

**SCHOOL OF SCIENCE**

**DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCE**

**COMP 416: RESEARCH PROJECT**

**DESIGN AND DEVELOPMENT OF CAR RENTAL SYSTEM**

**SUBMITTED BY:**

**NAME**: NZOMO SAMUEL

**ADM NO**: COM/011/17

**Name of the supervisor:**

Supervisor: DR. KITTUR

**THIS PROJECT DOCUMENTATION IS SUBMITTED TO THE DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCE, UNIVERSITY OF ELDORET (UOE), IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF DEGREE IN BACHELOR OF SCIENCE IN COMPUTER SCIENCE**

# DECLARATION

I certify that this report does not incorporate, without acknowledgement, any material previously submitted for a degree or diploma in any university and to the best of my knowledge and belief, it does not contain any material previously published or written by another person or myself except where due reference is made in the text. I also hereby give consent for my report, if accepted, to be made available for referencing, and for the title and abstract to be made available to any person that wishes to make references.

Signature of Candidate: ………………………. Date: ……………………

Name of Candidate: ………………………………………………………………………………………...

Countersigned by:

Name of Supervisor: …………………………………………………………………….

Signature of Supervisor: ………………………. Date: ………………….

# CERTIFICATION

This is to certify that this Report entitled DESIGN AND DEVELOPMENT OF CAR RENTAL MANAGEMENT SYSTEM, which is submitted by NZOMO SAMUEL in partial fulfilment of the requirement for the award of degree for Bachelor of science in Computer Science to the Department of Mathematics and Computer Science, UNIVERSITY OF ELDORET is a record of the candidate’s own work carried out by the candidate under my own supervision. The matter embodied in this thesis is original and has not been submitted for the award of any other degree within this institution to my own belief and to the best of knowledge.

**APPROVED BY:**

**SUPERVISOR: DR. KITTUR**

**Signature: …………………………………………………………….**

# DEDICATION

This work is dedicated to God Almighty, my strong pillar, my source of inspiration, wisdom, knowledge and understanding. He has been the source of my strength throughout this program.

This work is also dedicated to my parents, my lecturers through my course of study and as well to my supervisor Dr. KITTUR for full support throughout the design and development.

Thank you all for your support. God bless you.

# ABSTRACT

This application presents a data management system for a car rental company where employees simply referred to as users within this system can manage all activities regarding to the system. This enables them to keep track of all the customer’s and car’s information. This system increases customer retention and simplify vehicle and customer’s 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, a user can manage rentals, car returns, 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 user. The transaction/return reports of the car rental system can be retrieved by the user, when its 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.

Table of Contents

[DECLARATION 2](#_Toc81763206)

[CERTIFICATION 3](#_Toc81763207)

[DEDICATION 4](#_Toc81763208)

[ABSTRACT 5](#_Toc81763209)

[CHAPTER 1 9](#_Toc81763210)

[1.0 INTRODUCTION 9](#_Toc81763211)

[1.1 BACKGROUNT AND MOTIVATION 10](#_Toc81763212)

[1.1.0. BACKGROUND STUDY 10](#_Toc81763213)

[1.2 PROBLEM STATEMENT 11](#_Toc81763214)

[1.3 AIMS AND OBJECTIVES 11](#_Toc81763215)

[1.4 JUSTIFICATION 12](#_Toc81763216)

[1.5 SIGNIFICANCE OF THE SYSTEM 13](#_Toc81763217)

[CHAPTER 2 16](#_Toc81763218)

[2.0 LITERATURE REVIEW AND SIMILAR SOLUTIONS 16](#_Toc81763219)

[Introduction. 16](#_Toc81763220)

[Fenam car rental management system 17](#_Toc81763221)

[Suria Car Rental & Tour 17](#_Toc81763222)

[Summary 18](#_Toc81763223)

[CHAPTER 3 19](#_Toc81763224)

[3.0 REQUIREMENTS ANALYSIS AND DESIGN 19](#_Toc81763225)

[3.1 Introduction 19](#_Toc81763226)

[SYSTEM DESIGN 19](#_Toc81763227)

[System Architecture 19](#_Toc81763228)

[Use case Diagram 20](#_Toc81763229)

[DATA FLOW DIAGRAM 21](#_Toc81763230)

[Class Diagram 22](#_Toc81763231)

[Activity Diagram 22](#_Toc81763232)

[Database Design 22](#_Toc81763233)

[3.2 REQUIREMENT GATHERING 23](#_Toc81763234)

[3.2.1 DATA COLLECTION TECHNIQUE 23](#_Toc81763235)

[3.2.2Tools and Techniques 24](#_Toc81763236)

[3.2.3 Requirement Analysis 24](#_Toc81763237)

[CHAPTER 4 27](#_Toc81763238)

[4.0 METHODOLODY 27](#_Toc81763239)

[4.1 INTRODUCTION 27](#_Toc81763240)

[4.2 The Iterative Model 27](#_Toc81763241)

[Chapter 05 31](#_Toc81763242)

[5.0 IMPLEMENTATION 31](#_Toc81763243)

[5.1 IMPLEMENTATION ENVIRONMENT 31](#_Toc81763244)

[5.1.1 Development Tools 32](#_Toc81763245)

[5.1.2 Technology 32](#_Toc81763246)

[5.3 NETWORK IMPLEMENTATION 32](#_Toc81763247)

[5.4 SYSTEM SECURITY 33](#_Toc81763248)

[CHAPTER 6 35](#_Toc81763249)

[6.0 EVALUATION 35](#_Toc81763250)

[6.1 TESTING PROCEDURE 35](#_Toc81763251)

[6.2 TEST PLANS AND TEST CASES 36](#_Toc81763252)

[6.2.3 CAR Module 36](#_Toc81763253)

[6.2.4 USER Module 37](#_Toc81763254)

[6.2.4 CUSTOMER Module 37](#_Toc81763255)

[6.2.4 DASHBOARD Module 38](#_Toc81763256)

[6.2.4 RENTAL Module (MANAGE RETURNS) 39](#_Toc81763257)

[6.3 TEST DATA AND RESULTS 40](#_Toc81763258)

[6.4 ACCEPTANCE TESTING 40](#_Toc81763259)

[Chapter 07 41](#_Toc81763260)

[7.0 FUTURE ENHANCEMENTS, RECOMMENDATIONS AND CONCLUSION 41](#_Toc81763261)

[7.1 Future Enhancements 41](#_Toc81763262)

[7.2 Recommendations 41](#_Toc81763263)

[7.3 CONCLUSION 42](#_Toc81763264)

[7.4 LESSONS LEARNED 42](#_Toc81763265)

[CHAPTER 8 43](#_Toc81763266)

[8.0 REFERENCES 43](#_Toc81763267)

[1. Car Renting... Its Development... And Future 43](#_Toc81763268)

[2. Fenam car rental management system 43](#_Toc81763269)

[3. Suria Car Rental & Tour 43](#_Toc81763270)

[8.1 APPENDICES 43](#_Toc81763271)

[USER DOCUMENTATION 43](#_Toc81763272)

[START UP MODULE 43](#_Toc81763273)

[LOGIN 44](#_Toc81763274)

[main menu 44](#_Toc81763275)

[CAR MODULE 45](#_Toc81763276)

[CUSTOMER MODULE 45](#_Toc81763277)

[DASHBOARD MOLDULE 46](#_Toc81763278)

[About Module 47](#_Toc81763279)

# **CHAPTER** 1

## 1.0 INTRODUCTION

#### *Overview*

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.

## 1.1 BACKGROUNT AND MOTIVATION

### 1.1.0. BACKGROUND STUDY

This project was carried out to fulfill final year project requirement at the same time helping Car renting companies by delivering fully functional system to them. I preferred to develop the system as a desktop application to satisfy the companies need.

The problems of the existing manual system can be listed as follows:

* All the printing jobs related to rendition are recorded in a small system with an Access database and needs more functionality and reports, at the same time we have a lot of paperwork.
* Inventory control and managing was done manually in books which is hard to track car renting, returns and as well keeping records of customer’s information
* Printing machines does not have a computerized maintenance records which becomes hard to track breakdowns and maintenance work done which sometimes delay the overall activities within the company.
* Most of the data entry, validation and processing are done manually; this can be erroneous at times.
* Historical data cannot be systematically viewed or structured at any time.
* Generating reports with the existing data is time consuming and reports are not accessible from any location.

#### 1.1.1 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 desktop application is developed to overcome these problems. It would help the Company to manage time, effort, and resources more efficiently and assist the management in decision making process.

## 1.2 PROBLEM STATEMENT

Most of current systems used in different car rental companies are manual, where records are noted down in books. This costs the companies a lot with much time lost and ineffectiveness during renting or managing of car return to the company. It is also difficult for the manager to keep a track of which car was rented or returned and thus a great loss in case it becomes stolen or never returned to the company, and also in generating accurate reports. Even it’s a challenge to keep record of employees (the users of the system) and even updating their records as required.

Retrieval of data in the paper work system is very difficult because they are stored in a room thus time consuming to get a single file. One has to go manually and search for a file in a store /cabinet where they are stored

I decided to come up with this management system to overcome these challenges. The advantages of this system include:

* Employer will be in a position to keep record the users(employees) in the company
* It will enhance proper record keeping
* Paperwork will be reduced
* Retrieval of data will be made easier
* So efficient to customers who wish to rent a car from the company.

