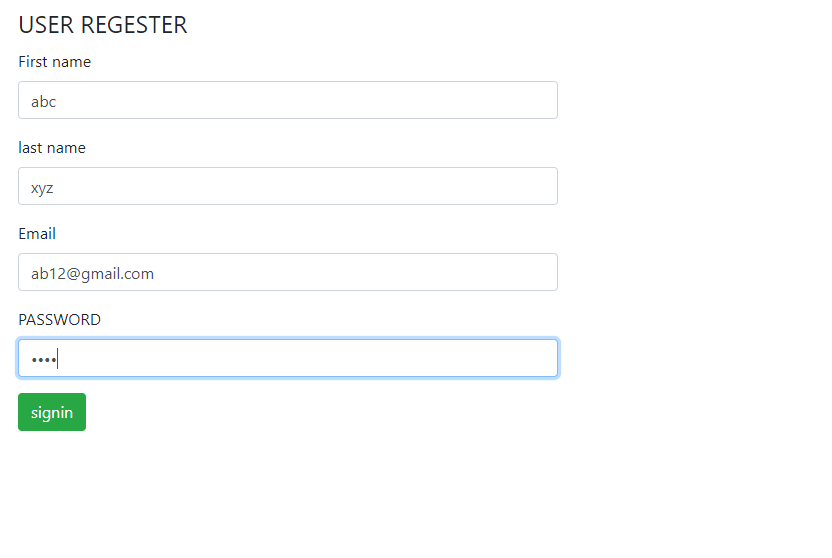


Figure 40: test 4 for registration

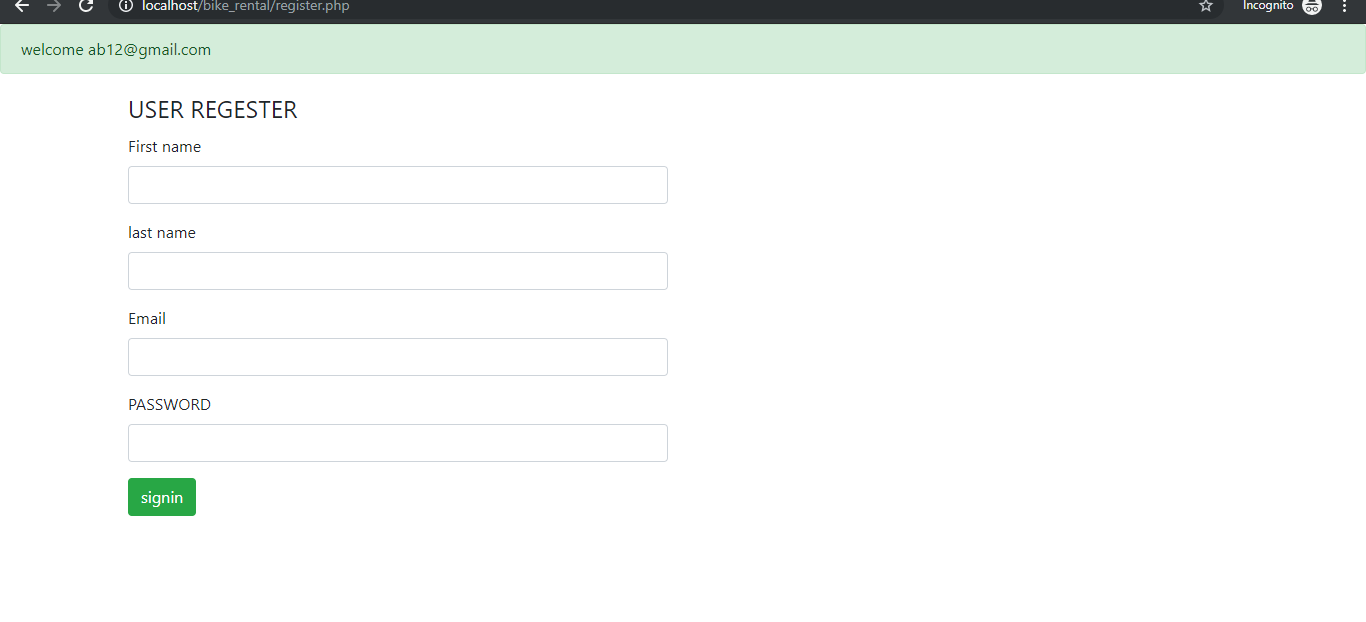


Figure 41: test 4 register success

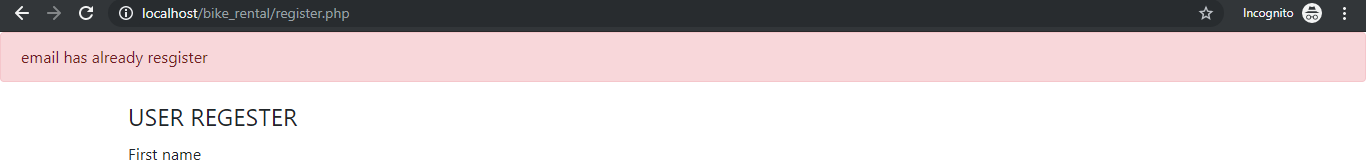


Figure 42: test 5 email already register



Figure 43: test 6 email empty

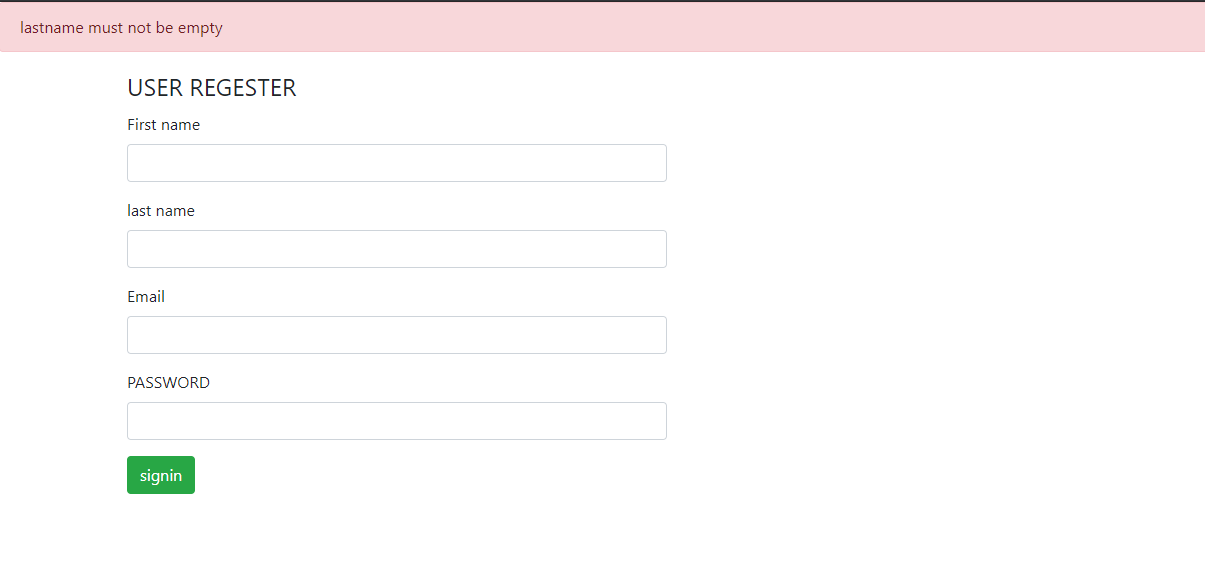


Figure 46: test 6 last name empty

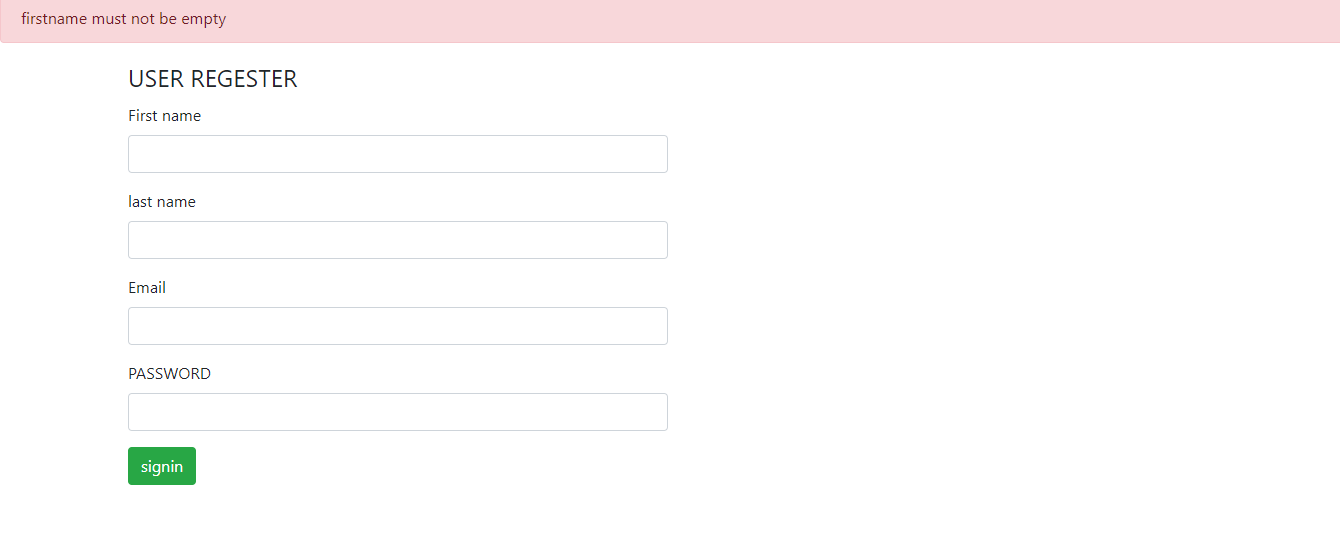


Figure 47: test 6 first name empty

Booking testing

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test case**  **ID** | **Test scenario** | **Test description** | **Test date** | **Expected** | **Actual** | **Status**  **(Fail/pass)** |
| 7 | User can book the bike | To book bike user have fill the form | 9/28/2019 | Booking should be done | Booking has been done | Pass |
| 8 | View booking | User can view the booking | 9/28/2019 | User should view booking | User can view booking | Pass |
| 9 | Cancel booking | User can cancel booking | 9/28/2019 | User should able to cancel booking | User can cancel booking | Pass |

Table6: booking testing

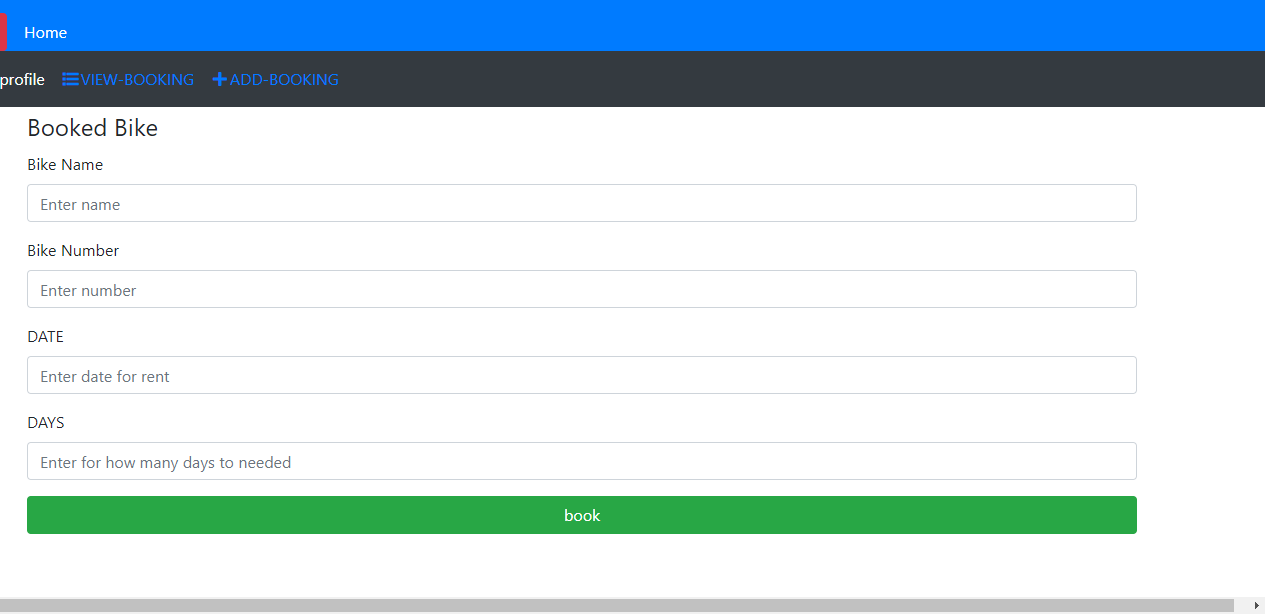


Figure 48: test 7 to book the bike

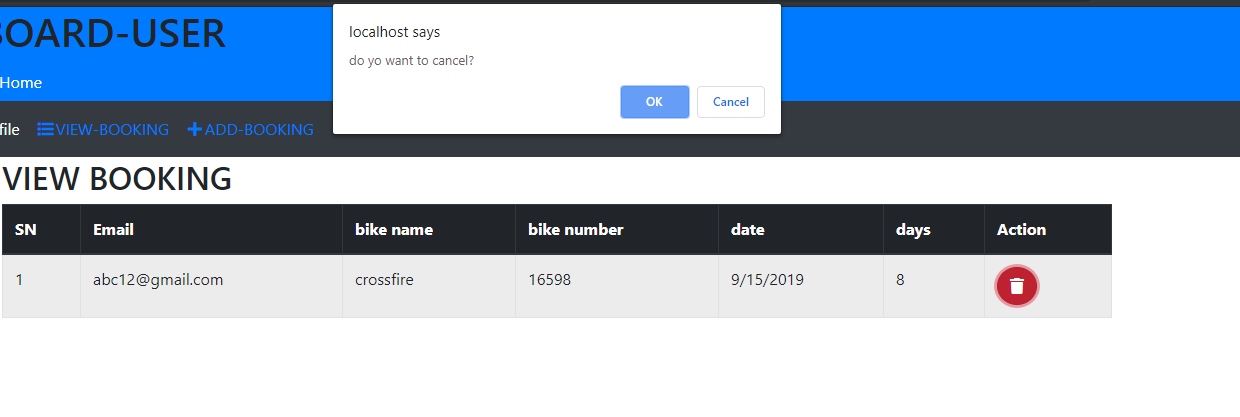


Figure 49: test 8, 9 view booking and for cancel booking

