# Analysis:

## 2.1 – Introduction

Analysis means a process of collecting data or information at the different parts of a project, how they fit together, and what action to take. Analysis is the interpreting facts, identifying problem and evaluate the information we have and the way of what the system actually works. Analysis mostly focus on what the system should be according to the user. in any project like car rental management system analysis is needed. It is the first phase of Software development life cycle, helps to find out the requirement of the project to be developed.

## 2.2 – Analysis Methodology

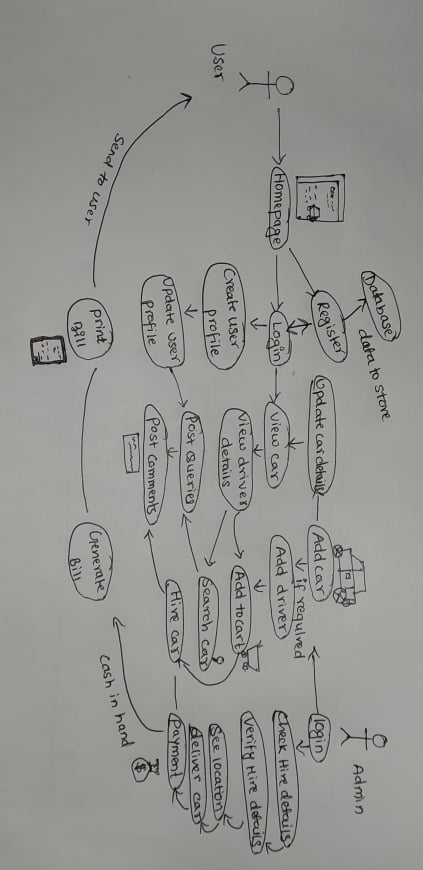
Analysis Methodology is required to develop the system for structure complex problem and to have flexible changes. There are several type of analysis methodology. Among them I have chosen soft system methodology in my project. Soft System Methodology is more people centric or people oriented. Information gathering is needed for recognize problematic situation.

**Advantages:**

* Human activity and involvement makes easy to work and efficient.
* Dealing with complex problem.

**Steps to taken in Soft System Methodology:**

1. Analyze and create rich picture.
2. Define root definition.
3. Produce conceptual models of system
4. Compare concept of the system with actual system
5. Define and select feasible options for development
6. Implement the system
7. **Rich Picture:** The Picture which is usually drawn by hand in the Paper including whole system view with structures, processes and issues. No rules and boundaries are explained. The priorities element are customers, competitors, suppliers, staff, products, department, hardware, software and so on. In the additional elements, some aspects like social, cultural roles, norms, values, goals, political power and problems. I have made the rich picture of my system and the picture below:

****

**Fig: Rich Picture**

I have drawn rich picture in my project to describe the real world as to identify the issue I wish to address and also it can describe of situation where the issue lies. Above rich picture describes whole my system as also which actor play which roles.

1. **Root** **definition**: Root definition defines what the system is and stating the main purpose of the system. It describes the aim and function of the system to be develop. There are two type of root definition: primary task which focus on system process and issue based task, focus on problems. CATWOE analysis helps to defining root definition. It helps identify all the people, processes, external factors involved in the information system that to be analyzed. In my project CATWOE is important to prompt thin king what the system is trying to do and categorized the whole system. CATWOE refer to:

**Customers/Clients**: Customers/Clients are those stakeholders whom the system is exists, who benefits or suffer when the system process updates.

**Actors/Agents**: Actors/Agents are required for listing out the stakeholders responsible for specific task for implementing the changes of the system.

