**RENT A CAR**

**Rent or Reserve a car (Car bookings)**

**Team Members:**

Muhammad Bilal (CS091182051)

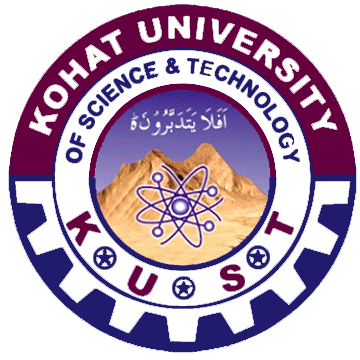
Muhammad Haroon (CS091182032)

**Supervisor**

Mr. Meer Wali Ur Rehman Khan, Lecturer

Institute of Computing

KUST, Hangu Campus.

****

**Institute of Computing**

**Kohat University of Science and Technology, Kohat-26000**

**Khyber Pakhtunkhwa, Pakistan**

**(August 2022)**

**RENT A CAR**

**Rent or Reserve a car (Car booking)**

Muhammad Bilal (CS091182051)

Muhammad Haroon (CS091182032)

A thesis submitted in partial fulfilment of the requirements for the degree of BSCS.

Thesis Supervisor:

**Mr. Meer Wali Ur Rehman Khan**

Team Leader Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Thesis Supervisor’s Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

FYP Coordinator’s Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Director’s Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Institute of Computing

Kohat University of Science and Technology, Kohat-26000

Khyber Pakhtunkhwa, Pakistan

# Declaration

We certify that this project titled “***RENT A CAR****”* is our work. The work has not been presented elsewhere for assessment. The material that has been used from other sources has been properly acknowledged/referred to.

Muhammad Bilal (CS091182051) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Muhammad Haroon (CS091182032) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Plagiarism Certificate (Turn in Report)

This thesis has been checked for Plagiarism. Turnitin report endorsed by Supervisor is attached at the end of the thesis.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Student’s signatures

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Supervisor signature

# Copyright Statement

* Copyright in the text of this thesis rests with the student author. Copies (by any process) either in full or of extracts may be made only following instructions given by the author and lodged in the Library of KUST. Details may be obtained by the Librarian. This page must form part of any such copies made. Further copies (by any process) may not be made without the permission (in writing) of the author.
* The ownership of any intellectual property rights which may be described in this thesis is vested in the Institute of Computing, KUST, subject to any prior agreement to the contrary, and may not be made available for use by third parties without the written permission of the institute, which will prescribe the terms and conditions of any such agreement.
* Further information on the conditions under which disclosures and exploitation may take place is available from the Library of KUST, Kohat.

# Acknowledgements

All acclamation and appreciations for ALMIGHTY ALLAH, who bestowed us with sight to observe, mind to think, judge and power to achieve something. Peace and blessing of Allah be upon the Holy Prophet and his pure and pious progeny. Any project cannot be completed without the help and efforts of a lot of people. We extend our appreciation and gratitude to our parents they encourage us and boosted our morale at every stage of the project.

We feel immense pleasure to express our cordial gratitude to our project supervisor Mr Meer Wali Ur Rehman Khan (Lecturer in Kohat University of science and technology for his keen interest, critical insight, valuable technical guidance, constant source very kind supervision of our work (noble venture). We are thankful to all teachers & friends who helped us in bringing the project from concept to reality.

# DEDICATION

This humble effort is dedicated to our **Parents**, relatives, advisor, friends, and family for their extreme support, love, and care who enable us to achieve the goal of success.

Muhammad Bilal (CS091182051)

Muhammad Haroon (CS091182032)

# Abstract

This application presents a data management system for a car rental company. This enables the administrator to keep track of all the customers’ information. This system increases customer retention and simplify vehicle and customers management in efficient way. The car rental management system has a very user-friendly interface. Thus, the users will feel very easy to work on it. By using this system admin can manage their rental, bookings, and customer issues and vehicle issues etc. The car rental information can be added to the system, or existing information can be edited or deleted by the administrator. The transaction reports of the car rental system can be retrieved by the administrator, when it’s required. Thus, there is no delay in the availability of any car information, when ever needed the car rental information can be captured very quickly and easily.

**Key Words:** *car, car booking, booking, reserve, rent, rent a car, fyp.*

**Table of contents**

Contents

[Declaration i](#_Toc100880770)

[Plagiarism Certificate (Turn in Report) ii](#_Toc100880771)

[Copyright Statement iii](#_Toc100880772)

[Acknowledgements iv](#_Toc100880773)

[DEDICATION v](#_Toc100880774)

[Abstract vi](#_Toc100880775)

[CHAPTER 1: INTRODUCTION 11](#_Toc100880776)

[INTRODUCTION 12](#_Toc100880777)

[PROJECT DEFINITION 12](#_Toc100880778)

[PROBLEM STATEMENT 12](#_Toc100880779)

[OBJECTIVES 13](#_Toc100880780)

[PROPOSED SOLUTION 14](#_Toc100880781)

[MOTIVATION 15](#_Toc100880782)

[SCOPE OF THE PROJECT 15](#_Toc100880783)

[CHAPTER 2: BACKGROUND / LITERATURE 16](#_Toc100880784)

[EXECUTIVE SUMMARY 17](#_Toc100880785)