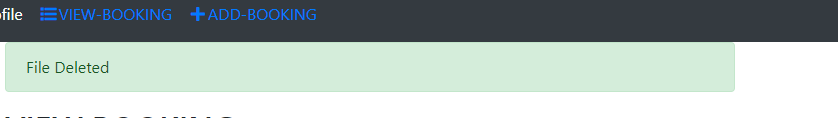


Figure 50: test 9 after booking cancel successfully

**Admin testing**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test case**  **ID** | **Test scenario** | **Test description** | **Test date** | **Expected** | **Actual** | **Status**  **(Fail/pass)** |
| 10 | Add bike information | Admin can add bike information | 9/28/2019 | Admin able to bike information | Admin can add bike information | Pass |
| 11 | View bike information | Admin can view information | 9/28/2019 | Admin should view the information about bike | Admin can view the information of bike | Pass |
| 12 | Update bike | Admin can update bike information | 9/28/2019 | Admin able to update bike information | Admin can update bike information | Pass |
| 13 | Delete information | Admin can delete information | 9/28/2019 | Admin able to delete information of bike | Admin able to delete bike information | Pass |
| 14 | Booking | Admin can view book | 9/28/2019 | Admin should able to view booking done by the user | Admin can view booking by the user | Pass |
| 15 | View bike information | User can view information about bike | 9/28/2019 | User should view bike information add by the admin | User can view the information about bike | Pass |

