Project Analysis On

**KING**

**Online Car Rental System**

Prabin Malla

00163020

Computing Project

Level 5 Diploma in Computing

Softwarica College of IT & E-Commerce

Kathmandu, Nepal

April 23, 2017

Contents

[Introduction: 3](#_Toc480879931)

[Use case: 4](#_Toc480879932)

[Requirements: 6](#_Toc480879933)

[Functional requirement: 6](#_Toc480879934)

[Non-functional requirement: 8](#_Toc480879935)

[Prioritization 9](#_Toc480879936)

[Architecture: 10](#_Toc480879937)

[System Architecture: 10](#_Toc480879938)

[Initial Class Diagram: 11](#_Toc480879939)

[Conclusion: 12](#_Toc480879940)

[References: 13](#_Toc480879941)

# Introduction:

The method of identifying the domain of the problem along with the scope of the project is known as Analysis. The analysis consists of the use cases, functional and non- functional requirements, prioritization and lastly the architecture of the system. Analysis is performed before the system is being developed because analysis helps to understand how the system should be and what the expectation from the system is. Analysis also helps to discover, analyse, define and also the document of the requirement needed to develop the system. Analyse interacts with the client regularly to make the system as effective as requirement of the clients demand. After the analysis is completed, then the developer and the clients have the overall idea of how the system will work.

As stated above analysis is carried in order to understand the system development and its purpose. Analysis is done through the following activities:

* Use case  
  Use case helps to define a real world actors interacting in the system. It gives the relationships between the use cases, actors and the system. Use case deals with only functional requirements used in the system.
* Prioritization  
  As per the requirements, the system development contains variety of requirements. So the requirements that are organised as per the need of the users is done through the help of the prioritization. In short the classification of the requirements of the system on the basis of the priority is known as prioritization.
* Requirements

The needs or the expectation of the users/ clients while building the system is known as the requirements. Basically there are two types of requirements i.e. Functional requirements and Non-Functional requirements. Functional requirements defines what the system should be able to do and the non-functional requirements defines how the system works.

* Architecture  
  The process of arranging all the variety of components of the system is known as the architecture. In development of this system n-tier architecture is used.

Analysis document consists of many sections which are given in the following order. The use case diagram in section 2, requirements in section 3 which is classified s functional and non- functional requirements respectively, prioritization in section 4 and lastly section 5 consists of the system architecture and the class diagram. Lastly the conclusion concludes the analysis document.

# Use case:

Use case diagram is the series of sequence of actions which describes how the users will perform the different task in the developed system. Alternatively, the use case defines a way in which the real world actor cooperated with the system. Use case is presented as sequences of simple steps which starts with the user’s goal and finished when the goal is achieved as required by users. Use case diagram is the graphical presentation of the system and how it works by interacting between the system and users (actors). The diagram is built in the analysis section because it explains, identify and organize the necessities of the systems.

Use cases is used because it gives following benefits:

* It helps to trace the development method is moving in right direction.
* It sorts out the complexity of the project by breaking the problem into the functions in the system.
* It helps to show how the system works from its graphical presentation.

Use case are basically called behavioural or structural diagram which are used to define set of the action performed (use case) on that system (subject) can do association with more than one external users of the system (actors).

The use case diagram of the proposed system is shown below along with the description of all functions/use cases.

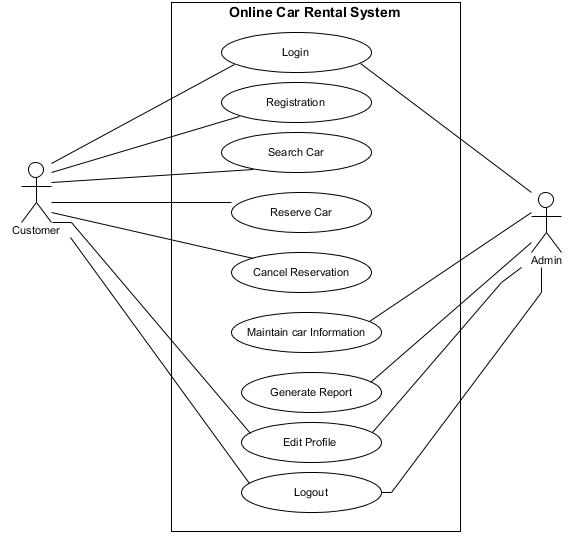


Figure 1: Use Case Diagram

The explanation of the function “Registration” is described below:

* This function is only accessible for customer. At first, customer open the registration form.
* Customer inputs all the details like name, username, password, address and so on in the registration form.
* Customer clicks the ‘Register’ button.
* System validate whether the form is correctly filled or not and also check where the user already exist in the system or not.
* If the validation process is successful, the system displays the successful notification.
* If the validation process is unsuccessful, the system displays the error message and redirect to the registration form again.