## 1.3 AIMS AND OBJECTIVES

From the above-mentioned problems, the aim is to develop a computer-based information system that will help to address the ongoing issues from the manual information system and help to facilitate some tasks that seems to be difficult for both the car rental company’s staff and those who are renting the cars (customers). The main objective is to design and implement a car rental management system for an organization. Specific objectives are:

* To study the current car rental management systems with too much paper works
* To identify the challenges in this paperwork car rental management system
* To design and develop a simple and secure system that protects user’s information, customer’s information and confidential information of the organization to overcome the challenges in paperwork car rental systems.
* To design a user-friendly system that enables customer check for availability of vehicle and make a rent or return back previously rented car.
* To implement geo-location and remote car deactivation upon expiration of rental period for ease of tracking back the car to the company’s store.

## 1.4 JUSTIFICATION

The system will:

* Eases the record keeping of user’s, customer’s, and car’s details
* Allow returns of cars by customers.
* Enable editing and deleting of existing rendition information.
* Compute on total cost as per hour on charge for each particular car.
* Show out a receipt and as well print it out.

## 1.5 SIGNIFICANCE OF THE SYSTEM

Projects provide a flexible framework for engaging students in exploring curricular topics and developing important skills, such as communication, teamwork, and technology skills.

The car rental management system will help to solve numerous problems associated with the manual way of doing things. Errors, waste of precious time and energy will be eliminated with the system. This will in turn enhance productivity and efficiency in an organization.

Also, it would help students and researchers that are working actively towards enhancing the car rental management system, this work would serve as a reference to them as they strive to develop the car rental company technologically.

#### Project Risks Assessment

##### Table 1. 1 Risk and Prevention

|  |  |
| --- | --- |
| RISK | PREVENTION |
| Loss of power | There is likelihood of loss of power when working on the project. To avoid that, the laptop used last for nearly five hours which is good enough between the times taken to bring back the light. |
| Inability to carry out research due to loss of hardware or software resources | Required hardware will be bought instantly while relevant software that may likely be lost will be kept in the computer for easy repair or reinstallation. |
| Loss of work due to equipment failure /loss | Weekly data backup to portable hard drive |
| Lack of Internet access | To prevent lack of internet access, two means of internet access was provided purposely for this project, which is accessing from free WIFI network provided by university and through use of paid services to buy internet bundles |

#### Scope/Project Organization

The aim of this project is to develop a desktop/Windows car rental management system. At this stage of development, the project would be launched using Microsoft SQL server Database which already comes along with Microsoft Visual Studio 2019.

**The Scope of this project is organized as follows:**

##### Login module

This module will use Username and password to access the system and also to recover lost passwords of the system users in the management module of users

##### Main Menu

This offers button links to navigate to all other system modules within

the system

##### Record keeping

Allows keeping track of all data fed to the system which can easy be viewed in datagridview from the system.

##### Dashboard module.

this module offers computation of charges of each car rented out for customer and as well shows out a receipt for the overall computation which can be printed out.

# CHAPTER 2

## 2.0 LITERATURE REVIEW AND SIMILAR SOLUTIONS

### Introduction.

So many car rental management systems that have been developed, and all these car rental systems are aiming at offering reliable services which can be accessed by customers at any time regardless of the location. This chapter contains the literature review of the car rental management system to be developed. This chapter will also review the existing systems that are similar to the car rental management system. References are made to sources from the internet.

A car rental is a company that rents automobiles for short periods of time, generally ranging from a few hours to a few weeks. Car rental agencies primarily serve people who require a temporary vehicle, for example, those who do not own their own car, travelers who are out of town, or owners of damaged or destroyed vehicles who are awaiting repair or insurance compensation.

**Car Renting... Its Development... And Future**  
Harris Saunders, a partner with his three brothers in a struggling real estate firm, was out for a spin in his Moline Dreadnaught with his current sweetheart when the car broke down. Harris had to have the Dreadnaught towed back to town where it was determined that it would be inoperative for several weeks. Harris' brother Joe was dismayed when he learned the news as the Dreadnaught was also used for business and without it, Joe realized that it would be difficult to travel around Omaha to sell lots and houses. It was then that Joe came up with the idea that was to launch a billion-dollar industry. Why not rent a car?[1]  
Vehicle renting has grown so popular in recent years that some may regard it as a recent development. One of the first references to the rental of a car is found in the Minneapolis Journal of July 22, 1904 when a bicycle shop devoted a line of an advertisement to announcing that it offered cars for rent. The renting business has come a long way since the Saunders brothers and Walter Jacobs started out. [1]

### Fenam car rental management system

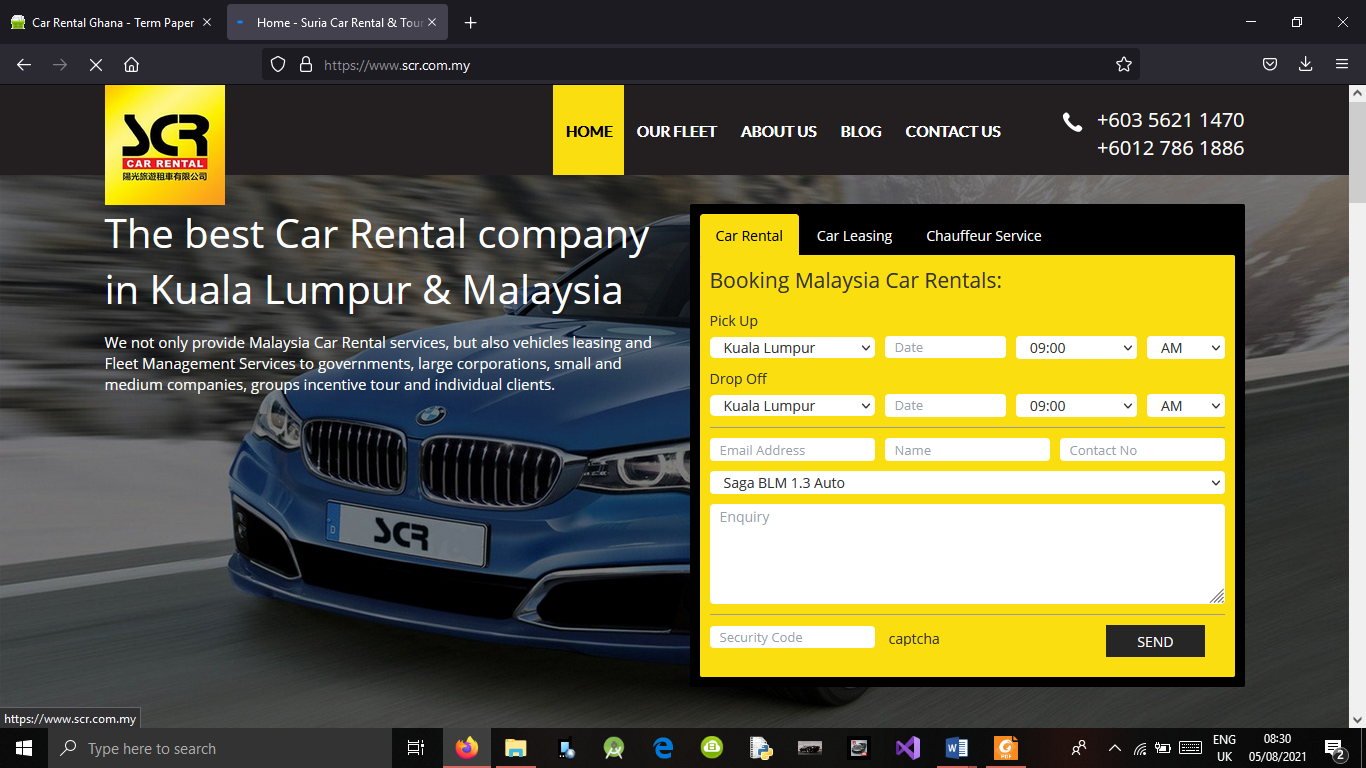
Fenam car rental management system is being developed in order for clients to book cars from the car rental company from anywhere in the world. This particular system will obtain data from the client by means of completing an online registration form with their personal information. By registering the web site, the customer will have the ability to book any car of their liking from the Fenam car rental website. [2] The Fenam car rental management system will entirely be incorporated on online systems. It computerizes the manual process by being very efficient and successful. The computerized system makes it possible for clients to provide full information on what kind of car they want to rent. [2] It consists of the kind of car they would like to hire as well as the location. The main aim of the fenam car rental management system is to create a web site for all people so they can book cars for rent together with their individual necessities from anywhere. [2]

### Suria Car Rental & Tour

Suria Car Rental & Tour Sdn TOUR Bhd (SCR &) has been committed in offering exceptionally great Budget Car Rental service and comfortable Malaysia Car Rental Services for both domestic and international travellers [3]. SCR focus on delivering customer-oriented services. In other word, they focus on customers’ needs and rent a vehicle tailored to their needs. Their philosophy is reflected in the services they offer, whereby they provide Budget Car Rental and Cheap Car Rental in Malaysia with excellent car condition and high-quality service which not only meeting customers transportation needs yet save their pocket. SCR have an extensive choice of Budget Car

Rental solution and services in Malaysia including single journey to/from the airport within Klang Valley, daily chauffeur rate, in Kuala Lumpur or out of the town, economical, family or luxury vehicles, and more![3]

In cooperation with reliable Strategic Partners, SCR also provide customized individual and corporate services including incentive tour, MICE and group outing both in Malaysia and overseas. Their customized service in the travel industry has benefited their clients from various industries covering government, large corporation, small and medium enterprises and individual clients (Who we are, 2020). In order to achieve the successfulness of their services, SCR and its Strategic Partners in Malaysia and oversea always ensure its daily operation being carried out professionally and reliable, which in turn, allowing them to maintain the quality services standard that is beyond industry benchmark over the years.



