CAR RENTING SYSTEM

Software Engineering Project Report

Group-16

Report submitted July, 2022

A project submitted to Dr. Rudra Pratap Deb Nath, Associate Professor, Department of Computer Science and Engineering, Chittagong University (CU) in partial fulfillment of the requirements for the Software Engineering Lab course. The project is not submitted to any other organization at the same time.

Table 1: Details of Group-16

Roll Id	Name	Sigature	Date	Supervisor Approval
19701074	Md Abu Noman sikdar	Noman		
19701075	Md.Siam	Siam		
19701044	Md.Jahangir Alam Jehad	Jehad		

Abstract

This report will be focusing on car renting system. It will discuss the causes of the problem and then suggest a software solution. The report will first discuss the problem with a focus on the causes and possible solutions to these. This general view will then be used to create a problem statement for the software solution. Afterwards the architecture and interaction design of the solution will be formulated. The interaction design defines a design language which is a standard of the solutions interaction design. The architectural and interaction design are then used as a basis for the final solution. The product of the report is a software system. The last part of the report concludes and sets the future improvements for the program.

1 Preface

This report is written by Abu Noman Shawn, Md.Siam and Jahangir Alam Zihad. The intellectual property rights of this report and all of its contents belong to the members of the project. All material gathered from third party resources will be referred to according to the IEEE standard. The source code for the software solution of the project, is open source. The code is freely available at GitHub. The colors for this program, were chosen by personal preference. The figures used in the report, have been created using online tool call visual paradigm, unless stated otherwise. The authors would like to thank participants of the usability test for their time and constructive critique, as well as the informants for the interviews and user stories, for their cooperation. The report makers would also like to thank Rudra Pratap Deb Nath for his help and constructive criticism throughout the project process.

Contents

1 Preface **3**

2 Introduction **7**
2.1 Project Description 7
2.2 Problem Statement 8
2.3 Aim and Objectives 8

3 Requirement Gathering and Analysis **9**
3.1 Requirements Gathering 9
3.1.1 Interviews: 9
3.1.2 Document Analysis: 10
3.2 Types of Requirements 11
3.2.1 User Requirements: 11
3.2.2 System Requirements: 11
3.2.3 Functional Requirements: 11
3.2.4 Non-Functional Requirements: 13

4 Software Process Model **14**
4.1 Suitable Models 14
4.2 Not Suitable Models for Our System 16

5 System Choice **17**
5.1 Appreciate the situation: 17
5.1.1 Rich Picture 17
5.2 Cultivate new Ideas 19
5.3 FACTOR 19
5.4 System Definition 20

6 Problem Domain **21**
6.1 Classes 22
6.1.1 Physical 22
6.1.2 Person 22
6.1.3 Places 22
6.2 Class Diagram 23
6.3 Events 24
6.3.1 Event Definition: 24
6.3.2 Event Table 26
6.4 Behavior 27

7 Application Domain **28**
7.1 Usage . . . 28
7.1.1 Actors . . . 28
7.1.2 Use Case Diagram . . . 28
7.2 Functions and Complexity . . . 29

8 System Architecture Design **30**
8.1 Priority of Criteria . . . 30
8.2 Architechtural Design . . . 31
8.2.1 Architechture Component . . . 31
8.2.2 Model Component . . . 31

9 User Interface Design **32**

10 Implementation **33**
10.1 Code Snippet . . . 33

11 Testing **34**

12 Software Deployment **35**

13 Conclusion **36**

14 Bibliography **37**

List of Figures

1 The Waterfall Model . . . 15
2 The Iterative Model . . . 15
3 Rich Picture of Car Renting System . . . 17
4 Rich Picture of Car Renting System . . . 18
5 Class Diagram of Car Renting System . . . 23

List of Tables

1 Details of Group-16 . . . 1
2 Event table of Car renting system . . . 26
3 Priorities for different criteria of Car Renting System . . . 30

Listings

2 Introduction

Car Renting System is being developed for customers so that they can book their vehicles from any part of the world. This application takes information from the customers through filling their details. A customer being registered in the website has the facility to book a vehicle which he requires. The proposed system is completely integrated online systems. It automates manual procedure in an effective and efficient way. This automated system facilitates customer and provides to fill up the details according to their requirements. It includes type of vehicle they are trying to hire and location. The purpose of this system is to develop a web site for the people who can book their vehicles along with requirements from any part of the world.