Table7: admin testing

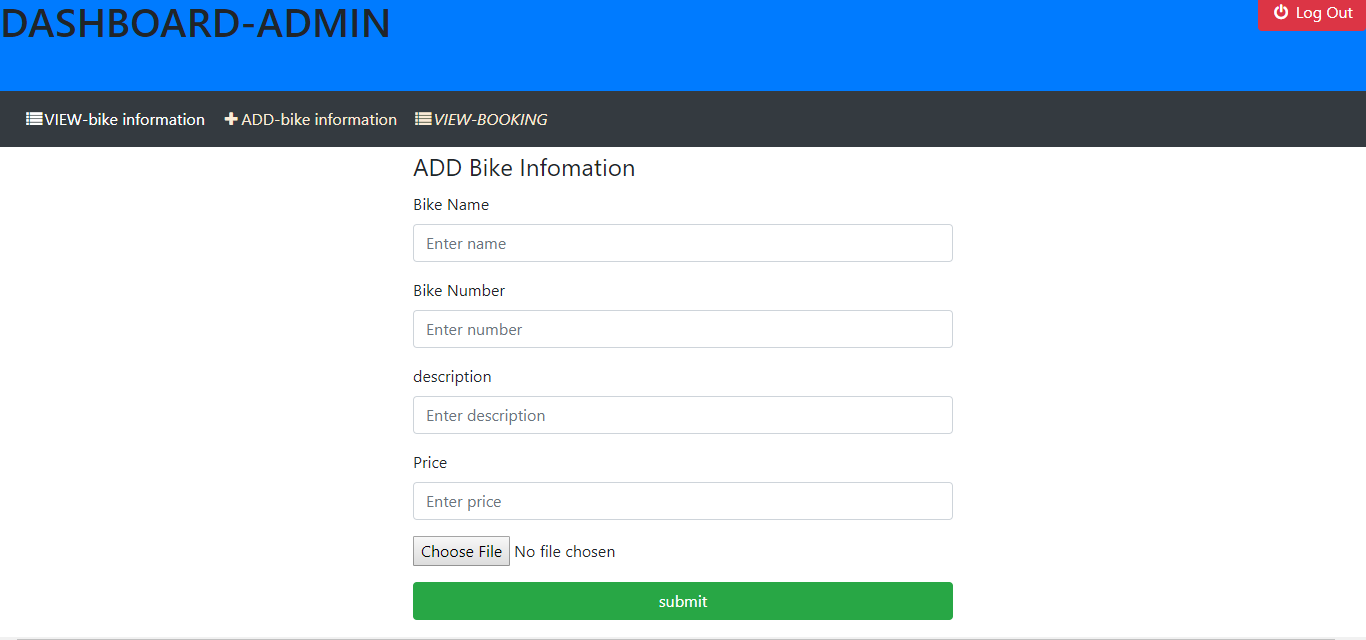


Figure51: test 10 to add bike information

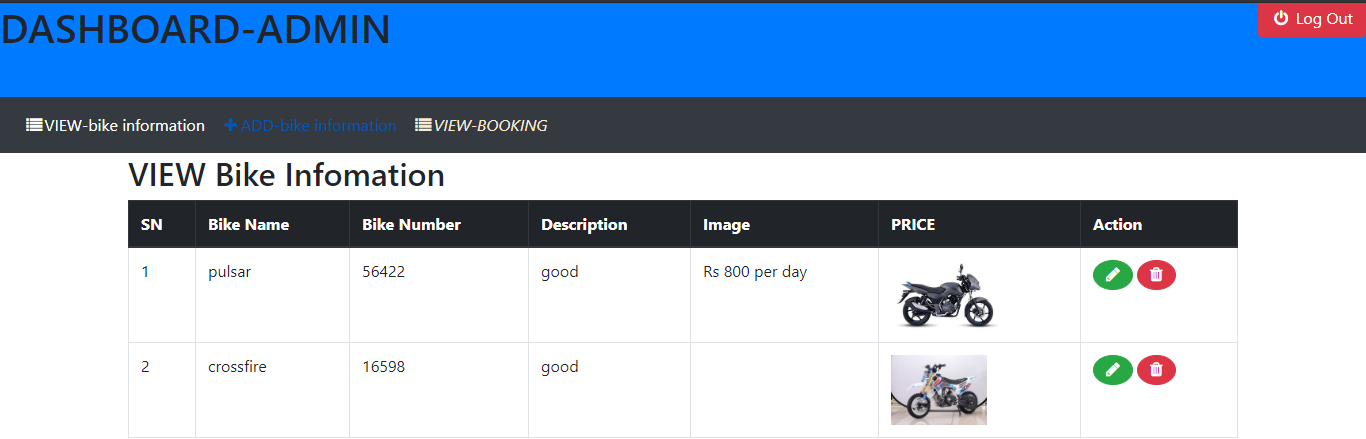


Figure52: test 11 to view bike information

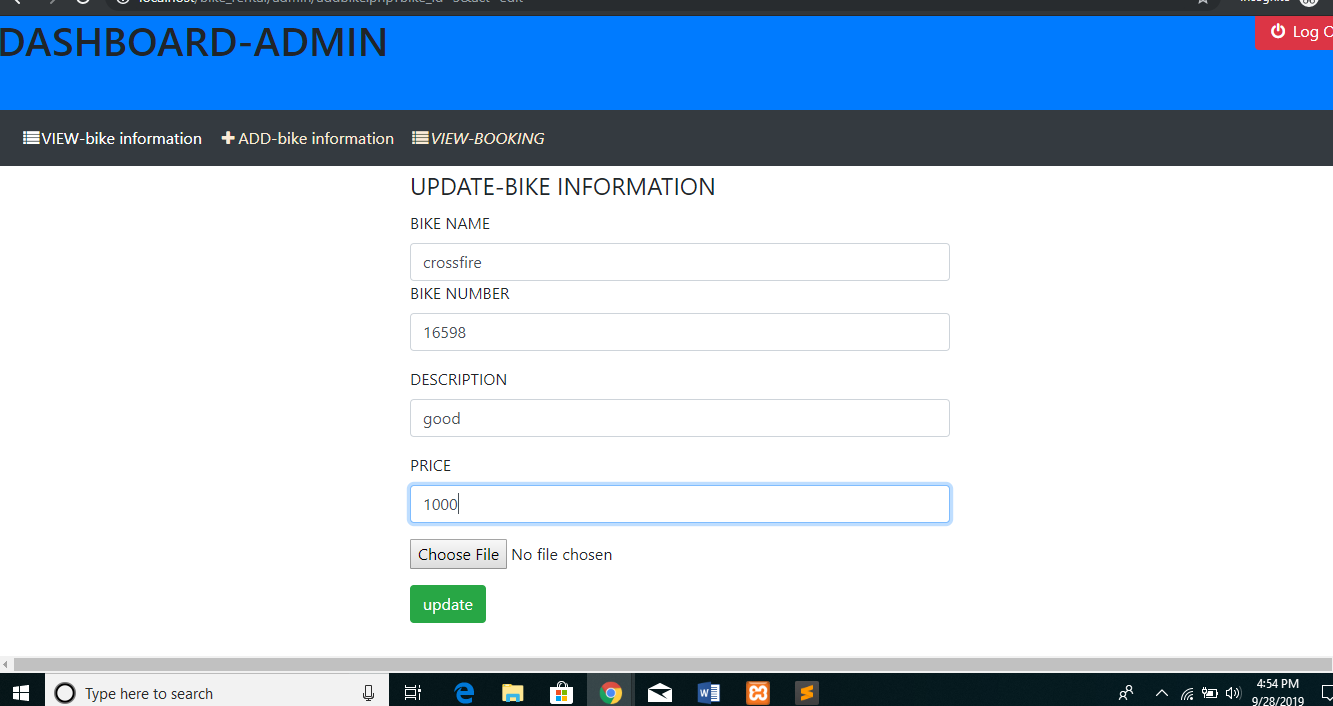


Figure 53: test 12 to update bike

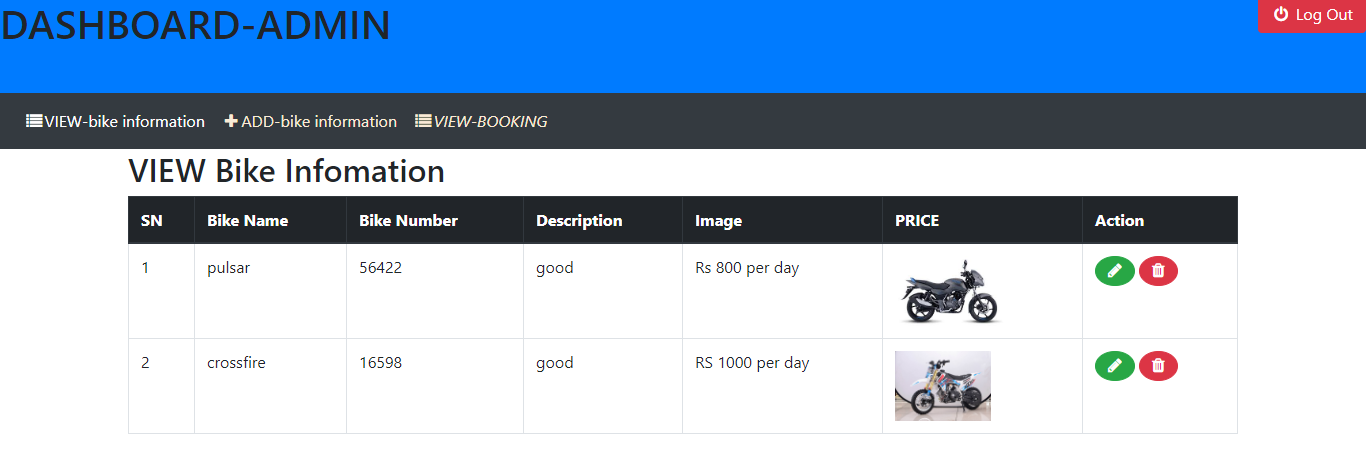


Figure 54: test 12 update success

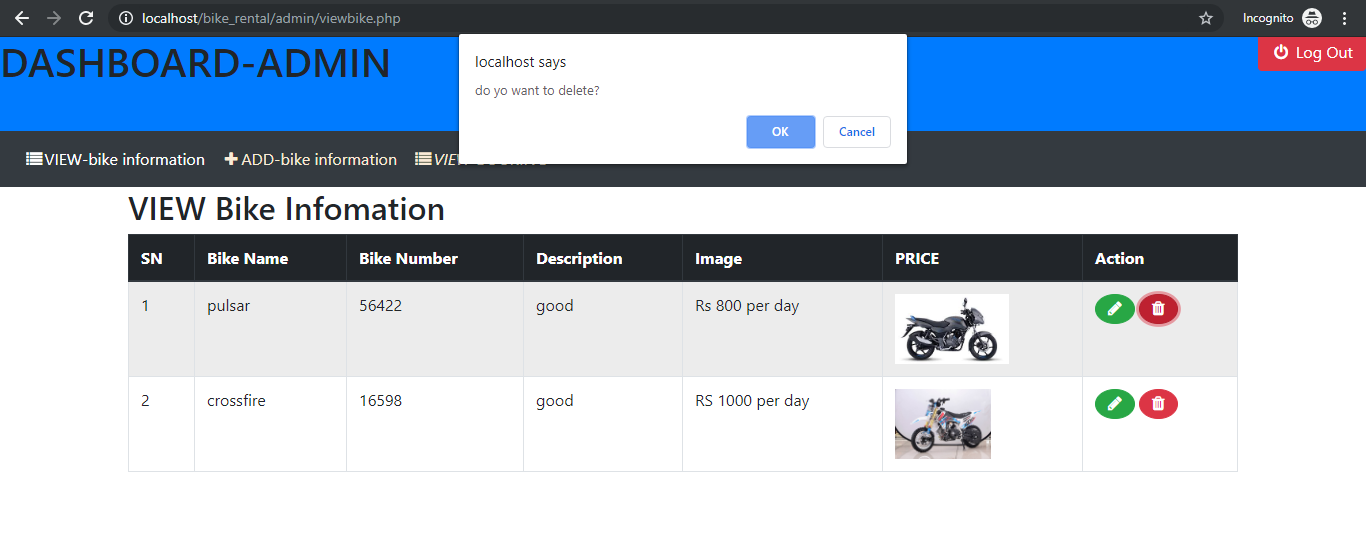


Figure 55: test 13 to delete bike information

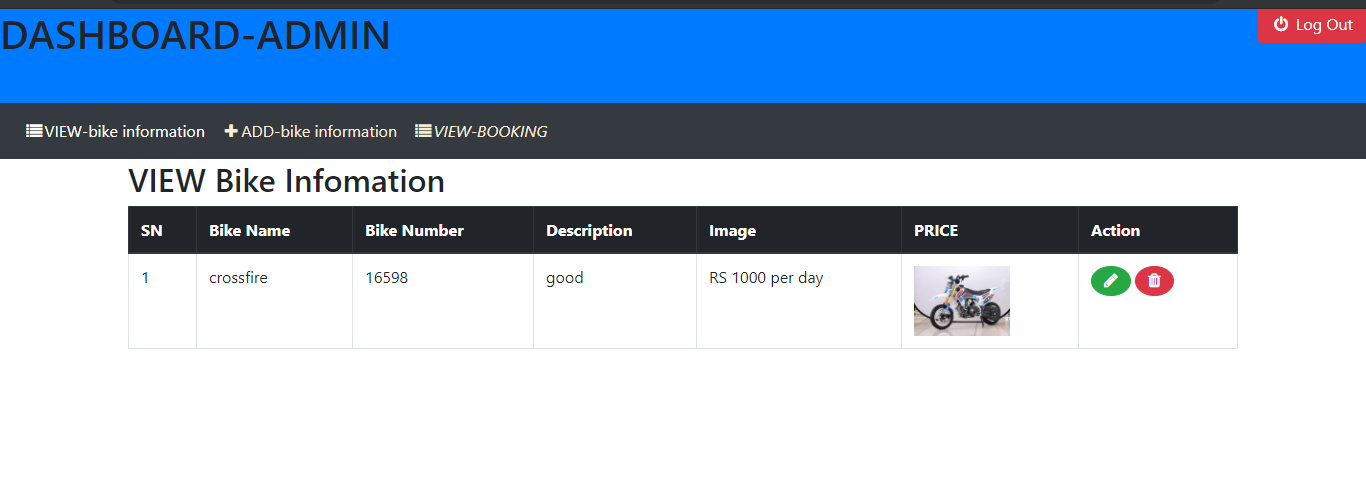


Figure 56: test 13 bike delete

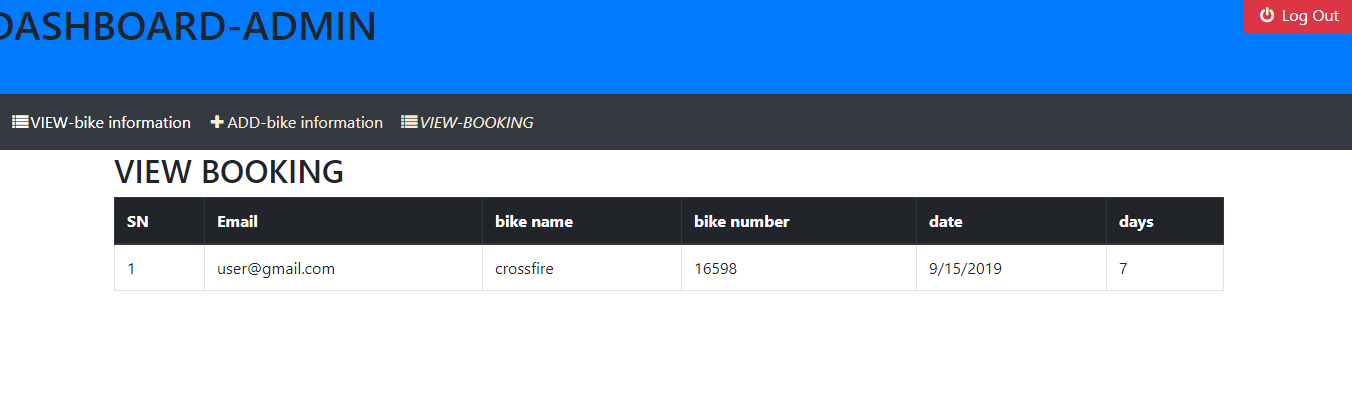


Figure 57: test 14 to view booking bike by user

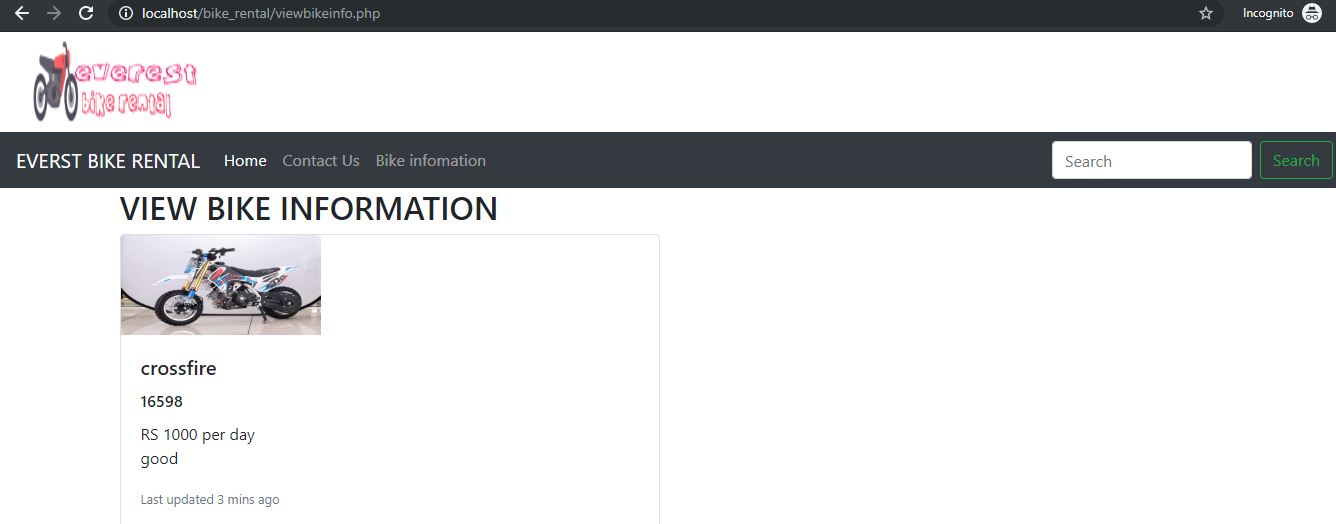


Figure 58: test 15 user can view bike .information