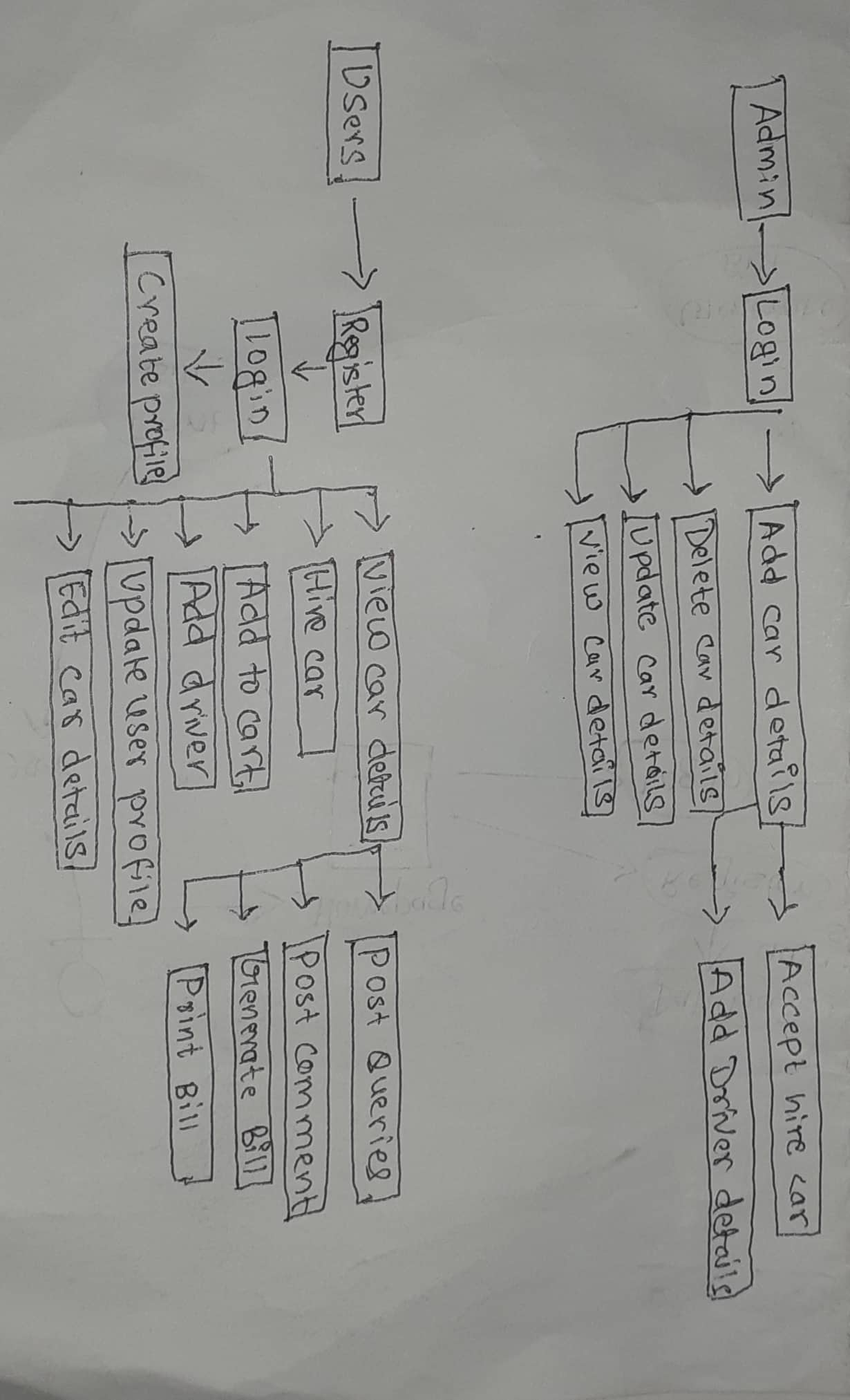
**Transformation**: Transformation refers to the processes affected by the development of the system.

**World View**: World View refers to the see what going outside and inside the organization that may be influencing the system development.

**Owners**: Owners refers to the person who own the organization or system.

**Environment**: Environment refers to the investigation whether this will be affected politics, social, economic, ethics and so on.