The explanation of the function “Login” is described below:

* At first, user opens the login form.
* User inputs their respective username and password in the login form.
* User clicks the ‘Login’ button.
* The system will verify their information, either the user exist or not in the system.
* If the provided information are correct, the system displays the successful notification and authenticate the user to access the system.
* If the provided information are incorrect, the system displays the error notification and redirects back to the login form.

The explanation of the function “Search Car” is described below:

* In search box, user inputs the location and date of pickup and drop off to search.
* System search for the data in the database.
* If the data is available, system displays the suitable car that is available is in that date and location.
* If the respective information is not available, the system displays error message.

The explanation of the function “Reserve Car” is described below:

* Firstly, user search the car available for rent and view their respective details.
* User clicks ‘Rent Car’ button.
* The system displays a form to input the information of reservation such as date, time, and duration to use the car etc.
* User enters all the details and clicks the ‘Submit’ button.

The explanation of the function “Cancel Reservation” is described below:

* User authenticate in the system using their username and password.
* User opens the reservation list.
* User clicks ‘Cancel Reservation’ button.
* The system displays a form to input the reason for cancelling the reservation.
* User enters the cancelling reasons and clicks the ‘Submit’ button.

The explanation of the function “Maintain Car Information” is described below:

* This function is only available for admin. At first, admin login into the system using their username and password.
* User can maintain (add, update, and delete) the information of the car.
* To add the car details, admin clicks ‘Add car information’ button then the system displays a form where admin inputs all the information of the car and click ‘Submit’ button.
* To update car details, admin clicks ‘Update car information’ button then the system displays a form where admin edit the information of the car and click ‘Update button.
* To delete car details, admin clicks ‘Delete’ button then the system displays the confirmation dialogue box and admin clicks ‘Ok’ button if he/she wants to delete otherwise clicks ‘cancel’ button.

The explanation of the function “Generate Report” is described below:

* This function is only available for admin user. Firstly, admin opens the generating report form by clicking ‘Report’ button.
* Admin selects date, type, and period of the car which he /she want to generate report of the details.
* The system shows the report that admin processed, which can be viewed by him/her and other.

The explanation of the function “Edit Profile” is described below:

* This function is available for both admin and customer. Firstly, admin opens the edit profile form by clicking ‘Profile button.
* User edit the field that they want to and click ‘Update Profile’ button.
* If all the fields of the form is properly field, system updates the data and display success message otherwise it will display error message.

The explanation of the function “Logout” is described below:

* User clicks ‘Logout’ button.
* System clears all session.
* System redirects user to the login page.

# Requirements:

Requirements is the process of identifying necessary features that the system must have in order to fulfil clients demand. Requirements analysis is the most important aspect in development of the project. The requirement analysis have many importance in the project managements which are given as:

* It helps to identify the functional requirements and non-functional requirements of system.
* It helps to identify the possible requirements of the users in system.
* It gives the time estimation in building the system.
* The changes that are needed to change as required by the users can be changed by providing the solution from requirements analysis.
* It helps to prioritize the requirements as per the user’s preference.

For analysing and gathering the requirements for the system there were different process involved with the clients. The first and most process was interviewing with the client. After the interview was conducted then the system was built along with the two types of the requirements which are as:

## Functional requirement:

It is a function or feature that must be required in the system and describes what the system should do. It manages what the system ought to do for user. Functional requirements incorporate explanation of the essential functions, details of facts to be alleged in the system.

(Sqa.org.uk, 2017)

<https://www.sqa.org.uk/e-learning/SDM03CD/page_02.htm>

Below description show the functional requirements of the proposed system:

* Login (FR1)

|  |  |
| --- | --- |
| Description | Existing user have to use their respective username and password to access the system. |
| Rational | To sustain user security and privacy |
| Dependencies | FR2 |

* Registration (FR2)

|  |  |
| --- | --- |
| Description | Customer should register their details in order to access the system. The customer must provide details like name, address, username, password etc. |
| Rational | To find new user and their login information. |
| Dependencies | N/A |

* Search (FR3)

|  |  |
| --- | --- |
| Description | Customer should be able to retrieve detailed information of the car that he/she wants to rent by the entering the pickup date, drop off date and location. |
| Rational | To get information fast which saves user time. |
| Dependencies | FR1 |

* Reserve Car (FR4)

|  |  |
| --- | --- |
| Description | Customer should be able to reserve car by providing information about reserving the like date, time, duration etc. |
| Rational | To save the details in the database and to save customer time by reserving from their area online via arriving to the office. |
| Dependencies | FR1 |

* Cancel Reservation (FR5)

|  |  |
| --- | --- |
| Description | Customer should be able to cancel their reservation and remove their details from the database. Customer have to select respective reserve car which they want to cancel. |
| Rational | To remove reservation details from the database and make the car available for other customer. |
| Dependencies | FR1, FR4 |