2.1 Project Description

This project is designed so as to be used by Car Rental Company specializing in renting cars to customers. It is an online system through which customers can view available cars, register, view profile and book car. The advancement in Information Technology and Internet penetration has greatly enhanced various business processes and communication between companies (services provider) and their customers of which car rental industry is not left out.

This Online Car Renting System is developed to provide the following services:

• Enhance Business Processes: To be able to use internet technology to project the rental company to the global world instead of limiting their services to their local domain alone, thus increase their return on investment (ROI).

• Online Vehicle Reservation: A tools through which customers can reserve available cars online prior to their expected pick-up date or time.

• Customer's registration: A registration portal to hold customer's details, monitor their transaction and used same to offer better and improve services to them.

• Group bookings: Allows the customer to book space for a group in the case of weddings or corporate meetings (Event management).

• The content management system (CMS) for managing the content of the cars.

• The data security system.

• Reporting of the cars, booking etc.

2.2 Problem Statement

A car rental is a vehicle that can be used temporarily for a fee during a specified period. Getting a rental car helps people get around despite the fact they do not have access to their own personal vehicle or don't own a vehicle at all. The individual who needs a car must contact a rental car company and contract out for a vehicle. This system increases customer retention and simplify vehicle and staff management.

2.3 Aim and Objectives

• To produce a web-based system that allow customer to register and reserve car online and for the company to effectively manage their car rental business.
• To ease customer's task whenever they need to rent a car.

3 Requirement Gathering and Analysis

A requirement is a condition or a capability needed by a user to solve a problem or achieve an objective. It is a description of the features and functionalities of the target system. A system development project usually starts by defining the requirements of the system. Requirements convey the expectations of users from the software product. The requirements can be obvious or hidden, known or unknown, expected or unexpected from the client's point of view. The requirements form the basis for development.

3.1 Requirements Gathering

Requirements are some criteria collected from users. It basically reflects what the user wants. There are different kinds of techniques to gather requirements.we describe some effective requirement-gathering processes.In this system, requirements are collected in two ways.

3.1.1 Interviews:

Interview is the most important part to collect requirements from users.We have arranged a meeting with Jahangir Jehad,a customer have faced some difficulties in travelling. Different types of questions have been asked on our behalf. Here are Some notable questions.
1. We heard you went to the Dhaka recently, How do you go there?
Ans: I went to Dhaka by renting a car.
2. When you reached Dhaka ?what kind of problems did you face there?
Ans:I reached Dhaka in late.The Driver is not Experienced.
3.Do you have to face any problem to rent the car?
Ans:I have to face huge problem.First day when i go to a renting office for renting there was no available car.For this reason Next day i again go and rent a car.But i can not find my favourite car.
4.What about the renting fees?
Ans:Fees are very high beacause there was no option for me.

3.1.2 Document Analysis:

Requirements elicitation phase of a project. It describes the act of reviewing the existing documentation of comparable systems in order to extract pieces of information that are relevant to the current project and therefore should be considered project requirements. Analyzing some documents online, we find closest car rental company for car renting.

Existing System:

•An existing system can provide manually paper work or excel sheet to track the booking and registered vehicles details.

•The user has to go in the office where the user can get the car on rent and book their car. Most of the time user does not get a sight of the car in which he is planning to travel. This results in compromising the travel comfort.

•In the existing system, you cannot provide feedback of the user to the admin directly. The user gets fluctuation every time he/she travels.

•Maintaining excel sheet or paper book record of reservation is very laborious work. Chances of error are more. No automation involves which means they are a very slow to process.

3.2 Types of Requirements

Various types of requirements related to our project are discussed below.

3.2.1 User Requirements:

User requirements are the abstract statement of the system. It's written for customers. The user demands the requirements our system will provide.
• Textual information in an emergency.
• Nearby repair centers and office locations.
• Automatic call and massaging service.

3.2.2 System Requirements:

After collecting requirements, we have to analyze them.By analyzing user requirements,we get system requirements.System requirements are description which determine the functionality of the system.

3.2.3 Functional Requirements:

These are statements of services the system should provide, how the system should react to particular inputs, and how the system should behave in particular situations. It specifies the application functionality that the developers must build into the product to enable users to accomplish their tasks.