1. **Conceptual model:** Conceptual model is construct with the help of rich picture and root definition. Conceptual is used to explain how system should function and necessary activities for the processes. The system performance can be measured too.

****

**Fig: Conceptual diagram**

Above conceptual diagram helps to describe how the system should function and what activities is necessary for my system.

## 2.3 – Feasibility Study

Feasibility study evaluates the project, therefore, project fits under circumstances and also evaluate the ability, different factors to be completed. It includes the estimate level of expertise require for the project, who can provide it quantitative and qualitative assessment of other essential resources. This study brings positive and negative outcomes of the project. There are different types of feasibility study. Some of them are:

1. **Technical Feasibility:** this technical feasibility helps to focus on technical resources available to the organization. It helps organizations determine whether the technical resources meet capacity and whether the technical team is capable of converting the ideas into working systems. It also involves evaluation of the hardware, software and other technology requirement of the system.
2. **Economic Feasibility Study:** Economic Feasibility refers to the fitness of the respective project to produce economic profit/benefits. The study is also known as cost benefit analysis. Here, cost to build the project is estimated such as budget, allocation, profitable or not. Our system needs website of own which are accessible easily from device having internet service. For website we need a domain name and it is affordable to have one. The order parts are the customer’s assets.
3. **Operational Feasibility:** Operational feasibility is the measure how effectively the proposed system can solve the problems, and fulfill the identified requirements. Management of the project is welly maintained. The system operation provides adequate and response time. Large number won’t be active at a time so there is no risk while operating.
4. **Scheduling feasibility**: this feasibility study helps to check the time allocated is matching while completing project or not. For my project the allocated time is matching while working in my project. I have created time estimated and Gantt chart for scheduling feasibility.
5. **Marketing feasibility**: this feasibility helps to marketing the project. The system will benefit the user so that they can book or hire car from their home. Marketing helps to exits the system to the market around the user.

## 2.4 – Requirement Analysis

Requirement Analysis is also called Requirement Engineering which means the process of determining the expectation for the product or software to be built. It includes all the task perform by the user to identify the user needs. The requirement must be relevant and detailed. Such requirements are of two types Functional and Non-Functional Requirements

### .2.4.1 – Functional Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Title** | **Description** | **Rational** | **Dependency** |
| FR.001 | Registration | Signing up user. | Keeping record by giving required information. | N/A |
| FR.002 | Login | Input username and password then access to the system. | Authenticate to the system. | FR.001 |
| FR.003 | Verification | Verify user exits or not. | Need to identify the customer. | FR.002, FR.013 |
| FR.004 | Add car Details | Any type of car are added. | Adding the car details along with car image as per colors also. | FR.002 |
| FR.005 | Update Car Details | Changes can be update. | Update the details in case of changes. | FR.002, FR.004 |
| FR.006 | Delete car Details | Delete the car in the case of not available. | Delete the car in not available situation. | FR.002, FR.004 |
| FR.007 | View Car Details | Choose car and view details. | Show the detail of car | FR.002, FR.004, FR.005, FR.006 |
| FR.008 | Search car | Search car by user. | Helps to search car as the requirements. | FR.002, FR.004, FR.007 |
| FR.009 | Add Driver | Driver should be added for riding the car. | As per the needs driver can be added. | FR.002 |
| FR.010 | View Driver Details | User should be able to see the detail about the driver. | Helps to choose the driver by customer by seeing the experience. | FR.002, FR.009 |
| FR.011 | Update Driver | The drivers detail should be able to update. | Helps to change the driver details. | FR.002, FR.009, FR.010 |
| FR.012 | Delete Driver | Delete the driver along with details in the case of not available. | Helps to delete the driver. | FR.002, FR.009, FR.010 |
| FR.013 | Hire Car | User will be able to hire a car for a long tour. | Helps to hire a car. | FR.001, FR.002, FR.004, FR.007 |
| FR.014 | Hire Driver | User will able to hire a driver if they needs. | Helps to hire a driver |  |
| FR.015 | Update user details | Changes in user profile. | Update the change in user information. | FR.001, FR.002 |
| FR.016 | Add to Cart | Collection of the favorite car. | Helps to order by choosing the car. | FR.002, FR.004, FR.009 |
| FR.017 | Generate Bill | User should be able to create bill of the hire cars. | To list out the car and generate the bill. | FR.002, FR.013 |
| FR.018 | Print Bill | User must be able to print the bill | After generating bill print bill helps for proof. | FR.016 |
| FR.019 | Post Quires | Any kind of question post question. | Helps to see the problem by the helps of users post. | FR.002, FR.004, FR.010 |
| FR.020 | Post comments | Any kind of suggestion, satisfaction can be shared. | Helps to see users feedback. | FR.002, FR.018 |
| FR.021 | Logout | Logout the user. | Logout the user from the system. | FR.002 |