### Summary

This chapter shows that there is increase in the development of car rental industry, many companies are established, and digital innovation has extremely changed the way car rental companies are running their businesses. In addition, customers can now rent a car without suffering from manual registrations, booking and payment. Car rental companies also can keep track of all their data without encountering any problem.

# CHAPTER 3

## 3.0 REQUIREMENTS ANALYSIS AND DESIGN

## 3.1 Introduction

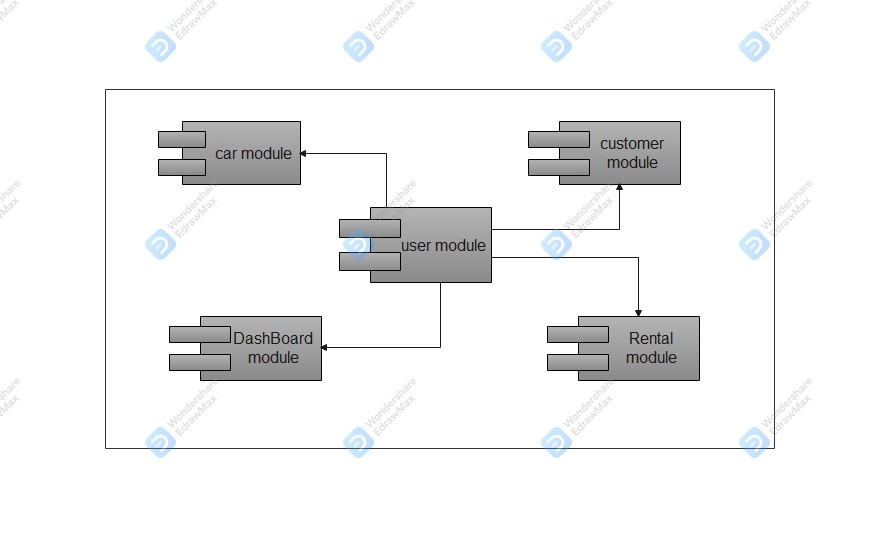
This chapter describes the analysis and design of car rental management system and the type of the methodology used in the system development. The requirements of the car rental management system are identified in this chapter, which includes functional and non-function requirements. The tools that are used when developing the system are identified, which consists of hardware and software tools as well as the technologies used.

In addition, this chapter describes the model of the system which is a simplified, complete, and consistent abstraction of the system, that is created for better understanding of the system using UML diagrams.

## SYSTEM DESIGN

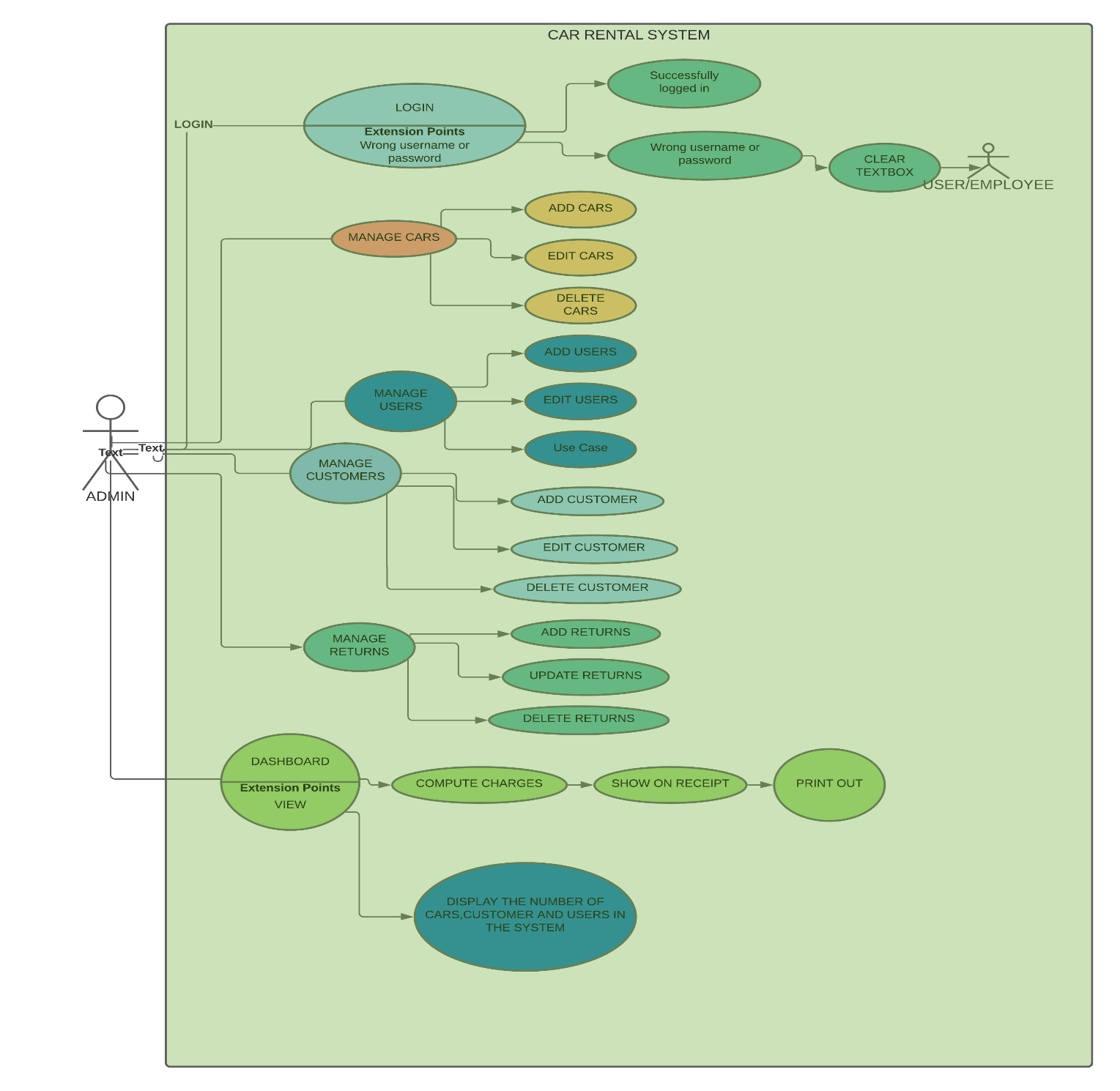
### System Architecture

The architectural design is the designing of the way that the overall system interacts with each and every module of the system. This is the process of identifying subsystems and building the framework for subsystems to control and communicate each other.



The figure above depicts the interaction between subsystems.

### Use case Diagram

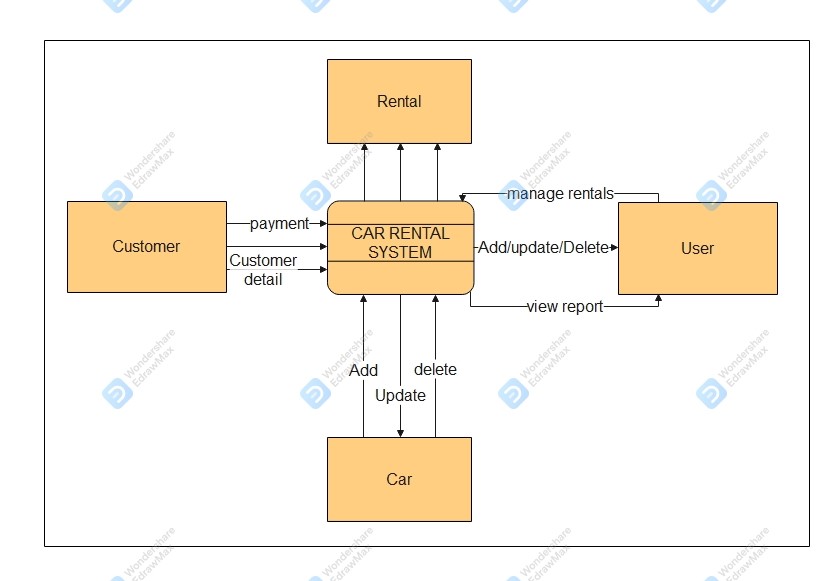
Below is a use case diagram that show the interaction between the external entities and the system 

This figure models the interaction between the admin and the system. In the diagram, from the left is an admin actor. Within the rectangle which represents the system are what are called **use cases**. The use cases are basically actions that an actor can perform in the system. From top left within the system is login use case, before login in to the system, the details for logging need to be valid, which is represented by the valid use case at the top-left corner.