Reservation

- The system must allow the customer to register for reservation.
- The system shall allow the customer to view detail description of particular car.
- The system must notify on selection of unavailable cars while reservation.
- The system must notify on selection of unavailable cars while reservation.
- The system shall allow the employee to view reservations made by customers.
- The system shall presents information on protection products and their daily costs, and requests the customer to accept or decline regulation terms during reservation.

Log in

- The system should allow manager to login to the system using their username and password.
- The system should allow employee to login to the system using their username and password.
- The system shall allow the manager to create new user account.
- The system shall allow manager to change account password.
- The system shall allow staff to logout.
- The system shall allow manager to logout.

Car

- The system should allow staff to register new cars.
- The system shall allow staff to select cars in the list.
- The system shall allow customer staff to Search cars by specific record.
- The system shall allow staff to update information of the car in need of modification.
- The system shall allow staff to display all rented car.
- The system shall allow staff to display all off duty car.

Rent

- The system shall allow staff to register customers into rental list.
- The system shall allow staff to update about customer rent record details in the rental list.
- The system shall allow staff to search rent record of customers using specific categories.
- The system shall allow staff to display customers, who rent cars.
- The system shall allow staff to display all customers rent record.
- The system must provide printable summary for successful committed rent

3.2.4 Non-Functional Requirements:

Non-functional requirements, as the name suggests, are requirements that are not directly concerned with the specific services delivered by the system to its users. They may relate to emergent system properties such as reliability, response time, and store occupancy. Alternatively, they may define constraints on the system implementation such as the capabilities of I/O devices or the data representations used in interfaces with other systems. Non-functional requirements, such as performance, security, or availability, usually specify or constrain characteristics of the system as a whole. These are the quality constraints that the system must satisfy according to the project contract. The priority or extent to which these factors are implemented varies from one project to another. They are also called non- behavioral requirements. Our system provides the following non-functional requirements.

• Simplicity: The architectural design of our system should be simple to use.System should be user-friendly

•Usability:The system provides a help and support menu in all interfaces for the user to interact with the system. The user can use the system by reading help and support

• Security: The system secure all confidential data.Must be security checked.The system provides username and password to prevent the system from unauthorized access.The staffs' password must be greater than eight characters.The subsystem should provide a high level of security and integrity of the data held by the system, only authorized Personnel of the company can gain access to the company's secured page on the system; and only users with valid password and username can login to view user's page.

• Performance: The system shall perform in iOS, Android, and Windows.Our system provides a quick response to users' actions.Performance should be better.The system response time for every instruction conducted by the user must not exceed more than a minimum of 10 seconds. The system should have high performance rate when executing user's input and should be able to provide response within a short time span usually 50 second for highly complicated task and 20 to 25 seconds for less complicated task.

• Availability: Our system provides 24x7 hours of service. Only needs internet to run our system.System must be portable. Also in the occurrence of any major system malfunctioning, the system should be available in 1 to 2working days, so that business process is not severely affected

• Flexibility: Our system is smooth to use.

4 Software Process Model

A software process is a set of related activities that leads to the production of a software product. We develop our software using both waterfall model and iterative model. The concepts of the methods, strengths and weaknesses will be discussed. This information will then be used to support the choice of method. Afterwards the chosen method and its application in the report will be described.For this reasons,we are using those model are discussed below in Subsection 4.1 and why the other models are not suitable for our system are discussed later in the Subsection 4.2

4.1 Suitable Models

The following is a discussion of the reasons why the model are appropriate for our purpose.
The waterfall model: Waterfall model is a plan driven type process and iterative model is agile type process.Waterfall model have well specified steps and help us to structure our project and write detailed documentation about project.One of the strengths of the waterfall method is the consecutive order of progress the structure gives. This gives the project members an overview of how far they are in a specific part or in the overall progress of the project.

The Iterative model: On the other hand Iterative model is flexible with it's activities and for changing the situation and if we further want to add a feature that don't notice to add.Iteration gives a better understanding of the project context, development and requirements.

So we can make version of our software or if a serious bug is issued than we can immediate change this. Both methods have their strengths and weaknesses and one method might have a strength that is a weakness for the other. When using this method it can be difficult to keep an overview because each iteration adds more information. On the other hand each iteration gives a better understanding of the project context, development and requirements. This could lead to an endless circle of improving and expanding the project, rendering it difficult to say when the project is actually finished. This could prove to be difficult to answer, whereas the waterfall method gives a structure that restricts the multiple iterations and sets an end point. The waterfall model provides an easy overview of the project progress. It is also a good choice because the system requirements and the goals that have been set, can be defined after each part of the documentation. This allows for confirmation of the goals and requirements that have been set.