### 2.4.2 – Non-Functional Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Title | Description | Relational | Dependency |
| N.FR.001 | Security | System provide username and password and encrypt data before storing. | Save from unauthorized access. | FR.002 |
| N.FR.002 | Performance | The system response must fast. | Need to have high performance rate. | N/A |
| N.FR.003 | Availability | In all working days need the system available. | Helps to run Business process smoothly. | N/A |
| N.FR.004 | Maintainability | Considerably minimized and fix error. | Bugs fix rapidly needed. | N/A |
| N.FR.005 | Efficiency | System should fast to effective work. | Increase working pulse. | N/A |
| N.FR.006 | Compatibility | System work in any devices and in browser. | Easy to work by any devices. | N/A |
| N.FR.007 | Scalability | System need to adapt any change. | Updates adapt. | N/A |
| N.FR.008 | Usability | System should user friendly. | Helps to understood easily. | N/A |
| N.FR.009 | Integrity | User data should not change by unauthorized person. | Increases user trust. | FR.001,  FR.005 ,FR.011 |
| N.FR.010 | Portability | System need to be access in any device. | Any devices access is needed. | N/A |

### 2.4.3 – Moscow Prioritization

It is used to determine which requirement of system should be including in certain delivery. The prioritization is important because it decrease the risk as high/important requirement is given more priorities. It helps to provide certain needy service to an application that can be used before fully completed application. The importance of Prioritization is:

* To identify which tasks is important and giving more time, attention, and energy to it.
* Urgent or high priority is given to the task and be cautious about it.
* If everything is important then everything is must be done and loses its urgency and mayn’t be delivered.

**MoSCoW table use in functional requirement:**

|  |  |  |
| --- | --- | --- |
| **ID** | **Title** | **MoSCoW** |
| FR.001 | Registration | Must have |
| FR.002 | Login | Must have |
| FR.003 | Verification | Should have |
| FR.004 | Add car Details | Must have |
| FR.005 | Update Car Details | Must have |
| FR.006 | Delete car Details | Should have |
| FR.007 | View Car Details | Must have |
| FR.008 | Search car | Must have |
| FR.009 | Add Driver | Should have |
| FR.010 | View Driver Details | Could have |
| FR.011 | Update Driver | Should have |
| FR.012 | Delete Driver | Should have |
| FR.013 | Hire Car | Must have |
| FR.014 | Hire Driver | Should have |
| FR.015 | Update user profile | Must have |
| FR.016 | Add to Cart | Should have |
| FR.017 | Generate Bill | Must have |
| FR.018 | Print Bill | Should have |
| FR.019 | Post Quires | Would have |
| FR.020 | Post comments | Should have |
| FR.021 | Logout | Must have |

**MoSCoW table use in non-functional requirement:**

|  |  |  |
| --- | --- | --- |
| **ID** | **Title** | **MoSCoW** |
| N.FR.001 | Security | Must have |
| N.FR.002 | Performance | Must have |
| N.FR.003 | Availability | Would have |
| N.FR.004 | Maintainability | Would have |
| N.FR.005 | Efficiency | Must have |
| N.FR.006 | Compatibility | Should have |
| N.FR.007 | Scalability | Could have |
| N.FR.008 | Usability | Should have |
| N.FR.009 | Integrity | Must have |
| N.FR.010 | Portability | Must have |