**Admin login**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test case**  **ID** | **Test scenario** | **Test description** | **Test date** | **Expected** | **Actual** | **Status**  **(Fail/pass)** |
| 16 | Login | Admin login | 9/28/2019 | After login success dashboard should be open | Login success dashboard open | Pass |
| 17 | Login | Username and password | 9/28/2019 | To login username and password should be valid | Invalid username and password message pop up | Pass |

Table8: admin login testing

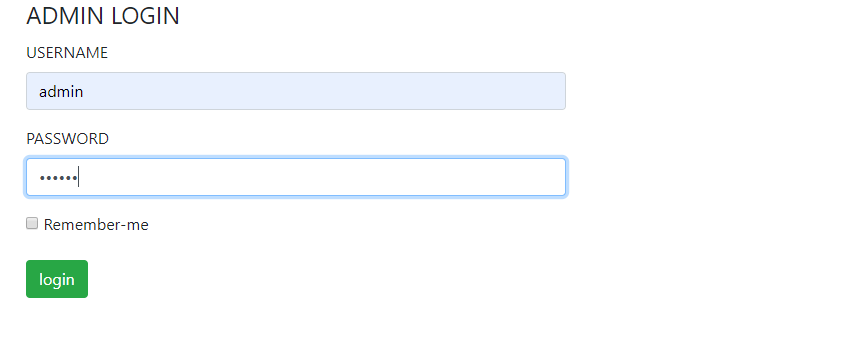
****

Figure 59: test 16 login by admin

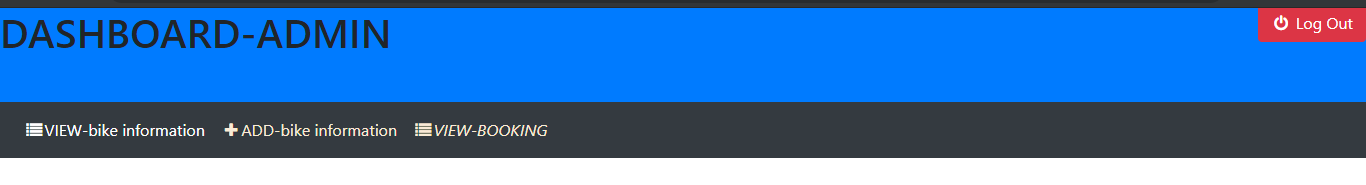
****

Figure 60: test 16 login success

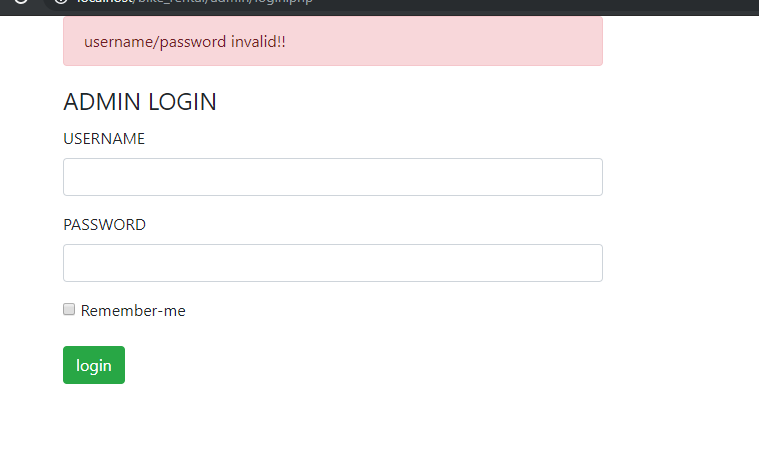
****

Figure 61: test 17 invalid username and password

# 6 Other project issues

## 6.1 Limitation

This system also have some limitation. The limitation are

* User can’t update their profile
* They can message to admin directly
* The system does not show the which bike is mostly booked
* Only one admin can handle back end of the system

## 6.2 Future Work

The future work for the system are

* User can update their profile
* Add new admin
* User can delete account if they want
* We can make online payment system
* User can send message to admin

## 6.3 Risk management

While developing the system risk I have list before in proposal I have face is lack of time. Because of lack of time I could not make user can message admin and admin can replay. I have manage is risk by allow all user to give feedback about bike or system.

## 6.4 Configuration management

For the backup I have use github. My account is **“binaythapa01”** in this account I have made folder CP.