Waterfall model steps:

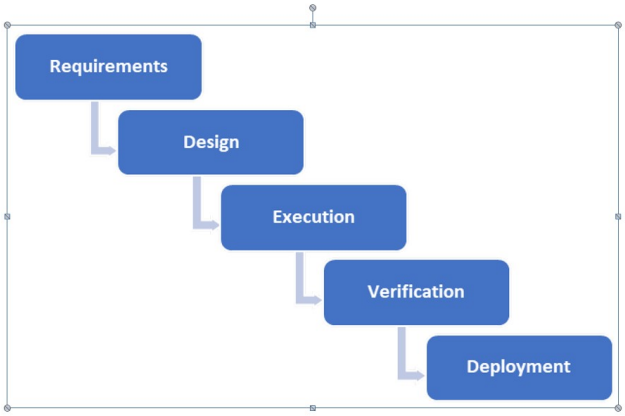

Figure 1: The Waterfall Model

Iterative model steps:

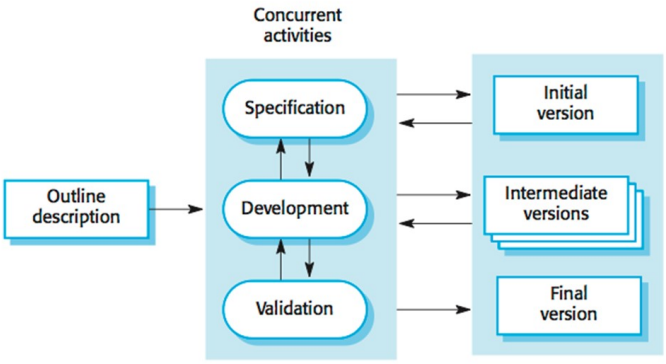

Figure 2: The Iterative Model

4.2 Not Suitable Models for Our System

The following is a discussion of the reasons why the other models are not appropriate for our purpose.

The Reuse-Oriented Model: When people working on the project are aware of designs or codes that are comparable to what is required, this frequently occurs informally. These are sought after, modified as necessary, and added to the system. The majority of the time, developers are not concerned with user needs. This could result in a system that doesn't actually serve the demands of people.

On a website, we looked for projects that were identical to ours or parts that were associated with it.But we failed to locate it.Additionally, we collect requirements through online surveys and interviews to ensure that they reflect the actual needs of users.As a result, this paradigm is inappropriate for our purpose.

The Spiral Process Model: The spiral model places a strong emphasis on risk analysis and combines the iterative development process model with sequential linear development model, or the waterfall model.

Use of this model when there is a budgetary restriction, risk assessment is crucial, and the consumer is unsure of what they need.The requirements from the customer for our project are rather straightforward.For our project, there is no need to place a lot of focus on risk analysis.We make an effort to keep project development expenses to a minimum.As a result, this paradigm is inappropriate for our purpose.

5 System Choice

System choice must be based on 3 sub-activities.

5.1 Appreciate the situation:

By working with "Rich Pictures",we can explicate important user views of a situation, facilitate debate and get an overview of the situation quickly.

5.1.1 Rich Picture

A rich picture is an informal drawing that presents the illustrator's understanding of a situation. It will give us the current scenario of what a person faces difficulty who is in bad situation and how to solve the problem without facing any problem.We will highlight the situation we are facing with the help of this rich picture.The purpose of this picture is to highlight the problems of the existing system and to understand them easily.