[INTRODUCTION 17](#_Toc100880786)

[SIMILAR EXISTING SOFTWARE APPLICATIONS 17](#_Toc100880787)

[PROBLEMS WITH EXISTING SOFTWARE 18](#_Toc100880788)

[PROPOSED SOLUTION FOR PROBLEMS 18](#_Toc100880789)

[CONCLUSION 18](#_Toc100880790)

[SDLC Software development life cycle: 19](#_Toc100880791)

[AGILE development model: 20](#_Toc100880792)

[CHAPTER 3: REQUIREMENTS 22](#_Toc100880793)

[REQUIREMENTS GATHERING 23](#_Toc100880794)

[FUNCTIONAL REQUIREMENTS 23](#_Toc100880795)

[**1.** **Content (VEHICLES) being registered by admin** 23](#_Toc100880796)

[**2.** **Operations performed by each screen** 23](#_Toc100880797)

[**3.** **Content approval and workflow process** 24](#_Toc100880798)

[NON-FUNCTIONAL REQUIREMENTS 24](#_Toc100880799)

[**1.** **System Performance** 24](#_Toc100880800)

[**2.** **Scalability** 24](#_Toc100880801)

[**3.** **Availability** 25](#_Toc100880802)

[**4.** **Security** 25](#_Toc100880803)

[**5.** **User-friendly Interface** 25](#_Toc100880804)

[METHODS OF REQUIREMENTS GATHERING 25](#_Toc100880805)

[**1.** **Interviews** 25](#_Toc100880806)

[**2.** **Surveys** 26](#_Toc100880807)

[**3.** **Questionnaires** 26](#_Toc100880808)

[**4.** **Brainstorming** 26](#_Toc100880809)

[**5.** **Prototyping** 26](#_Toc100880810)

[CHAPTER 4: DESIGN AND ANALYSIS 27](#_Toc100880811)

[DESIGN PHASE 28](#_Toc100880812)

[**1)** **Components involve** 28](#_Toc100880813)

[**2)** **Third-party packages** 28](#_Toc100880814)

[**3)** **User flow** 28](#_Toc100880815)

[**1)** **ERD (Entity Relationship Diagram)** 29](#_Toc100880816)

[**2)** **DFD (Data flow diagram)** 29](#_Toc100880817)

[**3)** **Activity diagram** 32](#_Toc100880818)

[Use case diagram 33](#_Toc100880819)

[**4)** **Components interaction with DB** 34](#_Toc100880820)

[ANALYSIS 34](#_Toc100880821)

[**1)** **Gather requirements** 34](#_Toc100880822)

[**2)** **Analyze gather requirements** 34](#_Toc100880823)

**List of Figures**

**Fig 1:** Proposed solution architecture …………………………………………………. 04

**Fig 2:** Demonstration of SDLC (Agile model) ………………………………………………... 10

**Fig 3:** RENT A CAR ERD (Entity relationship diagram) ….…………………………............. 20

**Fig 4:** Level Zero (0) of RENT A CAR DFD (overview) …………...…………… …….…… 21

**Fig 4.1:** Level 1 of RENT A CAR DFD (all the entities) ………...………… ……………….… 21

**Fig 4.2:** Level 2 of RENT A CAR DFD (User registration) …… ………………………….… 22

**Fig 4.3:**  Level 3 of RENT A CAR DFD (A user request for Idea or project) ……….……….…. 22

**Fig 4.4:** Level 4 of RENT A CAR DFD (Adding Comments) …………... ………………….… 23

**Fig 5:** RENT A CAR Activity diagram ………………..….…….………....…..……………… 23

# CHAPTER 1: INTRODUCTION

# INTRODUCTION

RENT A CAR (RAC) is a web based system for a company that rents out cars. This system enables the company to make their services available to the public through the internet and also keep records about their services. The world has become a place where there is a lot of technological development; where every single thing done physically has been transformed into computerized form. Nowadays, people’s activities have been transformed into work done by computerized systems. One of which is the main target of this project which is about Car Rental System. The system of renting cars exist back in the previous years, were people rent cars for their personal reasons. Car renting is essential to many peoples’ plan to travel or move from one place to another for business purposes, tour, and visit or holidays, for these reasons Car renting is very helpful.

# PROJECT DEFINITION

As the internet improves the life of people, it also gives access to things that were inaccessible before. The internet is one of the most important tools of communication. The world has become a place where there is a lot of technological development which brings the result of almost every single thing has been transformed into computerized form. These days, individual activities have been changed into work done by information systems. One of which is the primary objective of this project which is about car rental management system. Renting car system exists in the previous years where people rent cars for their own reasons. Car rentals is basic to numerous individuals’ arrangement to travel or move from one place to another for business purposes, tour, and visit occasions. Thus, car rental is extremely useful.

Our RENT A CAR (car rental management system) is a web-based system for an organization that rents out cars. This system empowers the organization to make their services accessible to the public through the web and furthermore keep records about their services.

# PROBLEM STATEMENT

The problem with some of the current system is that:

Based on observations, some small companies already have a car rental system which is not a web based application. This is a limitation that gives them capability to store customer’s details, but at the same time they cannot make their services more available to the public through the internet, they rather make use of posters to advertise their services to the public. These types of companies can overcome these problems by switching to the web base application of their type of system.