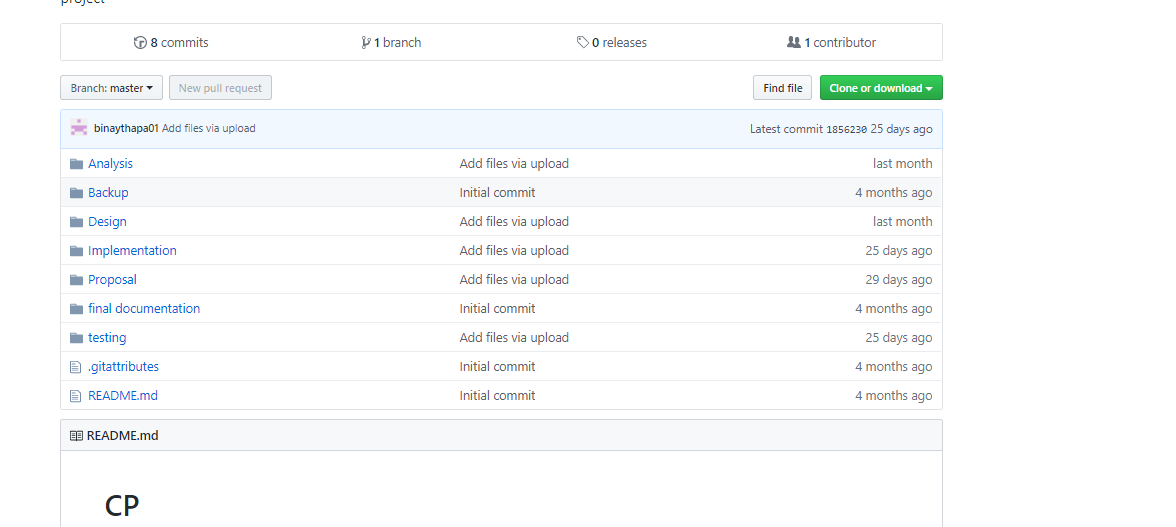


Figure 62: github file

In local computer I have create file. The direction of file

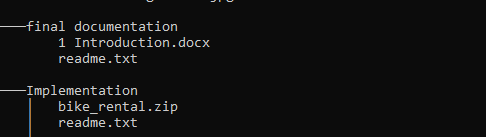
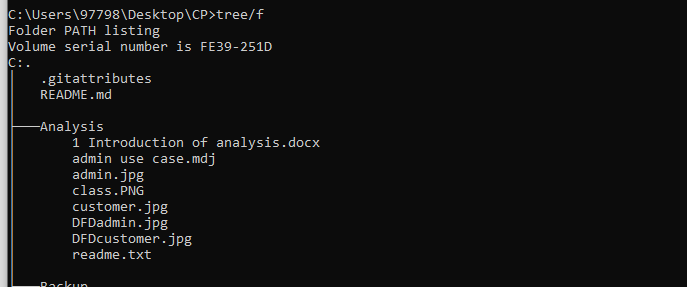