### 2.4.4 – SRS [Software & Hardware Requirements]

Software and Hardware Requirement in my system are the description of components like speed, model, processor, browser and so on. For the best performance of my system the hardware and software specification are:

Pre-project requirements

|  |  |
| --- | --- |
| Hardware | Software |
| Laptop(hp)  Processor(i5)  Ram (8 GB)  Hard Disk (750 GB) | XAMPP  Visual Studio Code  Windows 10 Pro  Google chrome  Star UML |

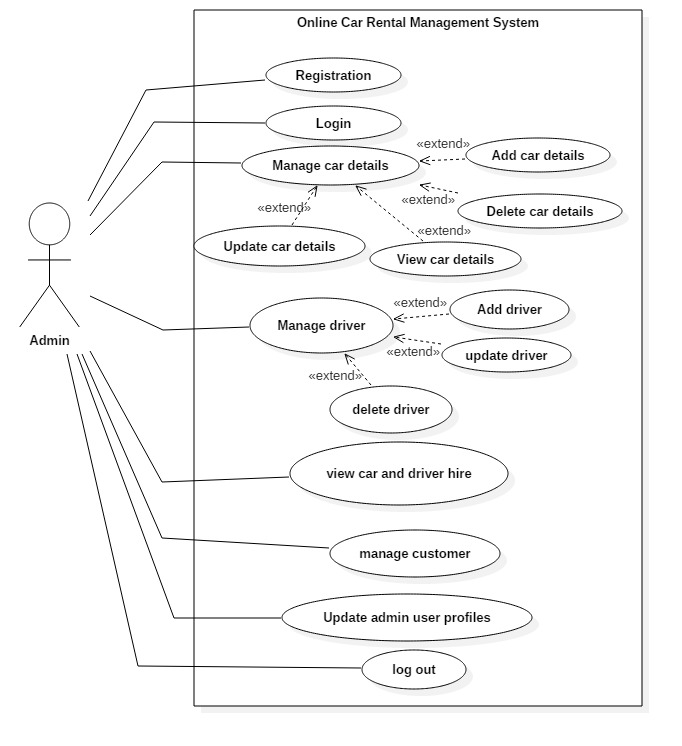
Post-project requirements

|  |  |
| --- | --- |
| Hardware | Software |
| Android devices  Internet connection  **For desktops and laptops**  Minimum Core 2 Duo and above processor  RAM minimum 2 GB  Hard Disk space (minimum 100GB) | **Operating System**  Windows 10, 8, 8.1  **Browsers**  Google chrome  Mozilla Firefox |