They also make use of phone call reservations which is also limited to many features as compare to a web base system. For example a customer may make a phone call reservation for a particular car, but when he/she comes to pick the car, he/she might turn not to like the car; this could be because the customer could not see a sample picture of the car he/she wants to rent. Some other problems with the manual renting a car are as follows:

* To rent a car, a prospective renter must first go to the nearest office to register as a client, what of if the customer doesn’t have enough time to do that?
* Cars that provide difficulties to rent out are normally advertised in local or national newspaper. It involves a lot of paper work and consumes time.
* The process of managing customer’s data is slow if the company is using manual system and there might be thousands of clients.
* It is very hard to keep record of all rental cars and so on.

# OBJECTIVES

The main objectives of this project are:

* To develop a web based system that will help manage the business transactions of car renting.
* To help in advertising the car rental services of a company, through the availability of the system online. Development of their proposed idea around the Software development Life Cycle (SDLC).
* To develop a simple and secure system that protects client information and confidential information of the organization
* To design a user-friendly system that enables client check for availability of vehicle and book or reserve a vehicle online.
* To design a system that enables clients pay their car rent online
* To develop a system that stores bookings and reservations information as well as payment history to help the organization keep track of transactions.

# PROPOSED SOLUTION

The proposed solution to the above-mentioned problems is to develop a web based system that will help manage the business transactions of car renting that will help clients or customers check for availability of vehicle and book or reserve a vehicle online. That will make sure that customer are capable of reserving car from the comfort of their home and don’t need to physically go to bargains for reserving cars for their next journey. Businesses will boost their revenue by collecting their car rents online at the time of car booking. While using manual system it’s very difficult to keep track of all the transactions and it’s difficult to manage customer’s data so this proposed system will keep track of all the transaction and manage the customer’s data automatically.

Figure 1. Proposed solution architecture

# MOTIVATION

In consideration with historical and rapid development of car rental companies, the way processes in the companies are taking place today which is quite problematic, this project is planned to ease those processes through developing an effective and efficient car rental system, just like other developed countries are using technologies towards facilitating their customers processes through projects. There are problems with already existing similar systems so keeping in mind those problems and constraints we are here with a plan to bring ease and comfort regarding those issues. This will help people not to go physically for renting a car for their next journey. In similar existing system there are no way of keeping transaction history, keep track of their customers and clients, keep their business up to date etc. So this web service will try to overcome those problems.

# SCOPE OF THE PROJECT

Over time, the scope of this project will expand and it will provide different renting car associated services that will cover lots of things. RENT A CAR will try their best to assist people in a wide variety. With time, its services will expands because somehow this will be in use by everyone related to vehicle bargains. Someone who wants to reserve a car for their business journey, for their road trip, or for visiting somewhere from the comfort of their home, this web service will help them accomplish their task very easily. This will help people with cars or people who have bargains to commercialize their work and put their business on cloud that will be accessible by everyone having active internet connection and will be available to anyone by 24/7 hours. Bargain owners will no need to advertise their shops by putting large banners on street walls or showing them on large boards present on roads, in fact once a customer visit this site he/she will be happy using this and will tell their known peoples to use this. Because its time saving, easy, convenient and useful to be used. In short words this web service will be used by customers/clients who wants to have a car for their upcoming journey, this will be used by those bargains who wants to keep track of their cars renting, customers and transactions. This will be used by bargains who wants to solve their renting issue and customers issue without calling customer to be physically appear and so on.

# CHAPTER 2: BACKGROUND / LITERATURE

# ****EXECUTIVE SUMMARY****

Before RENT A CAR (web service), problems were addressed to different platforms but the overall experience of the bargain owners and people who have habit of reserving car for their journey, was bad. People were facing lots of problems but there was no such tool exists that could efficiently solve those problems. People who wants to reserve car need to be go physically to the nearest bargains, which were time consuming and of course expensive. Where they need to wait for their turn and after need to be provide all of his/her details at the reception disk. This was also difficult for vehicles owners to keep track of their customer activities like transactions, renting etc. In short people were facing problems and issues regarding renting cars (vehicles) and bargain/vehicles owners were facing problems regarding their businesses. Now this web service will put their business on cloud and it will bring ease to both the clients and owners.

# ****INTRODUCTION****

After RENT A CAR was officially introduced and launched, we assert that there will be no issue for those who have before RENT A CAR. Because it will provide services in a wide range and will try to expand more in future. It will bring lot of ease in the life of many people. Before RENT A CAR people were using traditional way to reserve a car that will be needed for their upcoming business tour, or something else. Now people who wants to take a car on rent have a way to do it while staying at their home (no need to physically go and waste time and money). Now it’s a very convenient way of taking a car on rent because it can be done through using your smartphone or PC. All you need is to have an active internet connection and a smartphone. Reserving a car is now few taps away you need to take your smartphone make an account on RENT A CAR and reserve a car for yourself. .

# ****SIMILAR EXISTING SOFTWARE APPLICATIONS****

The following software products may have similar services with RENT A CAR.