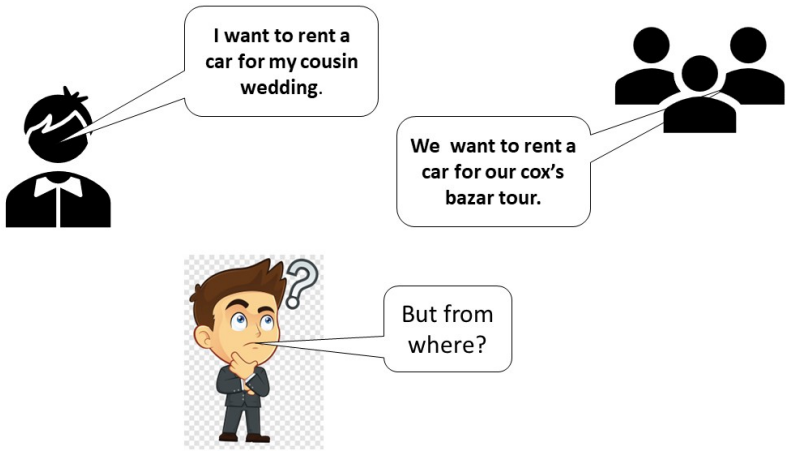

Figure 3: Rich Picture of Car Renting System

Figure 4: Rich Picture of Car Renting System

5.2 Cultivate new Ideas

After analyzing the current situation problem, we decided to develop a system that will solve the arising issue in the rich picture. If it happends any accident We decided to provide first-aid services which can be given by anyone. Besides survival guidelines are also given which can save someone's life. Most of the cases, people don't remember emergency services. Keeping this in mind, we decided to provide emergency services. In case, the emergency services is busy, for that we decided to provide a customized map that will show the nearest repair service or garages. If the customers are in the danger zone, we decided to provide an emergency calling or messaging service to all favorite numbers by artificial intellegence.

5.3 FACTOR

Before defining our system, we need to include some characteristics of the definition of the system which is known as FACTOR.
The FACTOR criteria of our Car Renting System are given below:
1. F-Functionality: Our system can be used for emergency repair service center,emergency helpline services,first-aid treatment,and locating nearby garage/sevice center through a customized map.
2. A-Application domain: A good raod for driving and nearby good machanic.
3. C-Conditions: Must have a smartphone.
4. T-Technology: Global mapping and automated calling and messaging service.
5. O-Objects: To help a damage car and an endangered person.
6. R-Responsibility: This system will act as a communication medium for saving times,money and life.

5.4 System Definition

A system definition expresses fundamental properties for system development and use. It describes the system in context, what information it should contain, which functions it should provide, where it is to be used, and which development conditions apply. A system definition should be brief and precise, and contain the most fundamental decisions about the system. Creating a brief and precise formulation provides an overview and makes it easier to compare alternatives. A long,detailed description makes this difficult.
The system definition of our system is:
"A computerized system which is used for seeking help from emergency,locate nearby garage/mechanic/rescue team and give support on problematic situation. The system should be operable in any smartphone with required memory space. Both video and textual description will be available to give servics and survival guideline to the person. The system can be used with no prior experience by a person who have the mentality to help a traveller who is struggling. A user account should be created to send and message to multiple person those who are likely to help the person."

6 Problem Domain

In order to understand a system well and develop it in a good way, a developer has to know the problem domain of the system well. The Problem domain is the part of the system, which gets administrated, monitored, or controlled by the system. The purpose of the Problem domain is trying to identify the parts of the system, and model them in the categories: Classes, Objects, Structures, and Behavior to give an overview of that part of the system. Once an overview has gotten achieved, the developer can start to supply details to different categories. As an example, this can get achievable by structuring relations between Classes and Objects. An experimental attitude is vital in order to make a strong single coherent model, is an iterative process, which in most takes time to perfect. In many cases, even an experienced developer needs to go back and add or delete a new Class in the model. The final result is a coherent model, with good relations between Classes, Structures, and Behaviors. This Section and its subsections create an overview of this project's Classes and Class Relations. The goal is to achieve a clear overview of the system

6.1 Classes

Before being able to provide a full picture of the Problem domain, the classes of the Problem domain must be described in detail first. In this Section, the actors of the system will be introduced and somewhat described. actors would usually be dealt with in the Application domain of the system, but are introduced in this part since they play a crucial role in the system and within the Problem domain. Every class described in this Section is a blueprint for objects that contain data on different parts of the system, even the actors

The chosen classes from the class candidate analysis will be defined in this section.

6.1.1 Physical

Car: This class holds information about the car details.
TripDetails: This class hold information about the running and completed trip.
BookingInfo: This class hold the information about booking request and accepted request.

6.1.2 Person

Passenger: This class hold the information about passenger who are the general user of this system.
Admin: This class hold he information about the main authority of the system ho can control the user of the system.
Driver: This class hold the information about the driver of the specific car renting service center.

6.1.3 Places

CarRentCenter: This class hold information about the car service center,it's location,car collection,completed trip,contact number and providing service details.