### DATA FLOW DIAGRAM

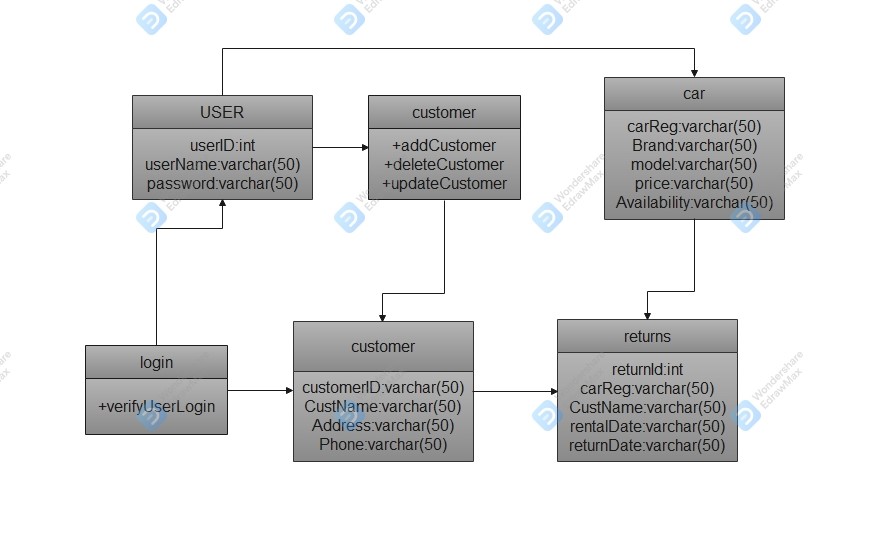
#### Context level *data flow diagram*

The Context Diagram above shows the overall view of the system, the data that goes in, and how it flows to other entities.



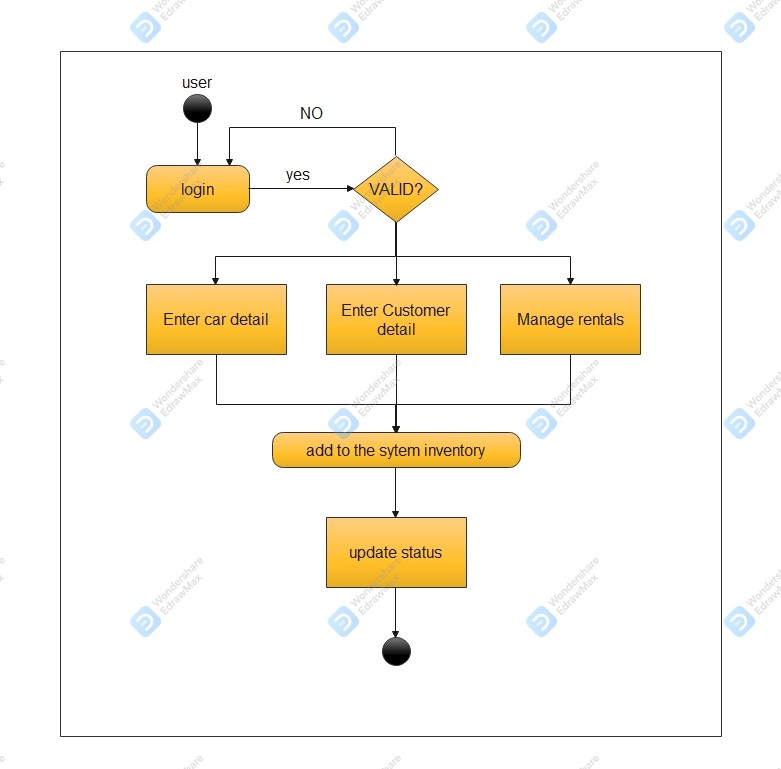
### Class Diagram

This figure below represents the class diagram of the system which shows the classes, including their attributes and operations.



### Activity Diagram

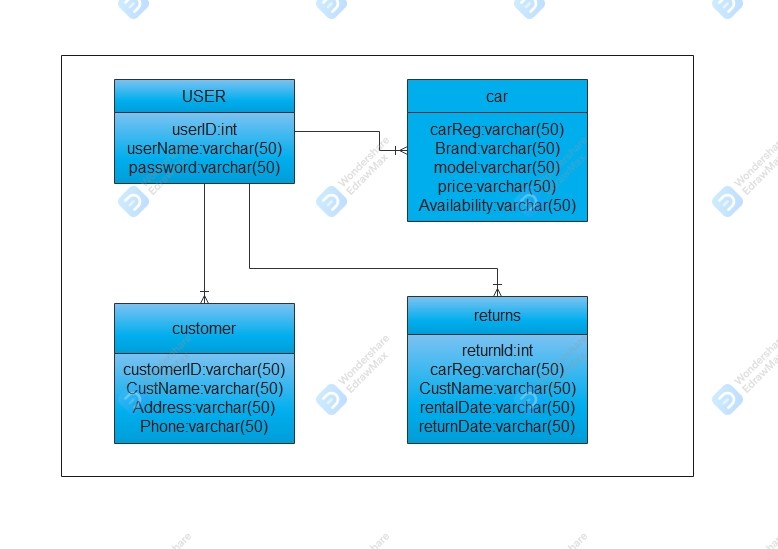
Below activity diagrams model the workflow within a system with the use of activity, action, decision, etc.



### Database Design

Database is modeled using Microsoft SQL Access Database. Database is a critical point to a system because this is where all the data related to the system is saved. The designed database is normalized to avoid redundancies

It is designed using the Entity Relationship diagram as below:



## 3.2 REQUIREMENT GATHERING

Requirement gathering hold a very important position since the requirement gathered are the basis of developing the software. Gathering requirements for a system is not an easy task since people may assume something that the analysts may know but actually, they do not know well about the problem domain

### 3.2.1 DATA COLLECTION TECHNIQUE

#### Observation

* 1. This involves getting the inner idea on working of current or previous fully functioning systems to checkout their problem domain and catch up the requirements which were mentioned in other methods of data collection.
  2. This method of observation is very useful to identify the workload other systems have when carrying out daily activities so as to work on developing a system that outcomes these problems.

### 3.2.2Tools and Techniques

The following tools are used throughout the project

* A computer with 2.5Ghz of processor, 4GB of RAM, and 500GB of Hard drive
* Microsoft office programs for necessary documentations and visualizations
* Microsoft Visual Studio as IDE
* Internet access to download Images that were fed to the system.

#### Technologies used in this project are as follows

* Windows Forms .Net Framework with C# programming language
* Microsoft SQL server database that comes along with visual studio

### 3.2.3 Requirement Analysis

At the end of the requirement analysis process, the expected deliverable is System Requirement Specification (SRS), which describes the behaviors and features of the system. It also contains the list of requirements of the system as well as the diagrams.

Requirements list are the list of functions that a system must possess.

Requirements are of two categories, which are:

#### Functional requirements

Functional requirements are the requirement that the stakeholders need from the system, how is the system going to be operated and what the system should have.

Functional requirements in this project includes:

**Dashboard module:**

* System should be able to generate the number of Cars, Users, Customers, in the system.
* System should perform computations for every customer on charges per hour on every car that was rented out.
* Generate a unique Reference number for every return made.
* Generate a Receipt for every return made
* Allow printing of the generated receipt.

**Car Module**

* + - System should allow Adding news cars to the system store.
    - Should also allow Updating/Deleting previous details available for each car in the store.

**User Module**

* + - System should allow Adding New Users to the system.
  + Should also allow Updating/Deleting previous details available for each User in the store.

**Customer Module**

* This module enables adding New customers to the system and the details store which can easily be retrieved for either editing or deleting.

**Rental Module**

* + This module takes the data captured from the dashboard module after a successful return has been made.
  + It also allows editing or deleting previous details done on each return.

#### Non-functional.

Non-functional requirements are the requirements that specifies the criteria that can be used to judge how the system operates.

For the non-functional requirements the system should be:

* The system should be accurate and consistent.
* The system should not give company data to an outsider (Security).
* **The Login Module.**

This allows entry to the system by only authorized users of the system.

* The system should be reliable.
* It should be user friendly.
* System need to flexible and expandable in the future.
* The system should have a very appealing User interface.

# CHAPTER 4

# 4.0 METHODOLODY

## 4.1 INTRODUCTION

Developing software needs a good design to success the project. Nowadays developers use iterative models to develop software which means several iterations of analyzing, designing, coding and testing. The first stages of the design have an overall view of the system and then the subsystems will be designed and expanded to fit the design of the system.