1. Car Rental Service
2. Let’s Reserve
3. Maybe other web services.

# ****PROBLEMS WITH EXISTING SOFTWARE****

As Google becomes a very popular search engine it is the number one search engine used by people. And it is difficult that on Google you are unable to find something related to education. The idea here is that maybe the mentioned platforms are related to RENT A CAR and maybe they are also somehow benefiting people with the services that RENT A CAR wants to. But the thing is how friendly they are, does people with a poor understanding of using the internet getting advantage of it? Here they fail and RENT A CAR comes in. This web portal has a very friendly user interface and people who are not from IT background can easily navigate through this website. In future, team will also try their best to put AI concept to make RENT A CAR capable of taking decisions for their clients.

# ****PROPOSED SOLUTION FOR PROBLEMS****

RENT A CAR is here to solve those problem being faced by people at the time of car reservation. This will help clients or customers as well as the bargains owners. Both the parties will get advantage of it. As it will save time and money of the customers who are willing to use rent cars for their journey. While it will boost bargains productivity by putting their business live for 24/7 hours. As their business will become online (computerized) handled by machines and we know machine do not get tired.

# ****CONCLUSION****

The study reveals that all the variables (Customers, Clients and Bargains) are the core value of RENT A CAR. So it will try their best to make sure that the following

* + - 1. Brand trust
      2. Customer satisfaction and
      3. Brand loyalty

Are high, meaning that consumers have already believed and are satisfied with the website and services and are loyal to RENT A CAR.

Besides, it appears that brand trust and customer satisfaction may directly influence brand loyalty.

# SDLC Software development life cycle:

SDLC or the Software Development Life Cycle is a process that helps us produces software that has the highest quality and lowest cost in the shortest time possible. SDLC provides a structured flow of phases that help an organization to quickly produce software that has high quality.

The SDLC involves six phases as given below

* Requirement gathering or analyzing
* Planning
* Design
* Software development (Actual coding)
* Testing (Unit and Integrated testing)
* Deployment (Launching)

There are lots of SDLC models used but some of them become outdated for nowadays use because nowadays customers want products that can adopt changes. Following are the well-known SDLC models.

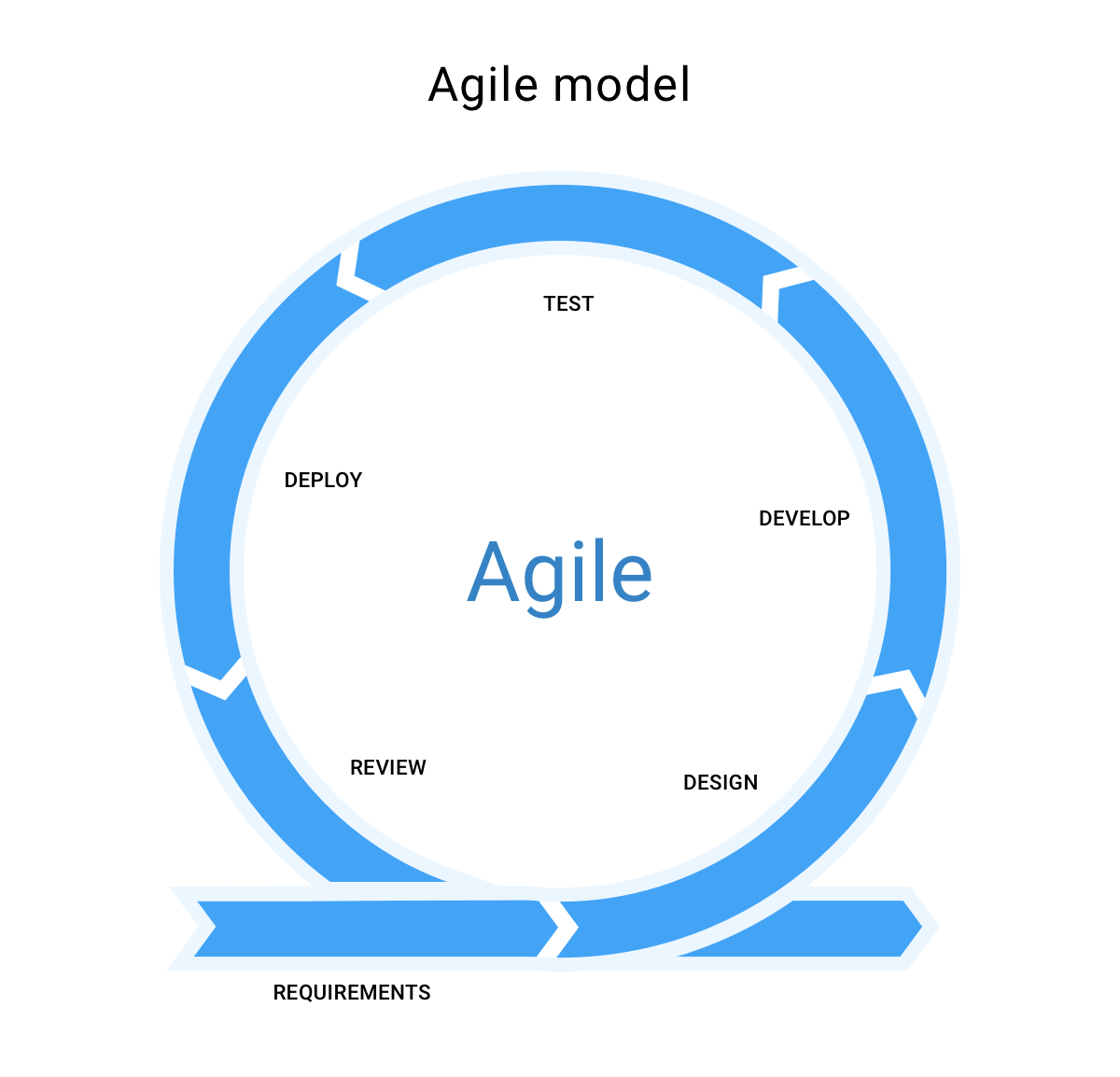
* Waterfall model
* Incremental model
* Spiral model
* Agile model