6.2 Class Diagram

A class diagram is used to describe and illustrate, in a visual format, the relation between classes in a given system. Using the visual attributes of the class diagram, it becomes much easier to understand how different parts of a system, is related.

The class diagram is structured with CarRentCenter as central class.Car class and Driver class are containing the details of car and driver associated with the CarRentCenter class. If the CarRentCenter will delete Car and Driver class will be automatically deleted.So Car class and Driver class have composition relationship with CarRentCenter class.CarRentCenter,Passenger and Admin class inherit the User class as at first all member's are User. Car class has association relationship with BookingInfo class.TripDetails class has aggregation relationship with BookingInfo class as if the BookingInfo class is deleted the TripDetails class have to sustain.

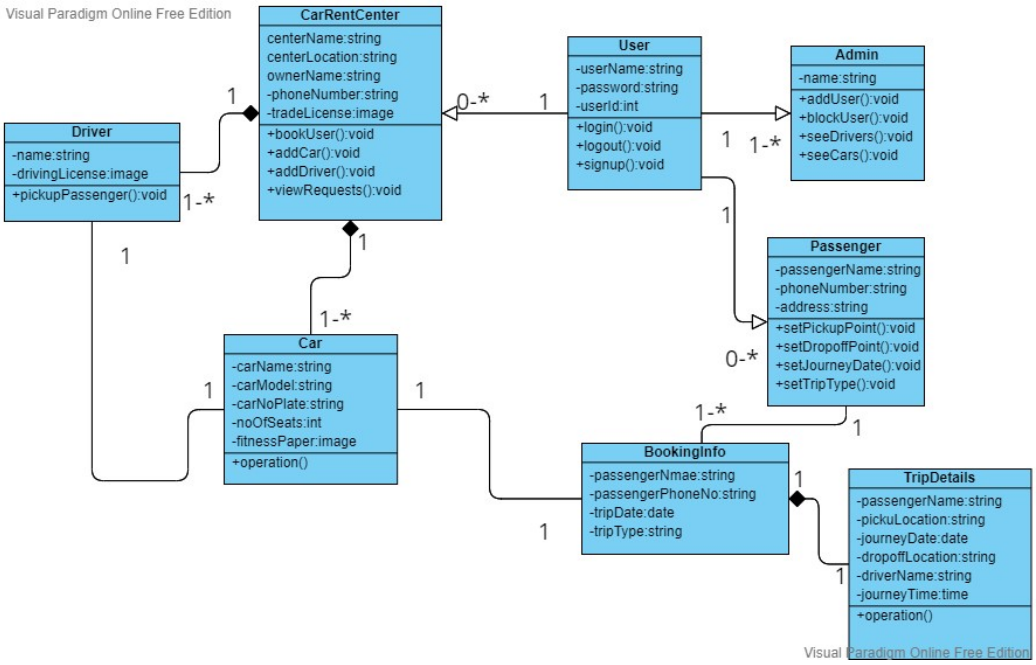

Figure 5: Class Diagram of Car Renting System

6.3 Events

In this section a list of events will be presented. The events described are those which have been chosen through an event candidate analysis. The events and what trigger them are listed below.

6.3.1 Event Definition:

Event selection defines the second set of building blocks for problem domain model. However,there is a fundamental difference between events and classes. Numerous verbs relate to the way users carry out their jobs. Such verbs do not belong to the problem domain, but rather to the application domain.We can eliminate many of these immediately.

The chosen events from the event candidate analysis will be defined in this section.

Car added: A car is added to the Car class when CarRentCenter want to add a new car to his car collection or have an addition from its original volume/quantity.
Car removed: A car should be removed from the car collection, or have an subtraction from its original volume/quantity.
Trip completed: When a trip is in drop-off date, it should be removed from the inventory and thrown out. Booked a passenger: From available request for this car booked a passenger.
Send a request for car: After searching the available car of car rent center a passenger can send a request.

Select Pick-up location: Passenger have to select pickup location for searching.
Select Drop-off location: Passenger have to select drop-off location for sea.
Pickup date:select pickup day , month and year from calendar.
Pickup time:select pickup time from set exact second, minute and hour in time frame.
Drop-off date:select drop-off day , month and year from calendar.
Drop-off time:select drop-off time from set exact second, minute and hour in time frame.
Add a driver: Rent service center owner add a new driver with additional information of driver like driver name, driver age etc.
Remove a driver: Delete a driver from all driver list.
Car on trip: Set car name ,driver name ,pickup and drop-off location.
Add trip type: Add different type of trip from wedding , trip around city and trip outside of the city . Trip type describe the service type that the car renting service center want to give passenger . Also set cost of per according to trip type.