## 4.2 The Iterative Model

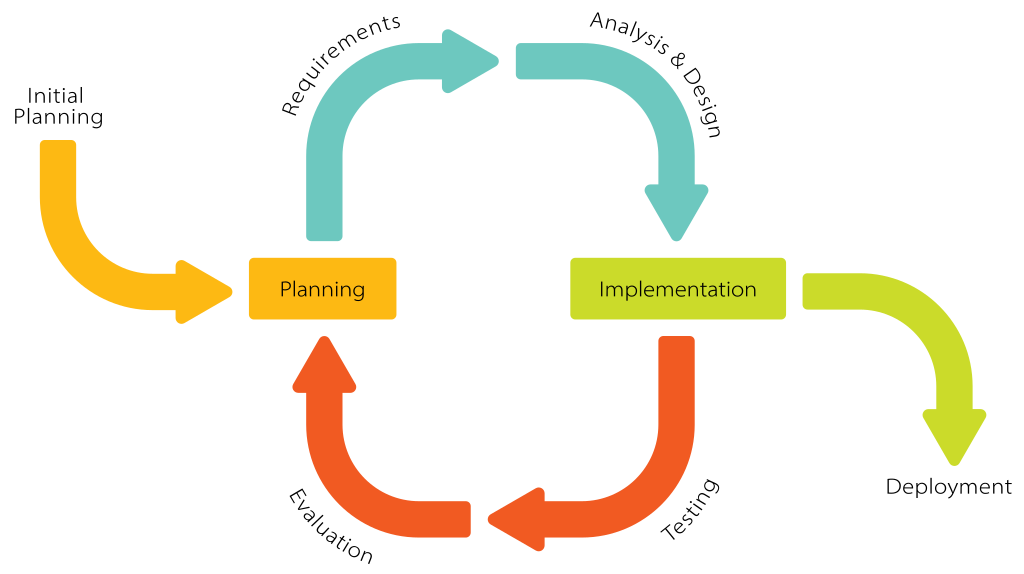
The selected method to develop the system is **Iterative model**, is more flexible and can be altered according to the developers’ choice.

The iterative model is a particular implementation of a software development life cycle (SDLC) that focuses on an initial, simplified implementation, which then progressively gains more complexity and a broader feature set until the final system is complete. When discussing the iterative method, the concept of incremental development will also often be used liberally and interchangeably, which describes the incremental alterations made during the design and implementation of each new iteration.

#### The Process

Unlike the more traditional waterfall model, which focuses on a stringent step-by-step process of development stages, the iterative model is best thought of as a cyclical process. After an initial planning phase, a small handful of stages are repeated over and over, with each completion of the cycle incrementally improving and iterating on the software. Enhancements can quickly be recognized and implemented throughout each iteration, allowing the next iteration to be at least marginally better than the last.

* **Planning & Requirements**: As with most any development project, the first step is going through an initial planning stage to map out the specification documents, establish software or hardware requirements, and generally prepare for the upcoming stages of the cycle.
* **Analysis & Design**: Once planning is complete, an analysis is performed to nail down the appropriate business logic, database models, and the like that will be required at this stage in the project. The design stage also occurs here, establishing any technical requirements (languages, data layers, services, etc.) that will be utilized in order to meet the needs of the analysis stage.
* **Implementation**: With the planning and analysis out of the way, the actual implementation and coding process can now begin. All planning, specification, and design docs up to this point are coded and implemented into this initial iteration of the project.
* **Testing**: Once this current build iteration has been coded and implemented, the next step is to go through a series of testing procedures to identify and locate any potential bugs or issues that have cropped up.
* **Evaluation**: Once all prior stages have been completed, it is time for a thorough evaluation of development up to this stage. This allows the entire team, as well as clients or other outside parties, to examine where the project is at, where it needs to be, what can or should change, and so on.

****

**Figure: Iteration Model phases**

##### Advantages of the Iterative Model

* **Inherent Versioning**: It is rather obvious that most software development life cycles will include some form of versioning, indicating the release stage of the software at any particular stage. However, the iterative model makes this even easier by ensuring that newer iterations are incrementally improved versions of previous iterations. Moreover, in the event that a new iteration fundamentally breaks a system in a catastrophic manner, a previous iteration can quickly and easily be implemented or “rolled back,” with minimal losses; a particular boon for post-release maintenance or web applications.
* **Rapid Turnaround**: While it may seem like each stage of the iterative process isn’t all that different from the stages of a more traditional model like the waterfall method — and thus the process will take a great deal of time — the beauty of the iterative process is that each stage can effectively be slimmed down into smaller and smaller time frames; whatever is necessary to suit the needs of the project or organization. While the initial run through of all stages may take some time, each subsequent iteration will be faster and faster, lending itself to that agile moniker so very well, and allowing the life cycle of each new iteration to be trimmed down to a matter of days or even hours in some cases.

* **Suited for Agile Organizations**: While a step-by-step process like the waterfall model may work well for large organizations with hundreds of team members, the iterative model really starts to shine when it’s in the hands of a smaller, more agile team. Particularly when combined with the power of modern version control systems, a full “iteration process” can effectively be performed by a number of individual team members, from planning and design through to implementation and testing, with little to no need for outside feedback or assistance.
* **Easy Adaptability**: Hinging on the core strength of constant, frequent iterations coming out on a regular basis, another primary advantage of the iterative model is the ability to rapidly adapt to the ever-changing needs of both the project or the whims of the client. Even fundamental changes to the underlying code structure or implementations (such as a new database system or service implementation) can typically be made within a minimal time frame and at a reasonable cost, because any detrimental changes can be recognized and reverted within a short time frame back to a previous iteration.

# Chapter 05

## 5.0 IMPLEMENTATION

In this chapter the implementation of the system is being discussed. Implementation is the process of converting system design into the real-world system which can be used in day to day business tasks. This is a critical task in developing a system and where most of the time consumes.

According to the system requirements, developers have the ability to choose tools and techniques which are more familiar to them for implementing the project. The developer should take care of the task that the program codes are accurate and readable

## 5.1 IMPLEMENTATION ENVIRONMENT

When considering about the developing environment most of the tools that have been used for development of the software are open source which will not end up with legal issues.

|  |  |
| --- | --- |
| **Hardware** | **Software** |
| * Intel Core i3 processor 2.53 GHz * 4 GB RAM and above * 320 GB Hard Disk and above | * Microsoft Windows 10 PRO * Microsoft Visual Studio 2019 IDE * Windows Forms .Net framework in C# * Microsoft Sql Server Database that comes along with Microsoft visual studio |

**Table 5.1.0**

Table 5.1.0 above shows the hardware and software used in implementing the system. The system was developed in a windows 10 computer.

### 5.1.1 Development Tools

* Microsoft Sql Server for Database handling
* Firefox and other web browsers for getting images that were fed to the system
* Lucid Chart app for UML designing for documentation
* Wondershare Edraw for modifying the UML designs.
* Microsoft Word for typing the documentation

### 5.1.2 Technology

* C# was the main development language used in coding the system.
* Microsoft SQL server database within the Microsoft Visual studio 2019 IDE was used in creating and handling databases.
* Microsoft Visual studio 2019 IDE was used in styling the interfaces for more attractive user-friendly look and feel.

## 5.3 NETWORK IMPLEMENTATION

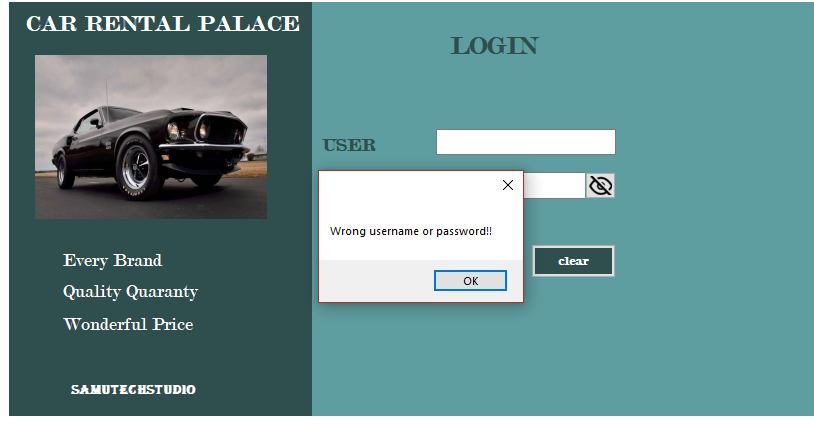
Since the system is built as a Desktop application, it can be installed on a computer where the employees of the Car rental company can access the system in the company.

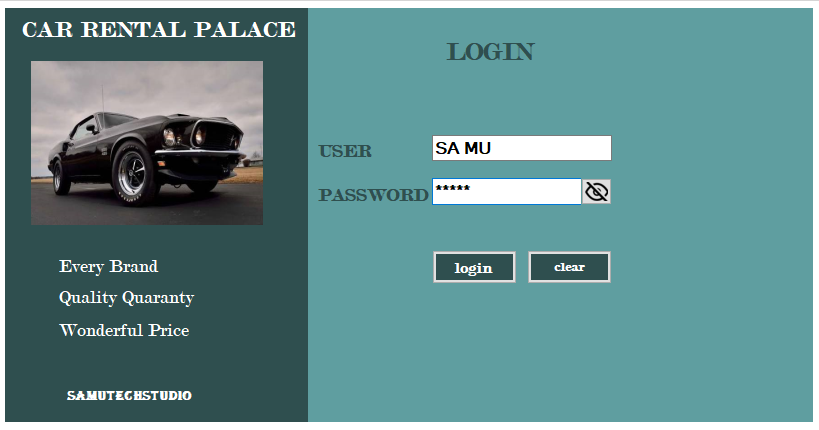
The system was installed and implemented from the same hardware where it was developed in, later a test done from another computer, but the challenge involved was reading and accessing the database modified in the first computer, hence a connection to access the database was modified and all operations worked as required. The database is also installed in the same computer as the system is.