In earlier life developers were using the waterfall model for developing software but it was not flexible and do not provide flexibility for developers. And nowadays developers face some problems using the waterfall model which is handling changes requests from the customer during development and the high cost and time required to make those changes. We don’t want to face these issues thus we decided to use the agile model for our project RENT A CAR.

# AGILE development model:

The agile model was primarily designed to help a project adapt to change requests. The main aim of the agile model is to make the project able to adapt to changes during the development phase. In the agile model, we remove all those activities which do not important for that project. So by doing this we save lots of time and effort. The agile model decomposes the project into iterations (time-boxes) where each iteration is completed separately and after planning, development, and deploying hands over to the customer. Just like other models AGILE model also consists of the following

* Requirements
* Planning
* Design
* Development (Actual coding)
* Testing (Unit and Integrated testing)
* Deployment (Launching)
* Review

Following is a picture was taken to demonstrate the functionality of the Agile model (SDLC).

# CHAPTER 3: REQUIREMENTS

# ****REQUIREMENTS GATHERING****

Requirements gathering is one of the most essential parts of any project and adds value to a project on multiple levels. Requirements gathering is easier said than done, it is generally an area that is given far less attention than it needs. Requirements gathering is also an exploratory process that involves researching and documenting the project’s exact requirements from start to finish. Effective requirements gathering and requirements management start at the beginning of the project.

# ****FUNCTIONAL REQUIREMENTS****

Functional requirements refer to the need and working of the project. As we are building a web application titled (RENT A CAR) here the functional requirements will be as to how the user will interact with (RENT A CAR). In the case of this project, our functional requirements may include the following.

## **Content (VEHICLES) being registered by admin**

Our first functional requirement will be how the system will respond to and accept the content (vehicles) being registered by admin. As there will be different types of vehicles, so the first focus will be on how the system will respond to these publications and it should happily accept things like these.

## **Operations performed by each screen**

The second thing that we need to put focus on is how each screen will perform its operation(s). Like if we have a registration page that will be used by new users/customer for registrations so we have to make sure that, that page works fine for getting user/customer details as a new user will provide. And same goes for every page and screen that will properly work for their respective task.

## **Content approval and workflow process**

As contents are everywhere and it is very important to have valid content. If you have lots of content present on your portal but they don’t have any valid meaning and do not help that portal to deliver its services effectively then it is very necessary to check them again and replace them with the specified contents. To accomplish that, RENT A CAR should also have the functionality of checking and verifying contents being provided by the users. RENT A CAR will approve those contents which are valid and have meaning to RENT A CAR that will improve this web service and will help provide services effectively. RENT A CAR should also focus on the workflow of the system like a user/customer cannot reserve a car if they aren’t registered. Or user/customer cannot reserve a car if it is already being reserved Etc.

# ****NON-FUNCTIONAL REQUIREMENTS****

Non-functional requirements refer to the conditions rather than specific behaviours. It is also known as the “quality attributes” of a system. While functional requirements are for what the system will do, non-functional requirements are for how the system will do that. Non-functional requirements are not that much necessary for a project but surely it adds more quality to a project and helps in making the workflow lots smoother and effective. Non-functional requirements may include:

## **System Performance**

Performance refers to how fast the system is while responding to a user request. In our case performance will be the time a page is taking while loading. You may not include this feature of being fast, it will work fine for the basic tasks but users love fast processing and it will add more value to your system if it is performing fast.

## **Scalability**

Scalability can be defined as, to grow as the users increase. This means that the system needs to be able of handling a large number of users’ data. And should be scalable if in future time the number of users or data increases. A system is said to be scalable if it is able of changing its behaviours over time and with the changes in data.

## **Availability**

The system availability is necessary it means that the system should be available and awake to provide its services to users. Availability should be as greater as you can and the downtime should be as less as you can. If a system uses to become available for some time and then go to sleep and then come available, the user will become bored and will start leaving that system and go away.

## **Security**

This contains the security of the contents and encryption. Its means that unauthorized user cannot access the content. Security also refers to show different content to different users. As there are many types of users so make sure to provide the user with the content they needed.

## **User-friendly Interface**

Visitors and users like a user-friendly interface where they can easily navigate among different things and features. Nowadays programmers work hard to make the interface also called UI as friendly as they can because everyone wants to have a large number of visitors and visitors love the user-friendly environment. So your system should be enough friendly to treat users and visitors in such a way that they wish to visit your system again and utilize the services.