6.3.2 Event Table

Based on the classes in Section 6.1 and the class relations in the class diagram in Section 6.2, the events between the classes will now be described. The figure 6.3.1 shows the most common events, which occurs between the classes in the system. The events on the figure mainly describe the events cause by the actor classes since these are the classes which cause most events to occur. Since the figure describes the classes from the previous Sections relations based on their events, it is possible to attain an even better overview of the full system.

Shows an event table that have been constructed in order to get an overview of the relations between classes and events. The event table allows for a better judgement of which classes are relevant in the program. A plus (+) in the table indicates, that an event can occur zero or one time, whereas a star (*) indicates that an event can occur zero or more time.

Event	Car	Passenger	TripDetails	CarRentCenter
Car added	*			
Car removed	*			
Booked a passenger		+		+
Send a request for car		*		*
Select pickup location		*	+	
Driver on trip			*	*
Add trip type			*	+
Select trip type		+		
Add cost per km into city			*	+
Add cost per km outside city			*	+
Select pickup date and time		*	+	
Select drop off date and time		*	+	
Add a driver				*
Remove a driver				*
Available car list	*			+
Available driver list				*
Car on trip	*		*	+
Trip completed	*		+	

Table 2: Event table of Car renting system

6.4 Behavior

A behavioral pattern orders individual events in time using fundamental control structures from structured programming. A behavioral pattern with sequence, selection, and iteration can be described most comprehensibly by a regular expression.
Sequence: Events in a set occur one by one.
Selection: Exactly one out of a set of events occurs.
Iteration: An event occurs zero or more times.

7 Application Domain

As stated in problem domain a developer needs to understand the system full context, and application domain is the other part of domain analysis. Application domain is that part of the system, which is operated by an organization that administrates, monitors and controls a problem domain. The purpose of the application domain is the understanding of the system's usage requirements. This is done by engaging with the users of the future system, in order to get as many information as possible, but even then it might not be enough. The user may not be able to understand technical options and not be able to write them down well enough. This can be overcome by making use case and actors of the future system, to get the proper image of the problem illustrated in pictures to get a better overview of the overall usage requirements. With a better overview of usage requirements, it is possible to start questioning the information processing capabilities, which will make developer able to the create a list of functions and related interfaces in the system. The final result is a complete list of overall usage requirements of the system.

7.1 Usage

7.1.1 Actors

To understand the usage of the product an understanding of how actors and the targeted system, an explanation will be given. An actor is an abstraction of users who can or will have an interaction with the system, which the system is the product. An example of an actor would be the instructor or the students who participate to get drivers licenses. To better understand the viewpoint of the access level for the system, a Table overview will present itself.

7.1.2 Use Case Diagram

To understand the usage of the product an understanding of how actors and the targeted system, an explanation will be given. An actor is an abstraction of users who can or will have an interaction with the system, which the system is the product. An example of an actor would be the instructor or the students who participate to get drivers licenses. To better understand the viewpoint of the access level for the system, a Table overview will present itself.

7.2 Functions and Complexity

8 System Architecture Design

This chapter describes the priority of system criteria, and the system structure which is divided into component architecture and process architecture, with descriptions of the models that has been used.

8.1 Priority of Criteria

When designing a software based system, it is important to consider which criteria are needed for the system, and if some of the criteria are more important than others. The criteria will be organised in to the bellow figure, the importance of them range from very important to trivially fulfilled.

Criteria	Very important	Important	Less important	Irrelevant	Trivially fulfilled
Useful		X			
Secure			X		
Effective			X		
Correct	X				
Reliable			X		
Maintenance		X			
Testable		X			
Flexible		X			
Understandable			X		
Reusable				X	
Movable				X	
Integrable			X		

Table 3: Priorities for different criteria of Car Renting System

Useful indicates the adaptation of the organisational, work related, and technical surroundings. This is important for the Car Renting system to fulfil, since the organisational surroundings indicates that the user has to be able to rent and trip correctly.
Secure indicates if the system is secure against unauthorized accessibility. This is less important in the Car renting system, since the system does not contain sensitive personal information.
Effective indicates the economical usage of the technical platform facilities. This is marked less important, as today's smartphones are capable of supporting the requirements needed for this system.