## 5.4 SYSTEM SECURITY

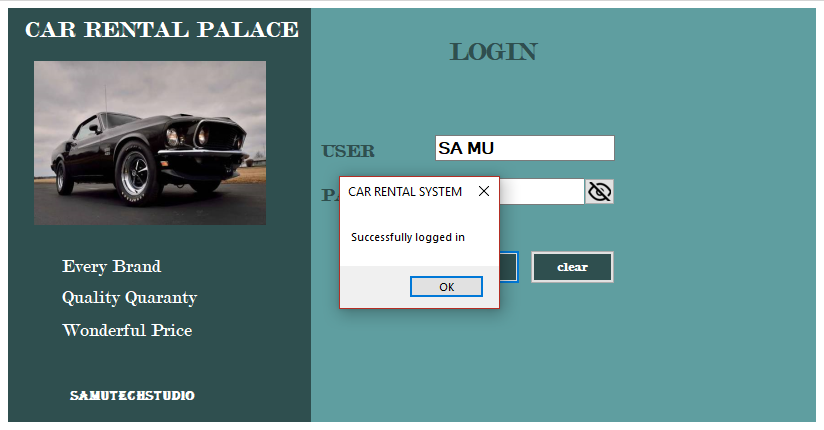
The system contains very sensitive data related to car renting and maintenance activities. It is important to have a proper access control mechanism to secure the system data.

When logging into the system the Username and password is checked by retrieving the data as it were coded when building the login module. If invalid, an alert message pops up indicating Wrong Username or Password! Else an alert pops out indicating a successful login





The figure above shows only the username, but the password hidden for more security measures to the system because when key-in it an outsider may see it

on a successful Input of valid username and password the alert message on Successful log in is displayed.

# CHAPTER 6

## 6.0 EVALUATION

Evaluation is where the system is tested. System testing can be done in several ways. Evaluation holds critical role when it comes to quality of the system because the system should do the expected task without any errors. This is a two-step process which is verification and validation. **Validation** is the process of determining whether a fully developed system conforms to its requirement specification. **Verification** is the process of determining whether the output of one phase of a software development conforms to that of previous phase. Testing is done in several stages of the system to reduce errors.

Once errors have been identified in a program code, it is necessary to first identify the precise program statement responsible for the error and then fix them.

## 6.1 TESTING PROCEDURE

Testing a program consists of providing program with set of test inputs and observing if the program is as expected. If the program fails to behave as expected the condition under which failure occurs are noted for later debugging and correction. Aim of testing is to identify all the defects existing in a software product.

Software testing is most of the times integrated with system development in today’s context to ease of debugging and to reduce the errors.

The Car Rental system was tested for each and every interface and class while in the development phase. This can be considered as **unit testing** and tests were performed in both white box and black box methods. Unit testing is undertaken after a module has been coded and successfully reviewed. Different units were tested in isolation. **White-Box Testing** involves test cases where there is thorough knowledge about the internal structure. **Black-Box Testing** approach, test cases are designed using only the functional specification of the system. Test done without any knowledge of the internal structure of the software. Also referred to as Functional Testing.

In the Car rental system, white box method in each and every code section is revisited to check if the logic is correct then in black box testing the page or the module is used to perform the required task and check with the database whether the task has happened correctly.

Also, the system was tested for integration, functionality, etc. Integration testing, its primarily objective is to test the module interface. During Integration testing different modules are integrated in a planned manner. That is, in our system, Car rental system load from start to Login Module to Main Menu from Which Other Modules are linked and easily accessed.

To overcome bugs and errors of the system integration testing, regression testing, interface testing, defect testing and the system testing were performed according to the test cases listed below.

## 6.2 TEST PLANS AND TEST CASES

Test plans that are listed below will cover all types and phases of testing that were used to guide the testing process. Test plan was designed in the analyzing and designing phase of the system development. Some of the test cases were developed before the implementation, some were developed while implementing and some were at the end of implementation.

Test cases were organized in the modular manner where the system was, and that got the steps to test and the result expected listed in the table.

### 6.2.3 CAR Module

Test cases for the car module is shown below;

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Test Cases** | **Procedure** | **Expected results** |
| 1 | Add Car description | add car description | Car added to the data grid |
| 2 | Update car details | Click a row on the data grid view and edit details within the text boxes | Car successfully updated |
| 3 | View Data grid | check whether entered  data is available | Updates for edit car details |
| 4 | Delete car from the system | Click row on data grid and click on delete button | Car deleted successfully |

**Table: Test cases for Car module**

### 6.2.4 USER Module

Test cases for user module is shown below;

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Test Cases** | **Procedure** | **Expected results** |
| 1 | Add User details | Write user details on every text box in the module and click on ADD button | User added to the data grid |
| 2 | Update User details | Click a row on the data grid view and edit details within the text boxes | User successfully updated |
| 3 | View Data grid | check whether entered  data is available | Updates for edit car details |
| 4 | Delete user from the system | Click row on data grid and click on delete button | User deleted successfully |

**Table: Test cases for User module**

### 6.2.4 CUSTOMER Module

Test cases for customer module is shown below;

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Test Cases** | **Procedure** | **Expected results** |
| 1 | Add Customer details | Write customer details on every text box in the module and click on ADD button | customer added to the data grid |
| 2 | Update customer details | Click a row on the data grid view and edit details within the text boxes | customer successfully updated |
| 3 | View Data grid | check whether entered  data is available | Updates for customer details |
| 4 | Delete customer from the system | Click row on data grid and click on delete button | customer deleted successfully from the system |

**Table: Test cases for customer module**

### 6.2.4 DASHBOARD Module

Test cases for user module is shown below;

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Test Cases** | **Procedure** | **Expected results** |
| 1 | View on dashboard | Checking if all data loaded on the dashboard | All data from Customer and cars databases should be displayed on different data grid view as modeled respectively.  The number of Cars, Users, Customers on the system should also be displayed |
| 2 | Compute Charges | Autofill all text boxes as per expected by clicking on different rows from Customer Data grid view and from Cars data grid view, then fill on Charges per hour and on fine and click on TOTAL button to compute the total charges. | The computation is accepted and shown on the total charge text box.  Data is sent to the Rental Module. |
| 3 | Show computed data on receipt | After a successful computation click on **show receipt** button | All data should be displayed on the receipt box. |
| 4 | Print out the receipt | Click on **print receipt** button | A print preview dialog should pop up on another window. |

**Table: Test cases for Dashboard module**

### 6.2.4 RENTAL Module (MANAGE RETURNS)

Test cases for Rental module is shown below;

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Test Cases** | **Procedure** | **Expected results** |
| 1 | View the data grid view | Data is sent to the rental module after a successful return in the dashboard module | Return are displayed in the data grid view |
| 2 | Add |  |  |
| 3 | Delete |  |  |
| 4 | Update |  |  |
| 5 | View on Update |  |  |

**Table: Test cases for Rental module**

## 6.3 TEST DATA AND RESULTS

Each module of the system was test according to the above criteria and common set of data items were used. Each form of the system was checked entering data in every field on the development phase at unit testing. Model data sets were used to check the functionality of the system.

Error messages are used where necessary to inform user regarding any errors and some error messages are used to ensure that all the required data is inserted before submitting the form.

## 6.4 ACCEPTANCE TESTING

Results of the test indicated that the system is performing well with the real environment. Some small bugs found were corrected as they were discovered. Feedback from the system user and managers was very good and the system helps them to perform well compared to the past.

# Chapter 07

## 7.0 FUTURE ENHANCEMENTS, RECOMMENDATIONS AND CONCLUSION

### 7.1 Future Enhancements

Some enhancements supposed to be made for this project is includes;

* **Application should run on mobile phones and be accessed from both mobile websites and desktop websites**

Nowadays, the number of mobile phone users is very high than computer users. The application is going to be in android, iOS platforms and also developed in order to increase accessibility and inclusiveness.

* **Geofencing and remote car deactivation**

When geofencing is integrated in the application, it will use GPS or WI-FI to trigger a preprogrammed action when the mobile device owned by the customer enters or leaves a virtual limit set up around a geographical area.

* **Car sales**

Car sales functionality will be implemented to the system when this project got online; therefore, user can register with the system as a salesman and add the car he/she wants to sale. So, when this functionality is in place, customers can not only rent car, but buy cars from the system as well.

### 7.2 Recommendations

The car rental company need to open branches in some other states. The project should have a cloud storage for backing up user’s data because the cloud storage can save costs, protects user’s data from ransomware or malware, regulatory compliance, and data tiering for cost savings. Also, the web-based car rental system should also be developed in form of mobile application, to help improve usability for mobile phone users.