* Add Car Information (FR6)

|  |  |
| --- | --- |
| Description | Admin should be able to add car details like car model, brand, price, color etc which is later available for the customer to make their reservation. |
| Rational | To save the details in the database and to make details available for the customer. |
| Dependencies | FR1 |

* Update Car Information (FR7)

|  |  |
| --- | --- |
| Description | Admin should be able to change the wrong information of the car. |
| Rational | To store cars correct information |
| Dependencies | FR1, FR6 |

* Delete Car Information (FR8)

|  |  |
| --- | --- |
| Description | The admin should be able to remove the car details permanently from the database in order to make visible for customer. |
| Rational | To remove car details from the database |
| Dependencies | FR1, FR6 |

* Generate Report (FR9)

|  |  |
| --- | --- |
| Description | The admin should be able to create the report by selecting period, type etc. |
| Rational | To view the status of the car. |
| Dependencies | FR1, FR4, FR6 |

* Edit Profile (FR10)

|  |  |
| --- | --- |
| Description | The user should be able to edit their profile such as changing email, password etc. |
| Rational | To correct any inconsistent data of the user. |
| Dependencies | FR1 |

## Non-functional requirement:

It defines how the system works. It basically explicit how the system should act and that is constraint upon the system activities. It is also a worth attributes for a system.

(Reqtest.com, 2017)

<http://reqtest.com/requirements-blog/functional-vs-non-functional-requirements/>

Below description show the non-functional requirements of the proposed system:

* Performance (NF1)

|  |  |
| --- | --- |
| Description | The system must not slack and finish all the actions like adding, updating, erasing details within 2-3 seconds each time the user requests such actions. |
| Rational | To protect user time and complete works within short time. |
| Dependencies | FR1, FR4, FR5, FR6, FR7, FR8 |

* Easy user interface (NF2)

|  |  |
| --- | --- |
| Description | The system is easy, simple so that any user can use the system without any kind knowledge of the application and can see system manual as a help. |
| Rational | To make communication easy for the users as well as to increases user numbers. |
| Dependencies | FR1, FR2, FR3, FR4, FR5, FR6, FR7, FR8 |

* Safe login technique (NF3)

|  |  |
| --- | --- |
| Description | The system can only be accessible for those users having correct username and password which is already store in the system database. |
| Rational | To sustain user security and provide authentication for valid user only. |
| Dependencies | FR1, FR2. |

Prioritization:

The process of categorizing the requirements on the basis of the priority is known as the prioritization. It is done to give the priority for the first steps while developing the system then the other, at first the highest priority is implemented and so on the lower priorities at last. The prioritization is done in the project due to the following points:

* To prioritise the main requirement for developing the system.
* To implement ideal sets of software requirement for implementing the system.
* To give the profit of business as against its cost.
* To make customer more involving and building more customer.
* To obtain the technical benefit along the improvising the market opportunity.

MoSCoW prioritization is one of the best methods to be implemented when undergoing requirement analysis. MoSCoW prioritization technique is used for prioritizing requirements in a collective way by the developer and users/clients. MoSCoW is short for **M**ust have, **S**hould have, **C**ould have and **W**ould/**W**on’t have. The clarification of MoSCoW prioritization is given as:

* Must (M) have:

Explains the essential requirements of the system that has to be fulfilled for the final solution to be acceptable.

* Should(S) have:

Explains high priority requirements that is important but not vital.

* Could(C) have:

Explains the desired or wanted but less important requirements. There will be less impact if they are left out also.

* Won’t or Would(W) have:

Explains the requirements that the project team and stakeholders have decided it will not be implemented.

The prioritizing table of the requirements is shown in below table:

|  |  |
| --- | --- |
| **Requirements** | **MoSCoW** |
| FR1. Login. | Must |
| FR2. Registration | Must |
| FR3. Search car | Must |
| FR4. Reserve car | Must |
| FR5. Cancel Reservation | Must |
| FR6. Add car information | Must |
| R6. FR7. Update car information | Must |
| FR8. Delete car information | Must |
| FR9. Generate report | Must |
| FR10. Edit Profile | Should |
| NF1. Performance | Should |
| NF2. Easy user interface. | Could |
| NF3. Safe login technique | Should |

Table: Prioritization of the requirements

# Architecture:

Architecture is defined as the framework that display the behaviour, structure and view of the system. It helps to change the system along with the maintaining the system. The construction of the system architecture is vital while building the successful system because of the following reasons:

* It helps to monitor the system along the required changes to change.
* It helps in maintain the system.
* It helps to increase the security of the system.
* It helps to repair and replace the system easily.