Correct indicates the level of fulfilment of the formulated requirements. The correctness of the Car renting system is very important, as it is needed to know about correct location for the system.
Reliable indicates the fulfilment of the required precision when functions are executed. The reliability is less important for the Car renting system, as functions can work a little different than stated, as long as the user gets a usable result.
Maintenance indicates the cost of finding and correcting errors in the system. The maintenance of the Car renting system is important, as errors will be made throughout the project. Therefore it is important that errors can be corrected, for the system to work properly.
Testable indicates the cost of securing that the system fulfils the requirements. It is important for the Car renting system to be testable, as testing is part of the waterfall method, and also a good way for the system to be maintainable which is important to the system.
Flexible indicates the cost of changing the system once it has been taken into use. Flexibility is important for the Car renting system, since changes and updates could be needed, in order to maintain the program in the future either because new functionality, or updates.
Understandable indicates the effort to make sure that the system can be understood in the context.Understand ability is less important to the Car renting system, as the system is not planned for other developers to work on, and will not be used in a larger context.
Reusable indicates the possibility to reuse parts of the system in similar systems. Reuse is irrelevant as the system is not planned to be used in relation with other systems.
Movable indicates the cost of moving the system to another technical platform. For the Car renting system it is irrelevant to consider the move ability, as the system is not planned to be moved to other technical platforms than smartphones and tablets.
Integrable indicates the cost of connecting the system to other systems.Integrability is less important for the Car renting system, because the system will probably not be connected to other systems,unless API are going to be included.

8.2 Architechtural Design

8.2.1 Architechture Component

8.2.2 Model Component

9 User Interface Design

10 Implementation

10.1 Code Snippet

description write here.....
Pest code here.....

description write here.....
Pest code here.....

11 Testing

12 Software Deployment

13 Conclusion

Car rental business has emerged with a new goodies compared to the past experience where every activity concerning car rental business is limited to a physical location only. Even though the physical location has not been totally eradicated; the nature of functions and how these functions are achieved has been reshaped by the power of internet. Nowadays, customers can reserve cars online, rent car online, and have the car brought to their door step once the customer is a registered member or go to the office to pick the car.
The web based car rental system has offered an advantage to both customers as well as Car Rental Company to efficiently and effectively manage the business and satisfies customers' need at the click of a button.

14 Bibliography

1.System, Online. 'Online Car Rental System'. Academia.edu. N.p., 2015. Web. 9 June 2015.

2.Scribd.com, Online. '49930505 Car Rental System Project Report'. N.p., 2015. Web. 9 June 2015.

3.Scribd.com, Online. 'Car Rental System Documentation'. N.p., 2015. Web. 9 June 2015.

4.Freelancer, Online. 'Project Documentation Car Rental Company Software Development Freelancers and Jobs - Freelancer'. N.p., 2015. Web. 9 June 2015.

5.Slideshare.net, Online. 'Zook Car Rental System Project'. N.p., 2015. Web. 9 June 2015.

6.Kaewman, Sasitorn. 'Online Decision Support System of Used Car Selection using K-Nearest Neighbor Technique'. IJFCC (2012): 164-166. Web.

7.Wikipedia, Online. 'Use Case Diagram'. N.p., 2015. Web. 9 June 2015.
8.Wikipedia, Online. 'Activity Diagram'. N.p., 2015. Web. 9 June 2015.
9.Tutorialspoint.com,Online'UML-Activity Diagrams'N.p,2015.Web.9 June 2015
10.Wikipedia, online. 'Swim Lane'. N.p., 2015. Web. 9 June 2015.
11.Mindtools.com, Online. 'Swim Lane/Rummler-Brache Diagrams: Mapping and Improving Processes in Your Organization'. N.p., 2015. Web. 9 June 2015.

12.Laudon, Kenneth C, and Jane Price Laudon. Management Information Systems. Upper Saddle River, NJ: Prentice Hall, 2000. Print.

13.Menkus, Belden. 'Car Rental Chain Former Owners Charged With Computer Frauds'. Computer Fraud Security Bulletin 1993.3 (1993): 3-4. Web.

14.Li, Zhang. 'Design And Realization Of Car Rental Management System Based On AJAX+SSH'. Information Technology J. 12.14 (2013): 2756-2761. Web.