DOKUZ EYLÜL UNIVERSITY

ENGINEERING FACULTY

DEPARTMENT OF COMPUTER ENGINEERING

CME 2210

Object Oriented Analysis and Design

RENT-A-CAR (RAC) MANAGEMENT SYSTEM

by

2017510003 Melisa Adaklı

2018510128 Hasan Korkmaz

2019510025 Cemil Dalar

İzmir

31.05.2022

Table of Contents

CHAPTER ONE

INTRODUCTION

1.1 What the problem is

The car rental system is a system where you can rent a car from the nearest location in Izmir as long as there is a suitable car at the desired time. There are many reasons for the creation of this system. The first of these is problems such as renting the same car to more than one person. It turns out, when a vehicle is rented, it does not appear to be suitable for renting because it is not suitable for rental, and it prevents customers from renting, thus preventing customers from being disappointed. In addition, it ensures that customers can get the vehicles they rent from the employees in the offices located in certain cities and districts, without being late, from the employees in the offices at the time they make an appointment.

One of the other positive aspects of the car rental system is that it saves the worry of regular maintenance in this period when time is one of the most important things, and it also provides financial advantages. It will be the most logical option to use the car rental system to get rid of loads such as insurance fee and service.In addition, it meets the customers' desire to drive a luxury vehicle. By renting a car that it would normally be difficult to buy, it will experience both comfort and happiness.Avoiding negative experiences, without a face-to-face system registration, the risk of being scammed and disappointed by car rental is avoided.Normally, using public transportation is cheaper than renting a car for customers, but when using public transportation, there is a high probability of not catching vehicles such as buses and trains on time and being late. In addition, in order not to miss the time, they should prepare hours in advance and be at public transportation stops in order to reach their destination. They waste their time on their journey.

1.2 Goals for the Project

The car rental management system is a system that manages car rental reservations, vehicles and even rents vehicles that are suitable for rental.The purpose of use of the Car Rental System is for both admins and customers, making reservations and management and reliability of reservations easier.The customer can create a reservation by entering their personal information to rent the vehicle at the appropriate hours and days.When this information is approved by the admin, the reservation is also confirmed.The customer can rent the vehicle the person wants to rent by specifying the rental hour and day, by creating credit card information and user registration.

1.3 Stakeholders

Several different types of stakeholders can be noted when it comes to our software. The most obvious are those that requested for this software: the admin, staff and customers. We made note to critical details on what the company wants, how they want it, and how it should benefit both them and their clients. We plan to keep them in the know 24/7, keeping them informed on what is being planned for their new software. This allows for the admin and customers to be aware of what they are getting from their new software.

Any local or district managers are also influenced by the software's stability. How easy the software is for the employees in turn represents how well the manager is able to guide them through it.

1.4 Motivation for the Project

Always thinking that the user deserves a better system, we have considered it our duty to be able to solve their needs in a timely manner. Designs are made that will alleviate the burden not only of customers, but also of the company and its employees.Our team has an immense amount of knowledge when it comes to problem solving, programming, and communication.Not only would we strive to give the car rental management system everything they desired, but we will continue to make sure the software is at its very best and beyond.Our team does not only design our software, but also carefully examines every detail, without skipping any steps, from testing to software management.

1.5 Process Flow Preview

We plan to use an iterative route throughout our process, so we think that communication throughout the process should be tight and proceed according to the conclusions drawn from these communications.A path is followed by discussing the requirements and developing studies on the notes taken.The modeling process does not have a definite start and end process.There will be times that we may have to go back to certain portions within the modeling activity to ensure a sufficient model. We wish to make sure that if we miss anything, we do not figure that out in the construction stage.Allowing users to see how the software is made will ease the transition during the deployment phase.

CHAPTER TWO

2.Analysis and Design

2.1 Plan for Requirements Engineering

Inception Task:

The goal for the beginning is to identify the business case created by the stakeholders. To get a basic understanding of the project, here were some more questions we asked:

" Who is behind the request for this work? "

" Who will use the solution? "

" What will be the economic benefit of a successful solution? "

" Is there another source for the solution that you need? "

These questions enable the requirements engineer to gain a better understanding of the problem and allow the customer to voice his or her perceptions about a solution:

How would you characterize "good" output that would be generated by a successful solution?

Can you show me (or describe) the business environment in which the solution will be used?

Elicitation Task:

Our goal at this stage is to identify the problem, suggest solutions, and brainstorm different ideas. Meetings with the software engineering team and stakeholders are scheduled to gain a more refined understanding. The plan is to get a solid idea of what the system's goals are, what needs to be achieved, and how the overall system fits into the rental car management system. In general, lists will be created to understand who the stakeholders involved are, descriptions of the technical environment, use cases and currently a requirements list.