### 7.3 CONCLUSION

This project is been developed for SAMU CAR RENTAL COMPANY based in KENYA, therefore the project is currently limited within KENYA, and most of the users will are Kenyans. Majority of the processes that will be carried out manually in the company are analyzed and implemented. For a user/ Employee to rent out a car, he/she must login to the system to get access to the system functionalities.

The system was developed using C# as the base programming language and used Microsoft Visual Studio 2019 IDE for the development task. The system design was carried out using UML (UNIFIED MODELLING LANGUAGE) as a tool for designing. Database was designed using Microsoft SQL database access and the database was used as the data storing part of the system.

### 7.4 LESSONS LEARNED

The knowledge gained through the project was worthy. With respect to the knowledge gained throughout the degree program by every module of study this is where I got the chance to apply my theoretical knowledge into real work problem.

This project helped me to gain knowledge regarding desktop Software development process. It also needed me to dig into some areas which are not covered by the course modules taught which led me to practice how an unknown problem to me can be solved with the research.

Further working on the project helped me to develop technical skills on C#, SQL database and Object-oriented approach. Also, the project helped me a lot to develop my communication skills and intellectual skills by collaborating with many individuals from collective fields.

# CHAPTER 8

## 8.0 REFERENCES

### Car Renting... Its Development... And Future

* + <https://www.termpaperwarehouse.com/essay-on/Car-Rental/393155>

### Fenam car rental management system

* + <https://www.termpaperwarehouse.com/essay-on/Thabisos-Upload/425077>

### Suria Car Rental & Tour

* + <https://www.scr.com>

## 8.1 APPENDICES

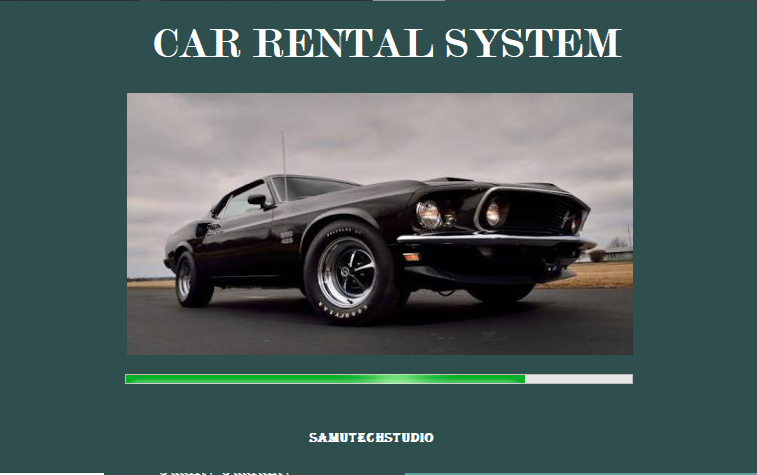
### USER DOCUMENTATION

#### QUICK STARTER GUIDE

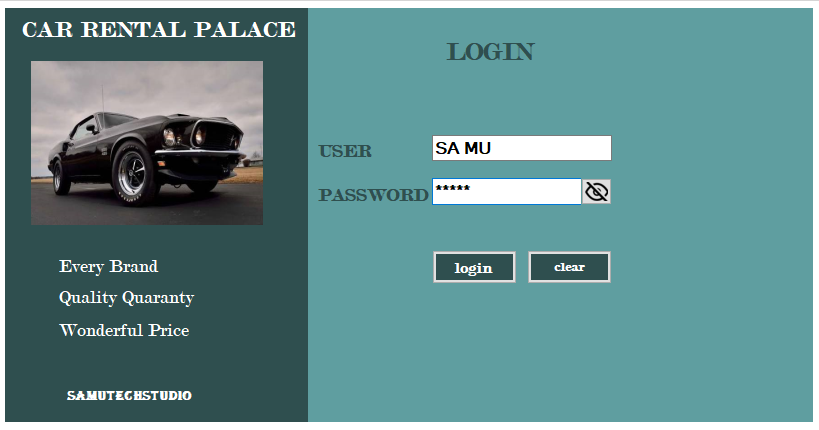
The system got 2 types of users and they got different access levels according to their role. Basically, the system administrator and the Employee, they can add, edit or delete users, customers, cars updates and as well manage car rentals and returns.

### START UP MODULE

It shows how the system loads when it is launched.



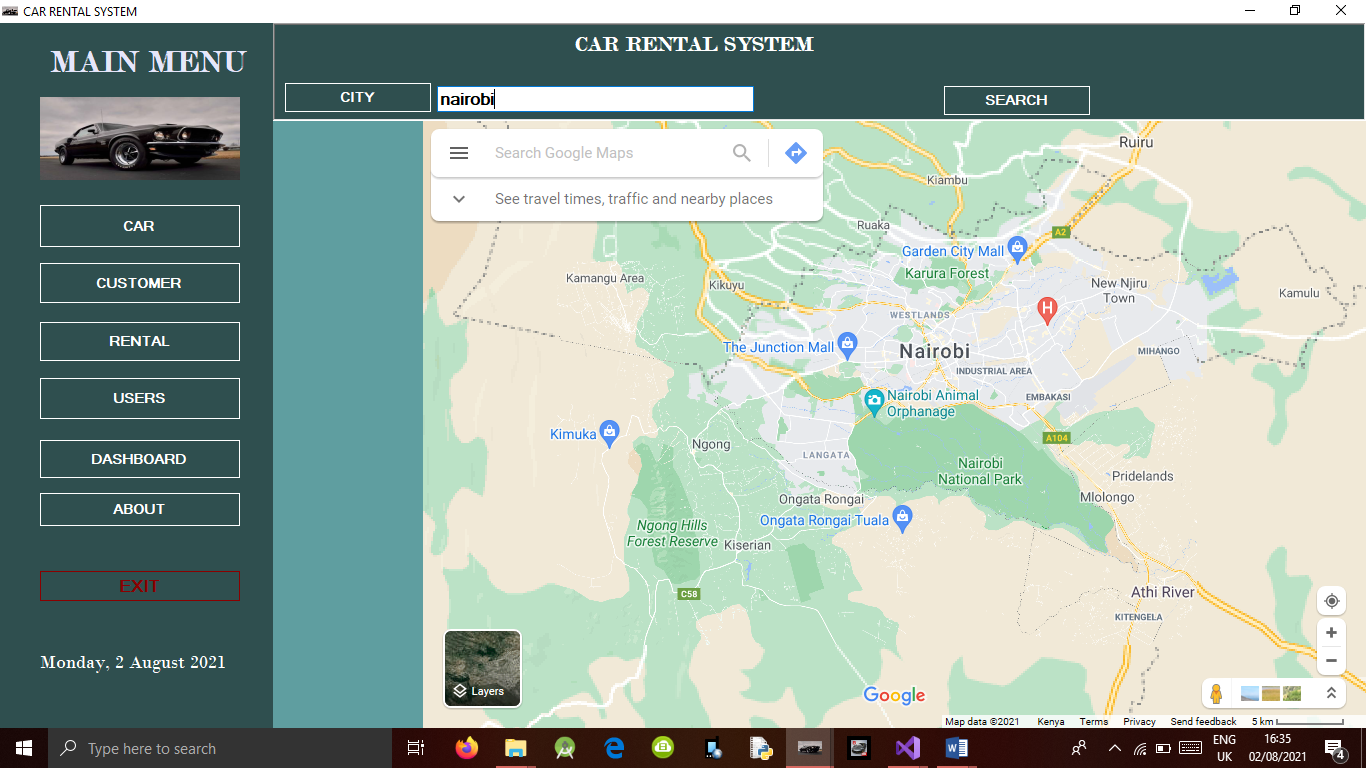
### LOGIN

****

shows login screen for the first time the system is launched, use SA MU as the user name and password as 12345 to login to the system once you logged in change password.

Once logged in, user is directed to the main menu interface of the system which provides full access to other system modules.

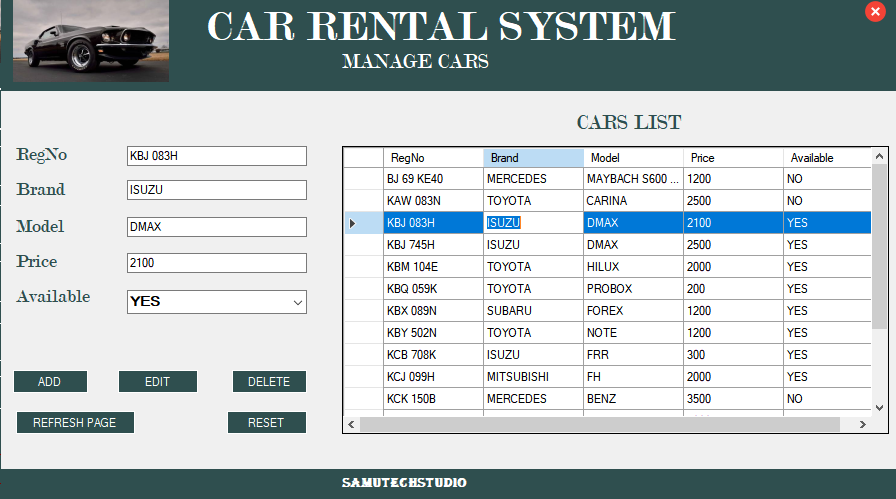
### main menu

****

It has several buttons that help the user to navigate to all other system modules

It has also a google map for Geofencing to track down a lost car in case it gets stolen or user never returns it.