# ****METHODS OF REQUIREMENTS GATHERING****

Requirements gathering begins with the idea of what and how the system will perform its tasks. As we need to collect requirements from different sources and places so there exists plenty of methods that can be used to gather requirements. Some of the common methods used for gathering functional and non-functional requirements are as follows:

## **Interviews**

The interview is the most common way of knowing the system requirements. Different interviews are held between the user and the client for the sake of knowing what the system is proposed to do. During the interview, the project manager or the programmer asks the clients about the problems they have been facing.

## **Surveys**

Surveys are used to collect information and requirements within a short time frame from a large user group. Surveys on different sites and places are done for collecting user problems and requirements.

## **Questionnaires**

Questionnaires refer to the interviewers, as they ask different types of system-related questions to collect more information. When the interview conduct between the programmers (software experts) and the client (to whom the project is going to be built) then the programmer or the project manager asks the client regarding problems they have been facing and notes down those problems to must-have solutions for these.

## **Brainstorming**

Subject specialists conduct different meetings and sessions to discuss different problems being faced by users before the proposed system and they do brainstorming to bring solutions to those problems and other complex issues. As we discussed earlier that in an interview the client asked different questions about the problems being faced by them but after the developer or the project experts also do brainstorming to add more features from their side as they are professionals and have more knowledge than the clients.

## **Prototyping**

Prototyping refers to making a prototype (not the actual system) of the system so that it will not be the actual system but will look like it and will help you understand the functions and requirements.

# CHAPTER 4: DESIGN AND ANALYSIS

# ****DESIGN PHASE****

During the design phase, developers and technical architects start the high-level design of the software and system to be able to deliver each requirement. The selected architectural design defines what components should be in, what third-party packages will be used, how the user flow will occur and different components communicate with the database along with front-end representation and behaviours of different components etc. In this phase, the gathered requirements document maps the requirements into an architecture. That architecture defines the components, their interfaces, and behaviours. Design deal with the UI (user interface) means that in this phase developers or architects focus on the alignment of different elements and components to make the system more attractive.

## **Components involve**

Components are the building blocks of every software, they may be buttons, headings, links, etc. Designers and developers focus on all the components involved in software, along with their alignments and interaction with one another in the front-end.

## **Third-party packages**

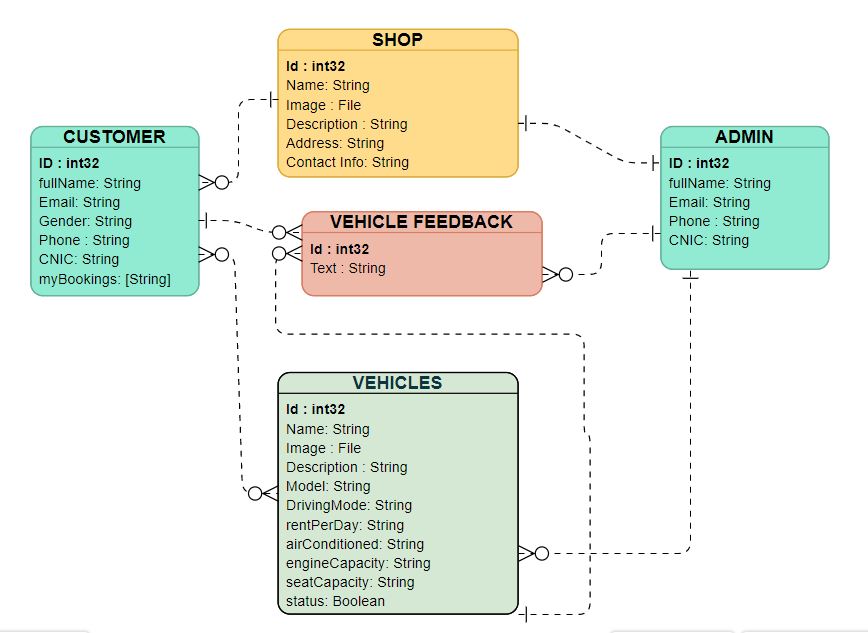
Third-party packages are libraries and packages which have built-in functionalities to improve the software. Just like NODE JS uses express for routing and Mongoose for mongo database are the third-party packages. So it is the time of designing, designers and developers decide which third-party packages and libraries would be used for an upcoming project (software).

## **User flow**

The designer also keep eye on how the user flow will occur. It means first of all how the user will register themselves then what should be the next screen to show when a user got to register. How our application will interact differently with a different type of user etc.