## 2.5 – Use Case Diagram

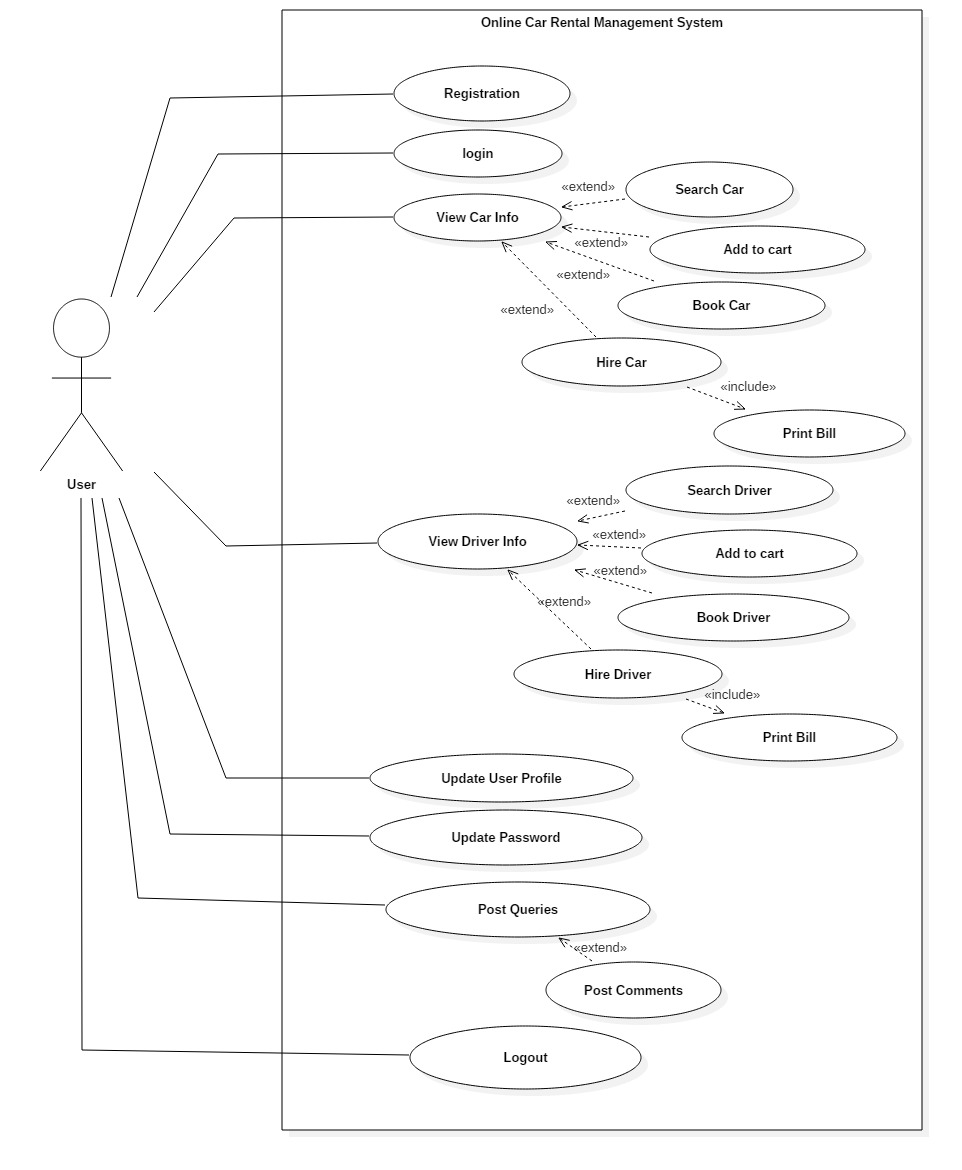
The diagram which is an illustration of user interacting with the system to achieve goal providing overview the relationship between user and variety of use cases. This process is use case diagram. The symbol description used in use case are: Rectangular box to illustrate the system. Actors are those who perform actions. Use Case are the list of process that describe the action perform by actor in a system. Objects are instance of class. Interface are elements that describe set of operations. “Online Car Rental Management System” is a system shown in a diagram below. In my system two actors I have created in my use case, admin and user.

The use case diagram of my proposed system is as below:



**Fig: Use case of admin**

In the above use case diagram, Admin can register and login to the system. After the login, Admin can add, update, delete, and view the car & driver details. Admin can verify the hire car and driver details. Admin can manage the customer’s details.



**Fig: use case of user**

In the above use case of user of my system, online car rental management system user can register and login to system. As per their needs of car they can search the car, add to cart, book car. And also in case of necessary of driver user can also hire driver and generate bill. Update their profile can be done by user. if the user wants to update or delete book cars they can delete and update it.

## 2.6 – NLA & Initial Class Diagram

The Car Rental Company located in Dili bazar, Kathmandu is providing service from 2016 to till. The existing system of this company have manual paperwork. The user has to visit office where user can get the car on rent and book. This system cannot provide feedback of the user to the admin. After 2 years of company running the owner of the company want to make online data store system along with online booking system.