Inn case of this system build, the architecture shows the system architecture and the initial class diagram which are described as follow:

## System Architecture:

System Architecture is the identification of how a system is operated and structured. It explains the skeletal layout of the system being developed. System architecture takes the responsibility of the system behaviour before being developed and after the system is being developed. System architecture is developed to make the development of system easier along with the maintaining the system to be more effective.

For development of the system, we have used two main tiers architecture. The first tier architecture is concerned with the user involvement and interaction and all the services that is provided by the system to the user. The second tier is the most important tier as it is concerned with the database of the system which handles the data related with the services provided by the system. These two tiers gets along with each other to form a complete effective system which led to business need along the clients. At first the user interacts with the first tier to input the values into the application and the application interact with database for performing the CRUD operation as users demand. Full processing is carried to the clients by the help of two tiers.

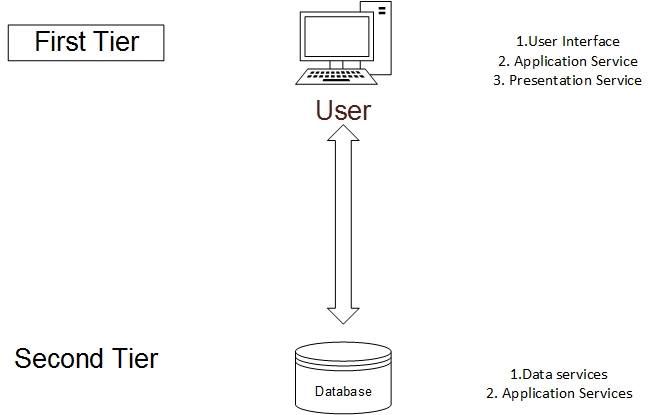


Fig : System Architecture

## Initial Class Diagram:

Class diagram is the unified modelling language structured diagram which defines the architecture of the system by showing the systems attributes, methods, classes and the relationship between the objects involved in the class diagram.

In this scenario, the initial class diagram is built for the system that is going to be build which is shown as:

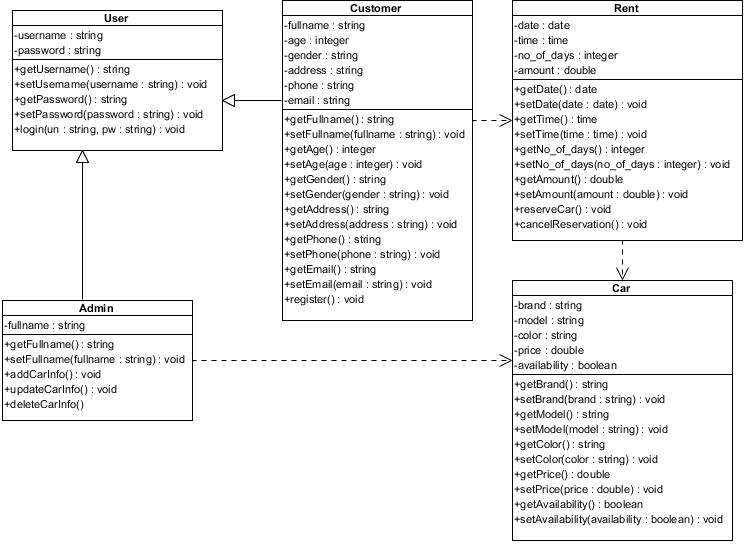


Fig: Initial class diagram

# Conclusion:

After making all the documents and different scenarios in the above work we can conclude that the analysis is successfully completed for this systems project. In this document the functional requirement and non-functional requirements were identified along their prioritization and lastly the system architecture was built for this project. To give the better understanding of the project the use case diagram and the initial class diagram is build. All this analysis done in this document helps to build the system successfully along with effectively.

# References:

Msdn.microsoft.com. (2017). UML Use Case Diagrams: Guidelines. [online] Available at: [https://msdn.microsoft.com/en-us/library/dd409432.aspx](https://msdn.microsoft.com/en-us/library/dd409432.aspx%20) [Accessed 20 Apr. 2017].

SearchSoftwareQuality. (2017). What is requirements analysis (requirements engineering) ? - Definition from WhatIs.com. [online] Available at: [http://searchsoftwarequality.techtarget.com/definition/requirements-analysis](http://searchsoftwarequality.techtarget.com/definition/requirements-analysis%20) [Accessed 20 Apr. 2017].

Sqa.org.uk. (2017). Functional and Non-Functional Requirements. [online] Available at: <https://www.sqa.org.uk/e-learning/SDM03CD/page_02.htm> [Accessed 20 Apr. 2017].

Reqtest.com. (2017). Cite a Website - Cite This For Me. [online] Available at: <http://reqtest.com/requirements-blog/functional-vs-non-functional-requirements/> [Accessed 21 Apr. 2017].