Figure 63: project local directory

## 6.5 Other project issue

To complete this project we does not face issue like

**Social:** This is collage project the it does not create any project issue

**Economic:** Because collage project I don’t face any problem by economic

**Technical:** All software that are used to develop this project are open source.

## 6.6 User manual

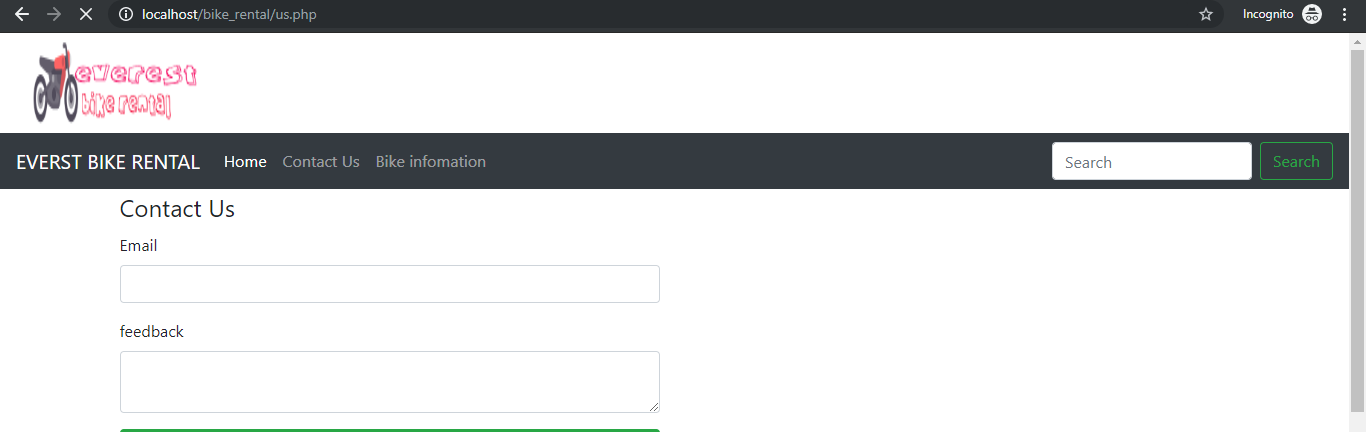


Figure 64: to feedback

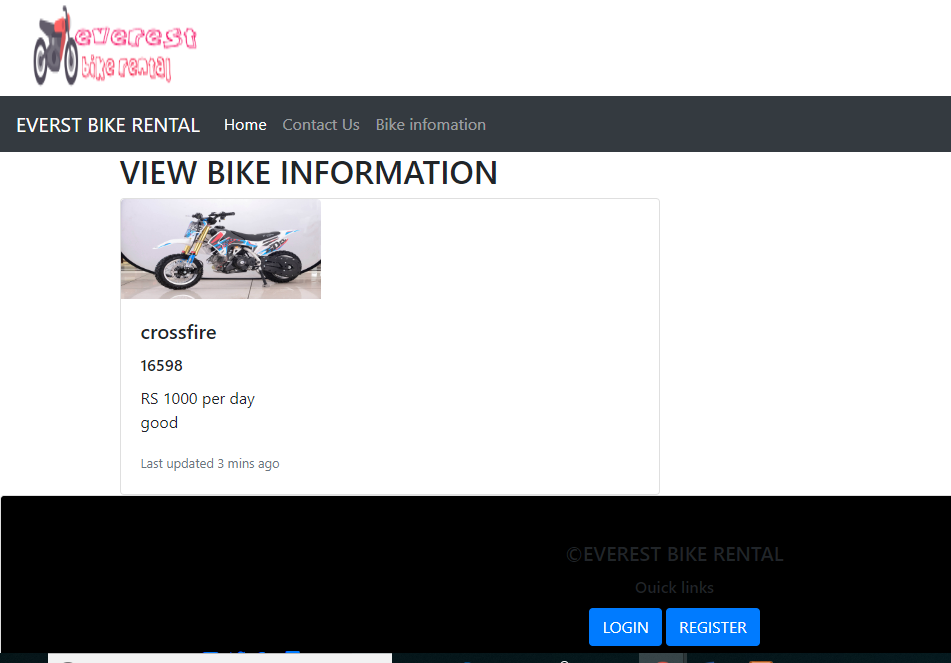


Figure 65: bike infomation

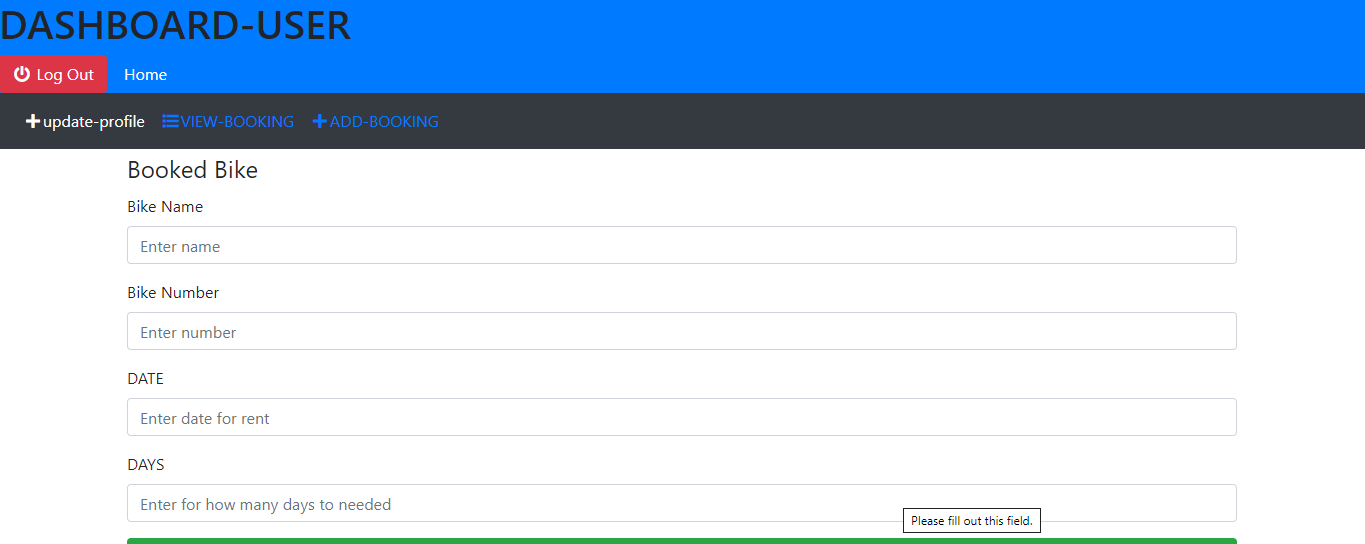


Figure 66: user dashboard

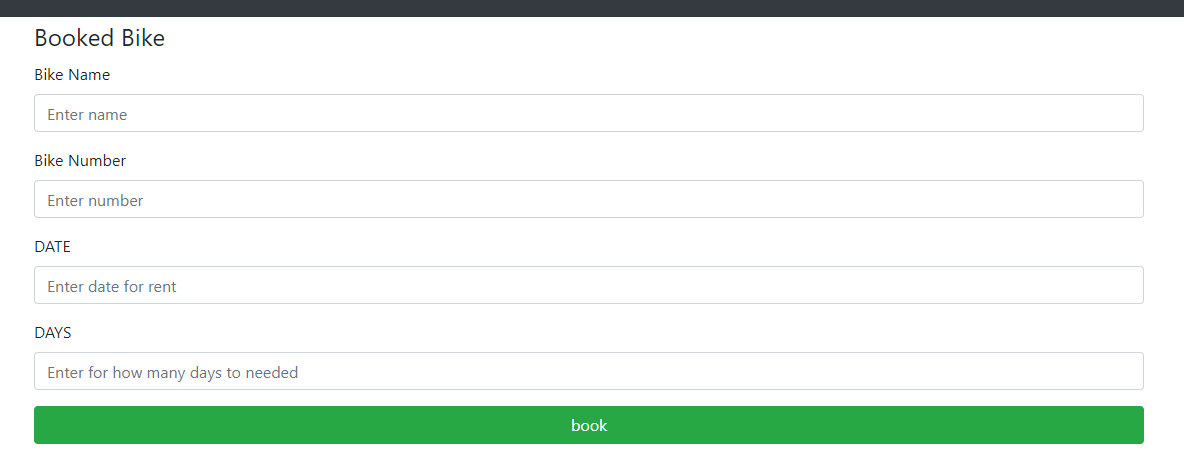


Figure67: booking form

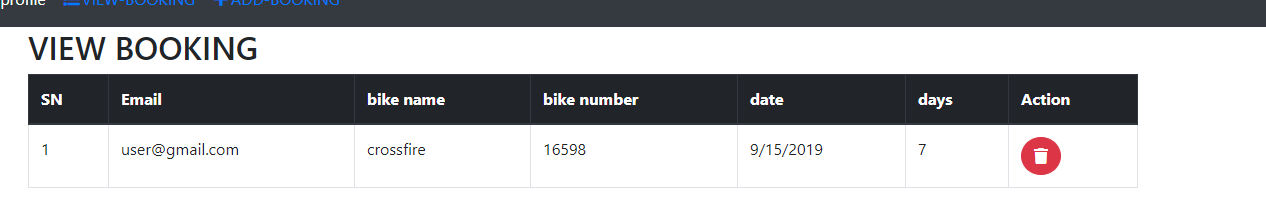


Figure 68: view booking

# 7 Conclusion

Overall, my project name is bike rental system. Its main objectives is customer can book bike. I used waterfall methodology to complete the project. I did analysis part. In analysis I gather all requirement of the project and made the use case diagram to show how user will interact with the system. After analysis is complete then I did design. In design it include behavioral, structural and database design and prototype of the system. Third part is codding after codding testing to check system has any bugs or not. By this way I complete my project. All work I have done is have back file in github.

# References

Anon., 2009. *class diagram.* [Online]   
Available at: https://en.wikipedia.org › wiki › Class\_diagram  
[Accessed february 2009].

Anon., 2009. *flow chart.* [Online]   
Available at: https://en.wikipedia.org › wiki › Flowchart  
[Accessed 12 july 2009].

Anon., 2011. *data flow.* [Online]   
Available at: https://www.smartdraw.com/data-flow-diagram/  
[Accessed 2011].

Anon., 2015. *unf.edu.* [Online]   
Available at: https://www.unf.edu › cen4010 › Chap 5 - System Modeling  
[Accessed 2015].

Anon., 2015. *wrik.* [Online]   
Available at: https://www.wrike.com › project design  
[Accessed 2015].

Anon., 2018. *geeksforgeeks.* [Online]   
Available at: https://www.geeksforgeeks.org › unified-modeling-language-uml-sequence-diagram  
[Accessed 27 aug 2018].

Anon., 2018. *smartdraw.* [Online]   
Available at: https://www.smartdraw.com › entity-relationship-diagram  
[Accessed june 2018].

Anon., 2019. *software testing.* [Online]   
Available at: https://www.guru99.com › software-testing-introduction-importance  
[Accessed 18 sep 2019].

Anon., 2019. *techopedia.* [Online]   
Available at: https://www.techopedia.com › definition › data-dictionary  
[Accessed 2019].