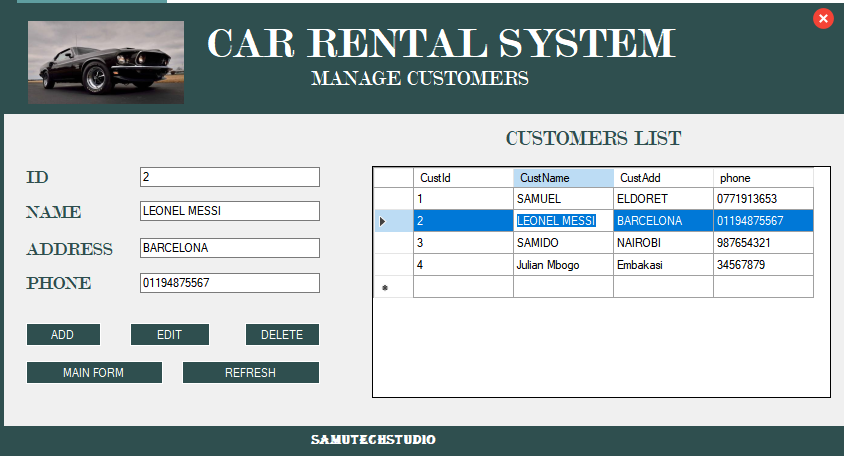
### CAR MODULE



User/employee can add new cars to the system, delete or update the existing details of the cars using the respective buttons as shown on the module.

Refresh page enables reloading the page to display any new data in the grid view.

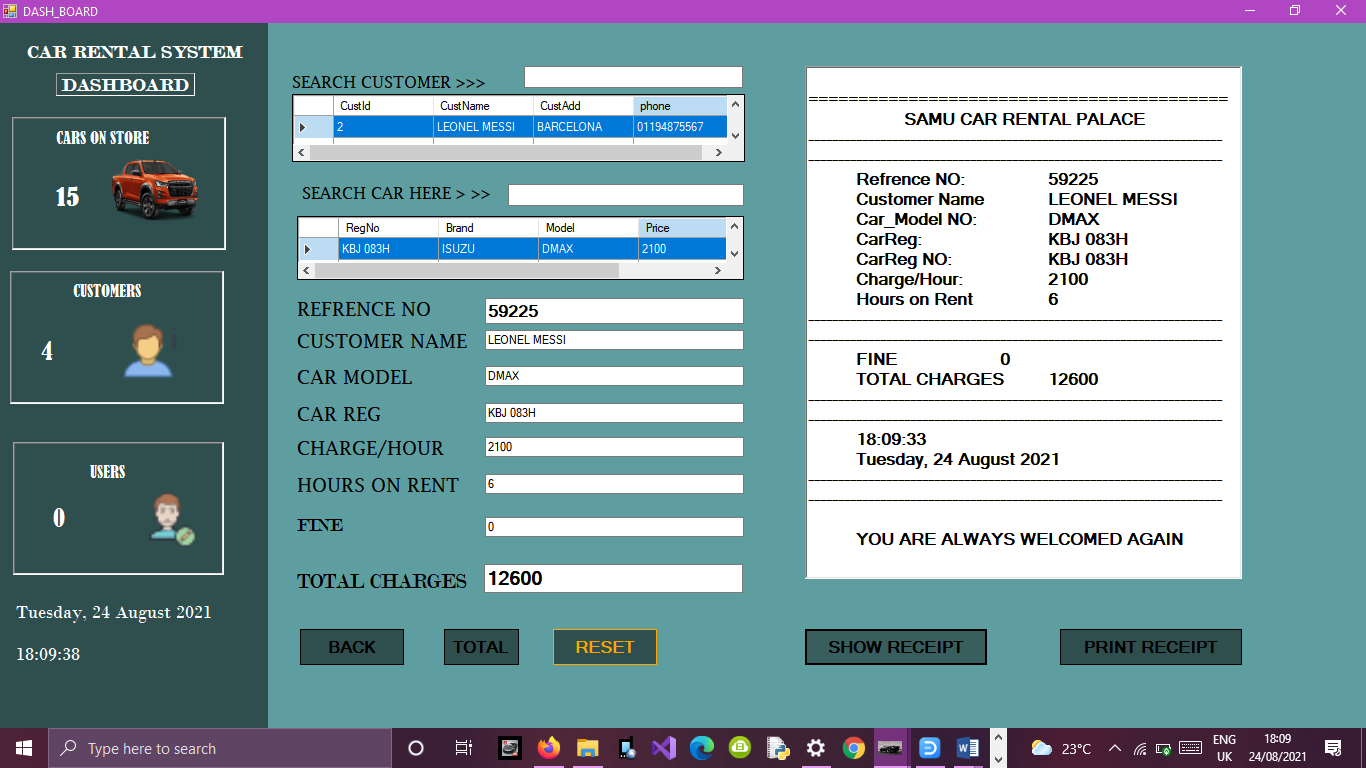
### CUSTOMER MODULE



User/employee can add new customers to the system, delete or update the existing details of the customer using the respective buttons as shown on the module.

Refresh page enables reloading the page to display any new data in the grid view.

### DASHBOARD MOLDULE

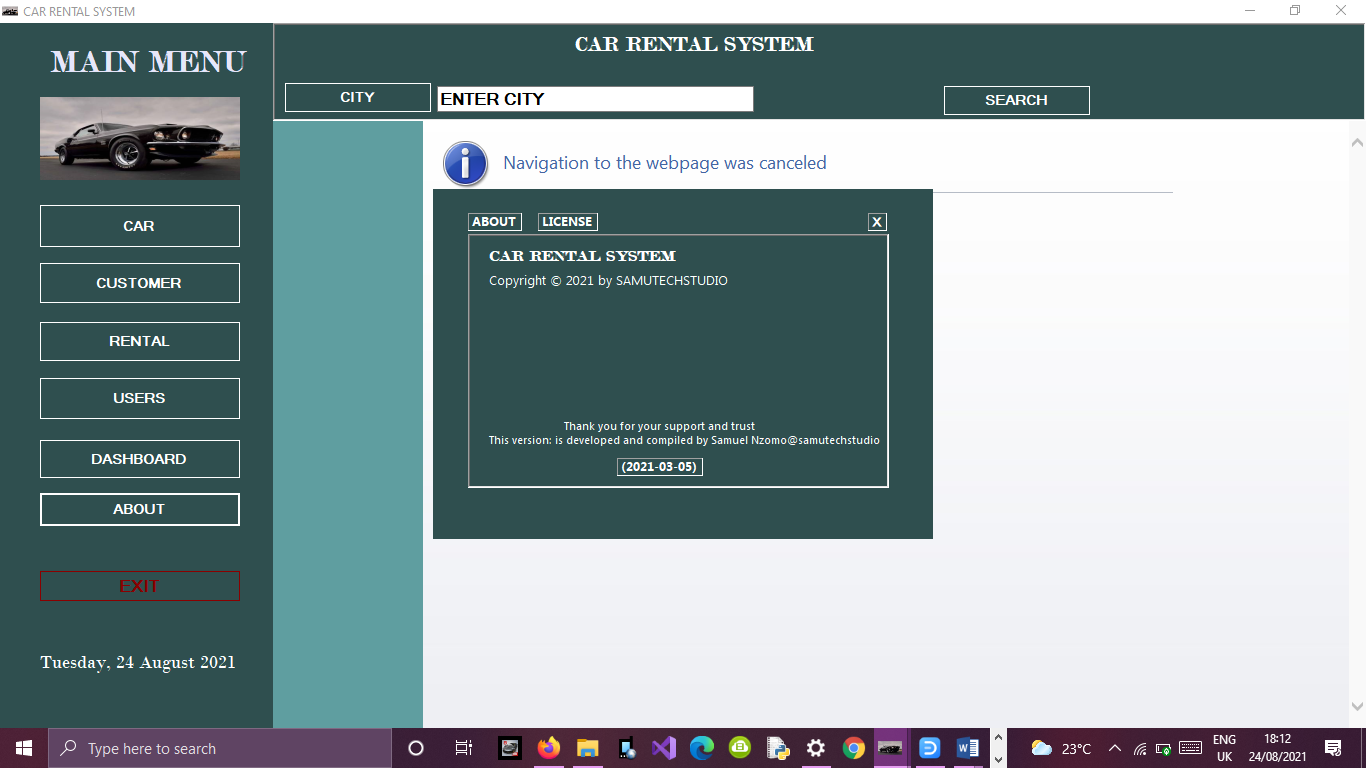


In this module a user can perform several actions:

* Access stored details of customer and car to automatically feed the data to various text box by just clicking on a given row on either the customer’s data or from the car’s data.
* Compute a charge for a given car by manually inputting the number of hours the customer was with the car and automatically the system generates the total charge by just clicking on the TOTAL button.
* User view the receipt of a given served customer and print it using the PRINT RECEIPT button
* User can view the number of cars, customers and users in the system.

### About Module

About module has information about the factory.



***GOD BLESS YOU.***