This company need Online Car Rental Management System is designed to simplify the management and booking process and allow to get on with running all type of vehicle rental business. This system needs features of admin like registration, log in, log out, add car, delete car, update car details, add driver, payment: cash on hand and manage booking car and rent. The user can registration, login, create profile, search car, add to cart, book car, give feedback using forum, send enquiry. This system must contain verification file like print of driving license and citizenship. User can view the order total price of the rented car. The user and admin both can generate bill for pay while delivering the car. User can contact in phone and email if any kind of problem happen.

Online Renting the car helps to save valuable time for visiting the office and fulfill order from far distances. For fulfilling the demand user should verify themselves by name, shipping address and then user can get what they want.

|  |  |  |
| --- | --- | --- |
| ***Noun*** | ***Adjective*** | ***verb*** |
| Registration, Account, admin, Login, User, Feedback, Car, Driver, Cart, | Email, Username, Password, enquiry. | Add, Delete, Update, Search, View, Book ,Hire ,Calculate |

**Candidate Class:**

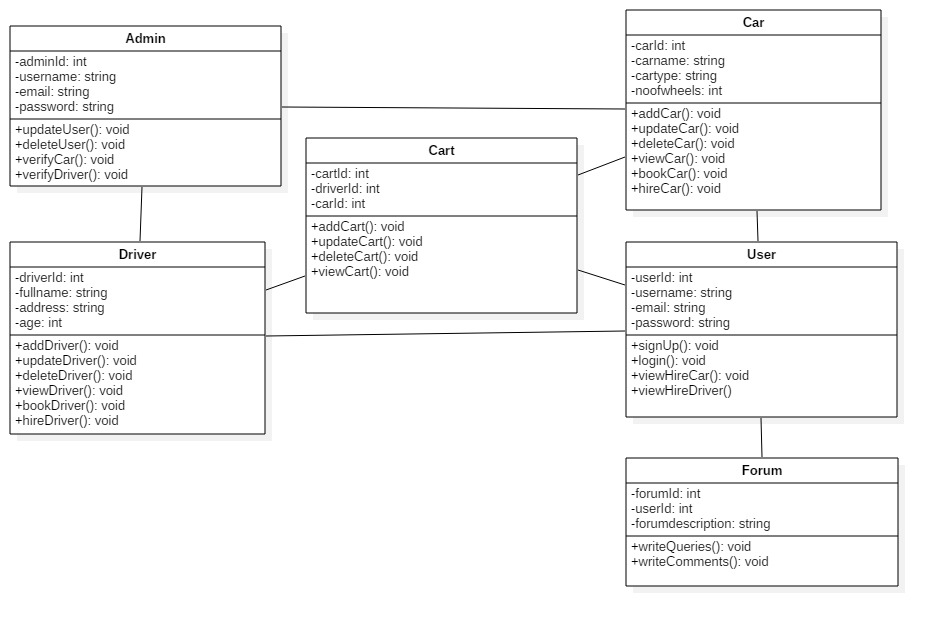
|  |  |
| --- | --- |
| **Candidate Class** | **Reasons** |
| Admin | Admin can manage the car and the driver details as well as the user profile. |
| User | User can book a car, view car and add to a cart and hire it. |
| Car | It has functions like add, delete, update, book a car and hire |
| Driver | It has functions like add, delete, update, book a driver and hire. |
| Forum | It can be able to add queries and comments about the car and driver. |

**Operation:**

Add, Delete, Update, Search, View, Book, Hire, Calculate

## Initial Class Diagram

An initial class diagram shows the classes within a model and illustrates which classes interact. The design of the relationships and dependencies between classes. It helps to implementation the system in object oriented language which connect the classes. We can initially draw all these classes on a diagram with simple links between classes to represent collaborations. In the diagram access modifier like public (+), private (-). The initial class diagram of my system, online car rental management system along with classes, attributes and methods is below:



**Fig: initial class diagram**