## **ERD (Entity Relationship Diagram)**

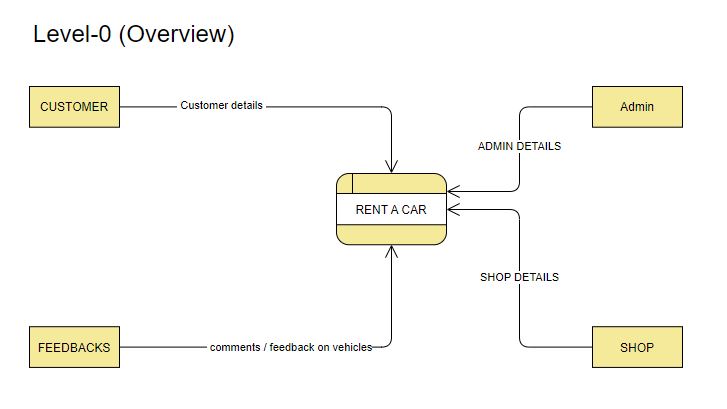
Entity Relationship Diagram, also known as ERD. An ERD consists of different symbols and connectors for their connection those symbols and connectors visualize two important information: **All the important entities within that system**, and their **inter-relationships.**

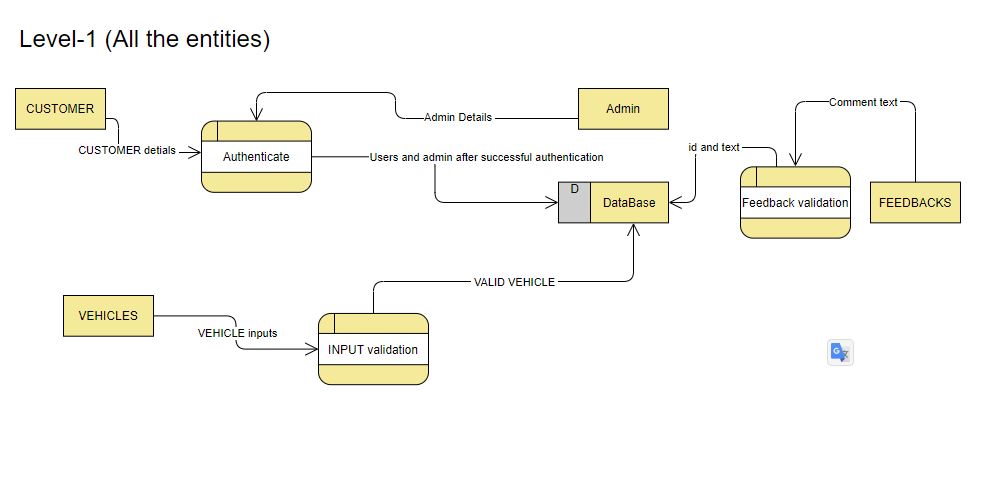
*Following is the* RENT A CAR *ERD.*

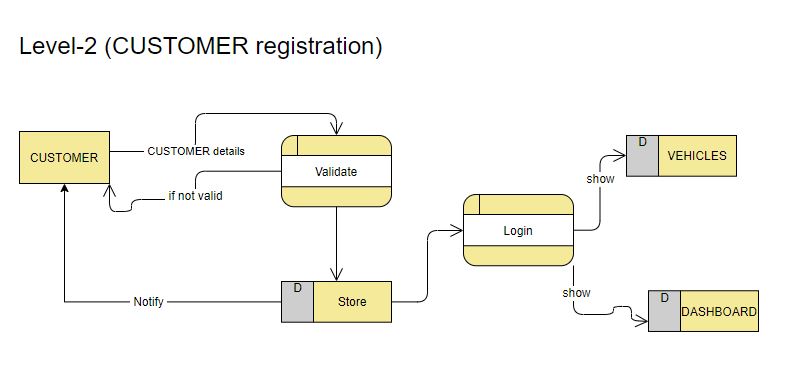
## **DFD (Data flow diagram)**

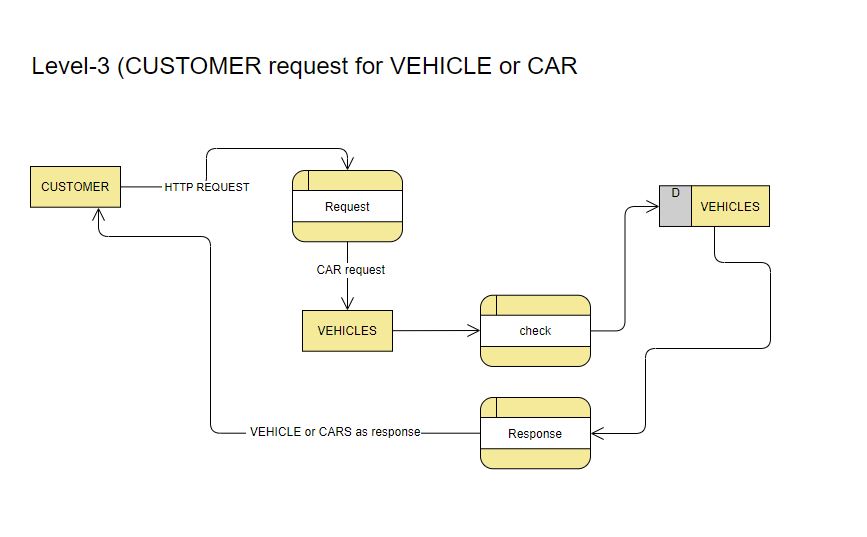
DFD tells us about the flow of data among the different components of the software. It shows us how the data flow occurs and how different components of the system are communicating among themselves. DFD shows us what data these components or elements transfer with each other and how that flow occurs.