## 

## 

# Appendix

**Admin function**



Figure 69: to add bike information

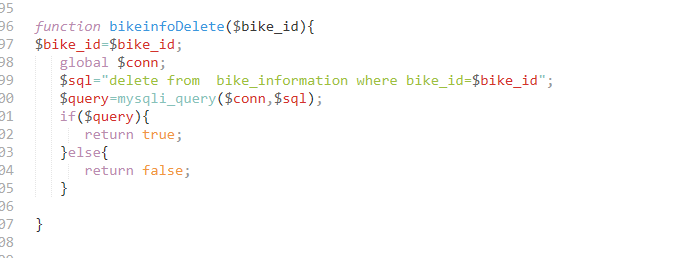


Figure 70: to delete bike information



Figure 71: update bike



Figure 72: view feedback

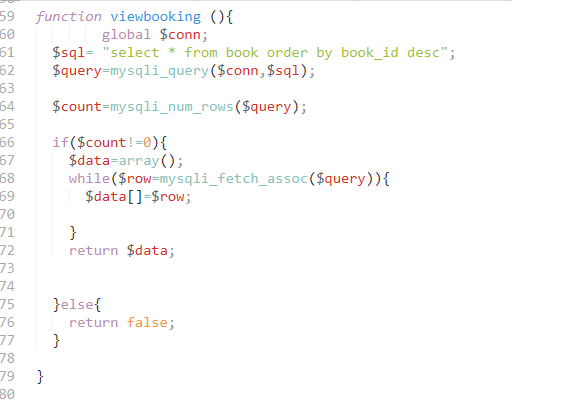


Figure 73: view booking

**User codding**



Figure74 : booking user



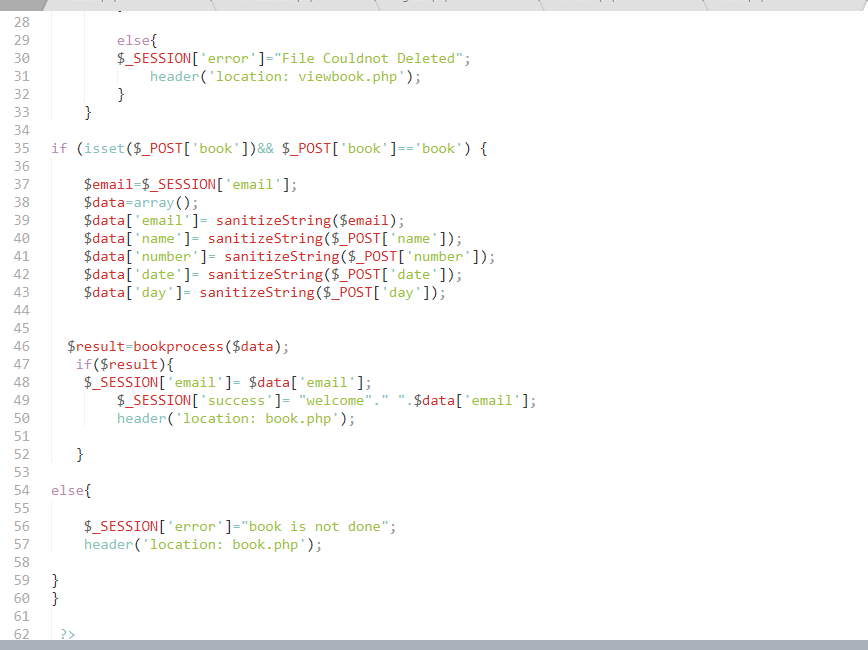


Figure 75: booking process

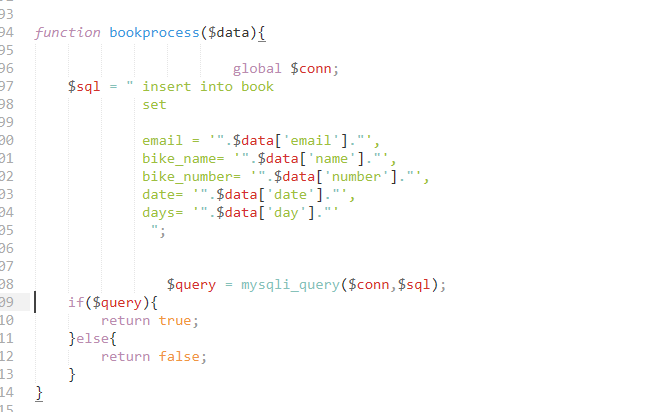


Figure 76: booking function

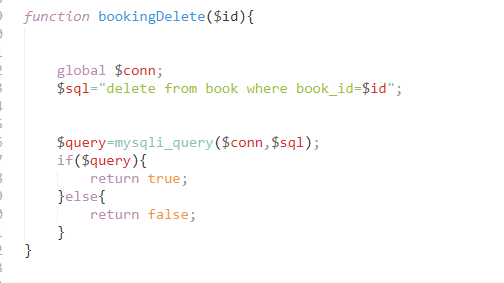


Figure 77: booking delete





Figure78: register page code



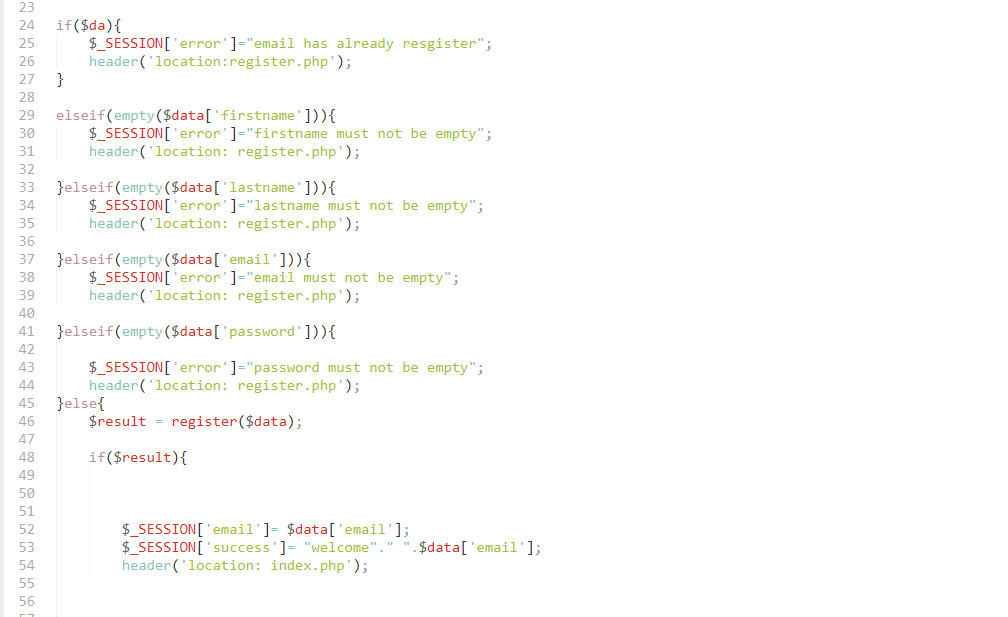




Figure79: register process



Figure80: function register

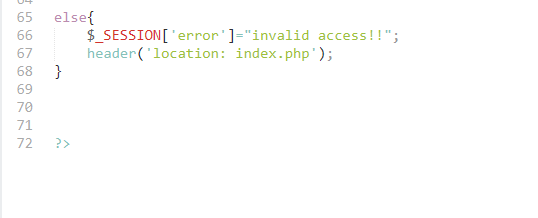
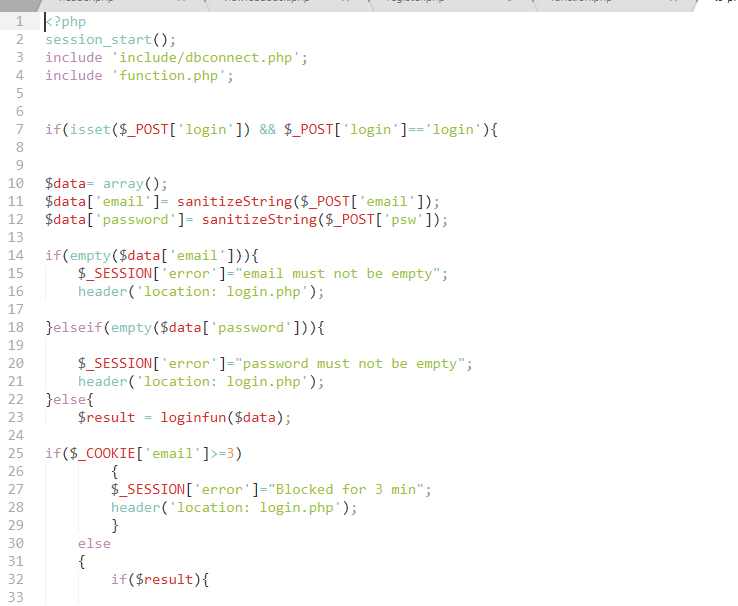


Figure 1: login process

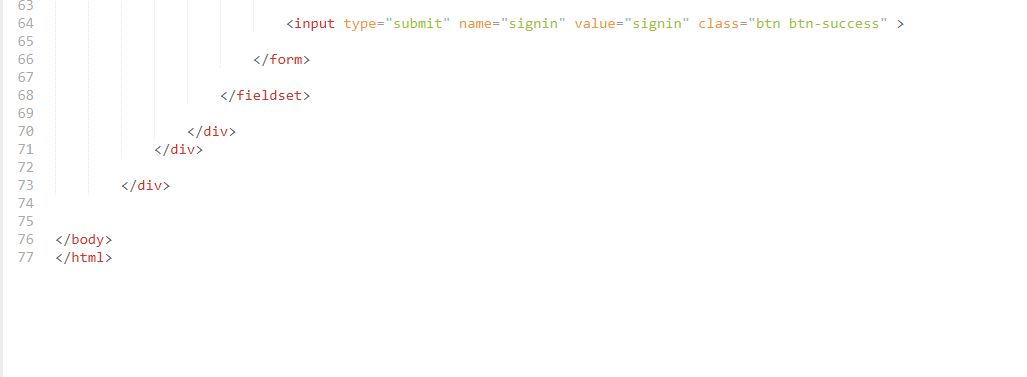


Figure82: login page code



Figure 83: login function



Figure 84: view bike information code user