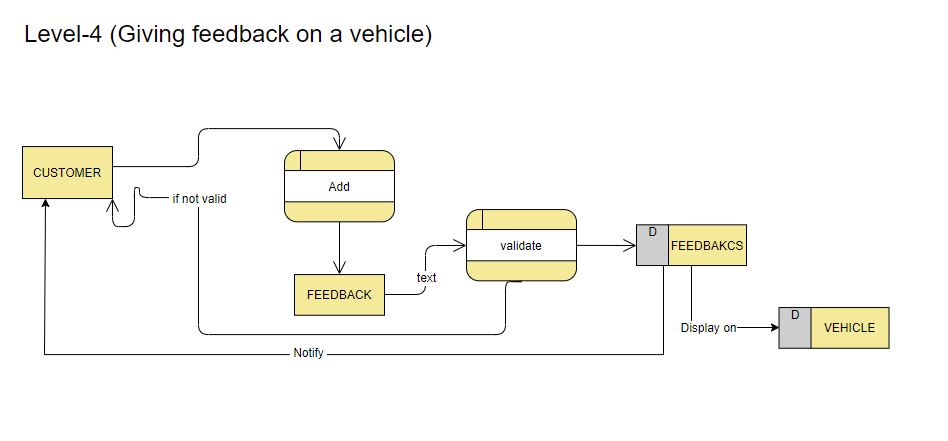
DFD has different levels and as RENT A CAR also have DFD so let’s have a look at the different level of DFD. Following are the different levels of DFD (data flow diagram) of RENT A CAR.









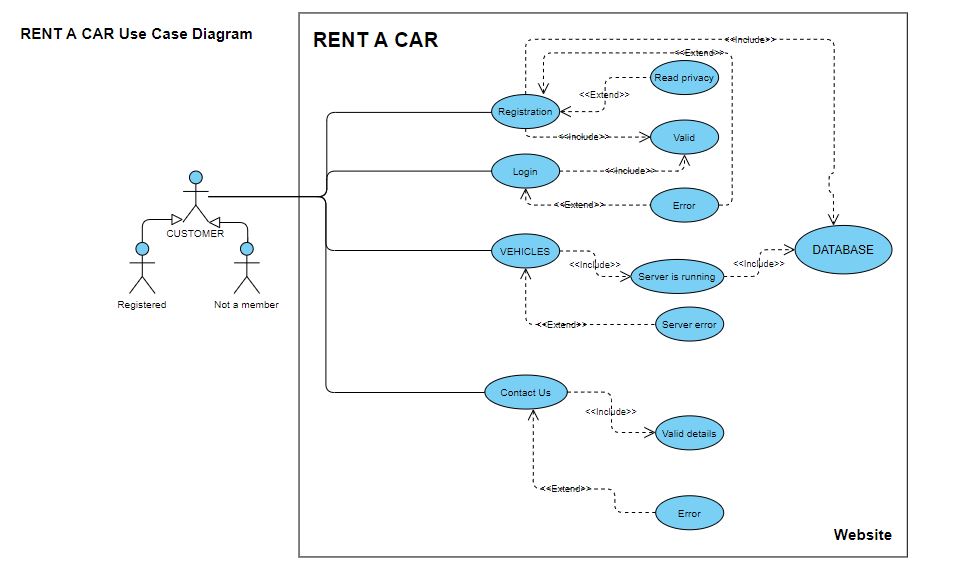


## **Activity diagram**

An activity diagram is used to represent the flow from one activity to another activity inside a system. The activity inside a system is an operation of that system. The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all types of flow control by using different elements.

# Use case diagram

Use a case diagram use to summarize the details of your system's users and tells about their interactions with the system. An effective use case diagram helps your team demonstrate the following.

* Scenarios in which your system interacts with people, or external entities
* Goals that your system helps those entities to achieve etc.
* Below is the Use case diagram of RENT A CAR

## **Components interaction with DB**

The database involves everywhere and it's the thing that keeps data for future use. Databases are the storage that saves application data somewhere on the disk or online. So it's also the responsibility of the designer to show how the application components will communicate and interact with the database, how data will come from the database to the front-end (in front of the user on-screen).

# ****ANALYSIS****

This process is also called discovery, in this phase, we discover and analyze all the requirements and relevant data. Our goal in this phase is to determine the specific ways of how you will get benefits from this application. In the analysis phase, we look after requirements and project expectations. The analysis phase consists of

## **Gather requirements**

We discussed earlier in the requirement phase that the first thing that we need is to analyze both the functional and non-functional requirements. Requirements are the expectation from the project (Application). This means that what the system will do. We also discover more relevant requirements in the analysis phase.

## **Analyze gather requirements**

It is also necessary to analyze and double-check already gather requirements for ambiguity, repetition, vulnerability. The main purpose of the analysis is to transform requirements into high-level requirements specified in the earlier phase into

* 1. Unambiguous: There should be no ambiguity (free of dauts)
  2. Testable: result well when tested.
  3. Complete: There should be no deficiency.
  4. Consistent: Consistency is the key, they should be consistent. Able to change over time mean should be adaptable over time.

In short, the analysis phase is all about gathering more specific requirements through questioners and asking and it’s about gathering information about

1. What should be inside the application (in our case in a web application)
2. What are the goals and objectives (what you want from the site to do for you)
3. What you want to communicate and whom you want to attract (your targeted audience).