Specification Task:

During this task, we plan to create a software requirements specification template. In this template we will get the overall purpose of the project and the target audience. Descriptions regarding the product features, user classes/characteristics, operating environment and design will be included. Also included are safety and security requirements, quality attributes, and what interfaces are to be used with this software.

Validation Task:

During validation, the work products produced as a result of requirements engineering are assessed for quality.

• The specification is examined to ensure that

– all software requirements have been stated unambiguously

– inconsistencies, omissions, and errors have been detected and corrected

– the work products conform to the standards established for the process, the project, and the product

• The formal technical review serves as the primary requirements validation mechanism

– Members include software engineers, customers, users, and other stakeholders.

Requirements Management:

• During requirements management, the project team performs a set of activities to identify, control, and track requirements and changes to the requirements at any time as the project proceeds

• Each requirement is assigned a unique identifier

• The requirements are then placed into one or more traceability tables

• These tables may be stored in a database that relate features, sources, dependencies, subsystems, and interfaces to the requirements

• A requirements traceability table is also placed at the end of the software requirements specification.

2.2 Functional Requirements

Hardware Requirements:

The software should be run on any sort of desktop or laptop environment on Windows.

Interface - Primary Tasks:

- View all available cars

Connects with the database through Java to call all cars and display

- Select the desired and suitable car

A select button will be displayed under the rental car previews that will allow for the selection

- Allow the customer to log in

Log in display form

- Search for desired car by model, car brand,sub limit,and up limit

Filter by added filters, and clear added all filters

- Allow for registration

Registration display form

Interface - Secondary Tasks:

- Allow the user select a car and provide a payment type

Display form - with options for typical card information

- Allow for rent cancellation by customer

Only registered users can rent a car and only registered users can use the cancellation option

- Authenticate any user logging in

Communicates with the database to verify the inputted username and password is correct

- Select date of desired delivery,or date of planned pickup at nearest store location

Calendar options will be displayed that will allow the user to select directly on

- Select length of rental and planned drop-off date

Rental length is max 7 days,the delivery date is automatically added to the contract

Company-side Software- Primary Tasks:

- Track customer payments

Displays with the customer's information

- Keep and display available rental car records

Searching and viewing much like with customers

- Track customer payments

Displays with the customer’s information

- Keep and display available rental car records

Searching and viewing

Company-side Software- Secondary Tasks:

- View contract

Booking contract with details such as the customer's payment and rental car.

- Allow lease cancellation by system

2.3 Non Functional Requirements

Performance Requirements:

- Have a quick recovery time if anything were to go wrong

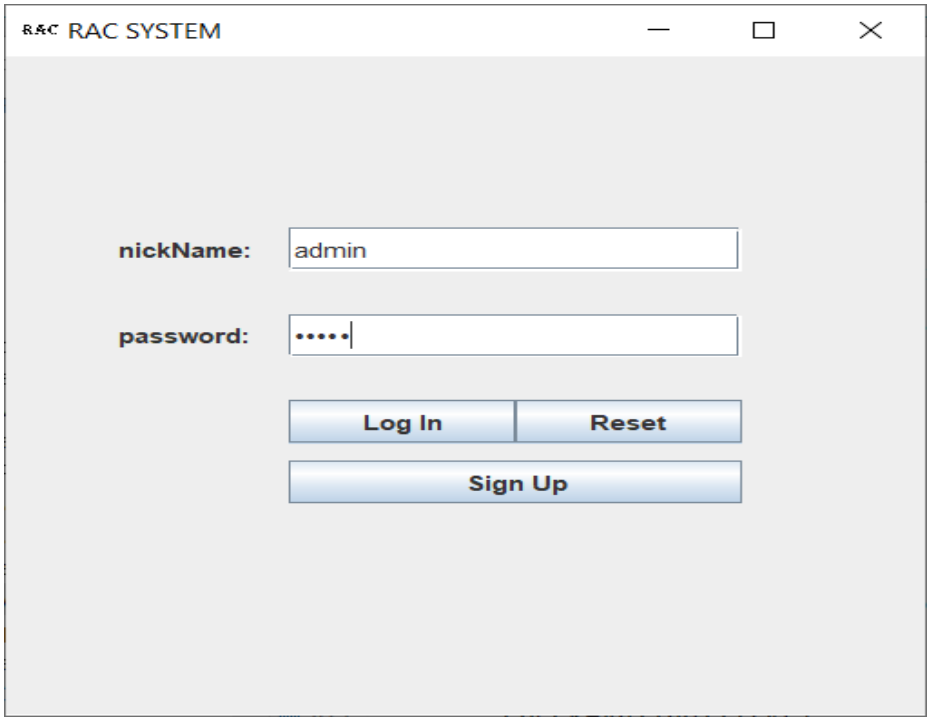
Security Requirements:

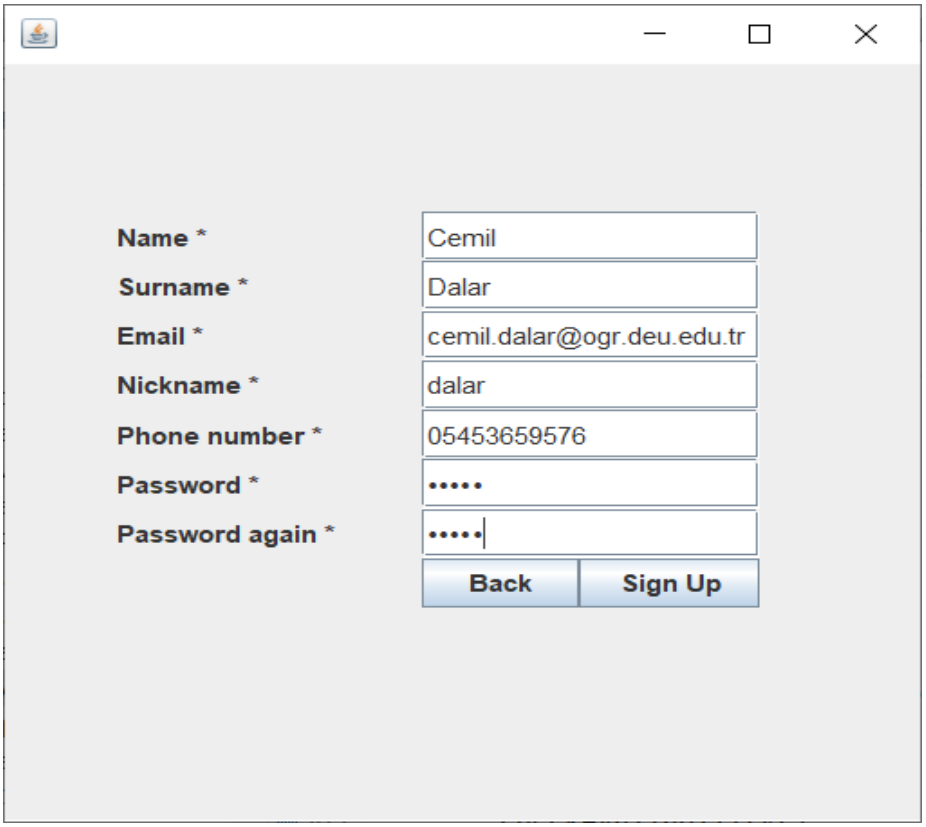
- Prevent any potential threats injections through the forms or search boxes.Prevent false information from being used as logging. ***

Quality Attributes:

- Maintainability : Over time, the project can be easily maintained.
- Availability : Resource availability is what resources you can use in the project, when they are available, and availability conditions.
- Usability: The abstract classes are used therefore, the project can be used even if the inputs change. User interface is explicit for easier use.
- Interoperability : This project does not require any external system.

Screenshot Mockups:





R&C Rent-A-Car

Rent | Add Car | Contracts | *Welcome admin* | Day+1 | 31/05/2022

CAR BRAND: SELECT | CAR TYPE: LUXURY | OFFICE NAME: SELECT | SUB LIMIT: 0 | UP LIMIT: 0 | Clear | Filter

TESLA
ROADSTER
35 ABC 477
3500
LUXURY
BUCA

Select

TESLA
MODEL3
35 BA 2407
3000
LUXURY
BORNOVA

Select

RENAULT
KADJAR
35 KL 360
1325
LUXURY
BUCA

Select

MERCEDES
GLA200
35 TT 1232
1550
LUXURY
ALSANCAK

Select

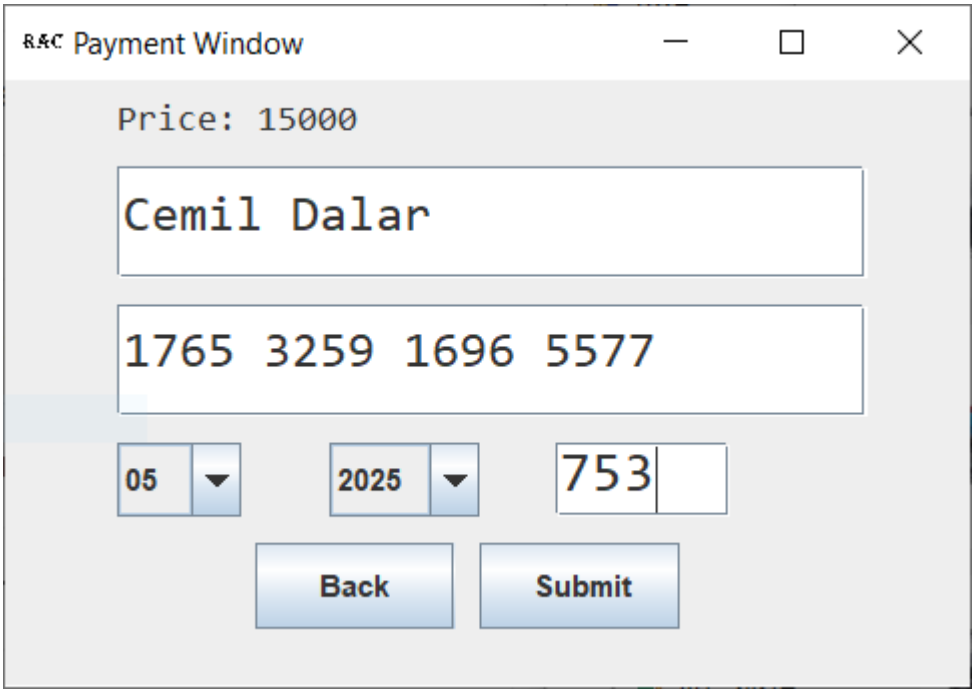
BMW
X5
35 TA 771
2250
LUXURY
ALSANCAK

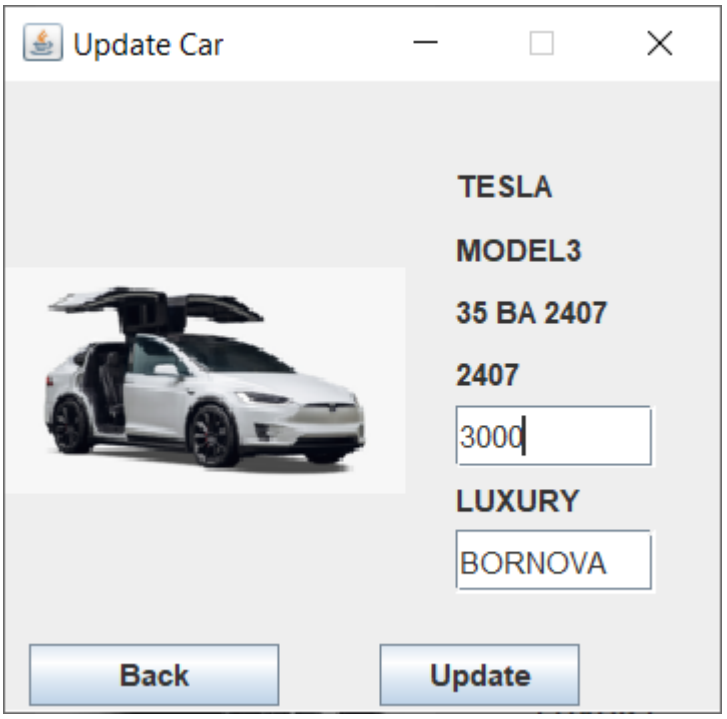
Select

BMW
520I
35 AAC 630
1450
LUXURY
MENEMEN

Select

ALPA:DEC | Sort | Prev | Next


R&C Payment Window
Price: 15000
Cemil Dalar
1765 3259 1696 5577
05
2025
753
Back
Submit


Update Car
TESLA
MODEL3
35 BA 2407
2407
3000
LUXURY
BORNOVA
Back
Update

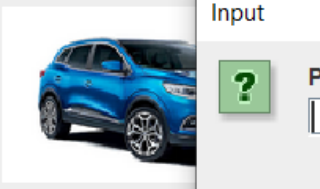
R&C Rent-A-Car

Delete | Add Car | Contracts | Welcome admin | Day+1 | 31/05/2022

CAR BRAND: SELECT | CAR TYPE: LUXURY | OFFICE NAME: SELECT | SUB LIMIT: 0 | UP LIMIT: 0 | Clear | Filter

TESLA
ROADSTER
35 ABC 477
3500
LUXURY
BUCA
Select

TESLA
MODEL3
35 BA 2407
3000
LUXURY
BORNOVA
Select



Select

Input

Please enter YES to delete the selected car.

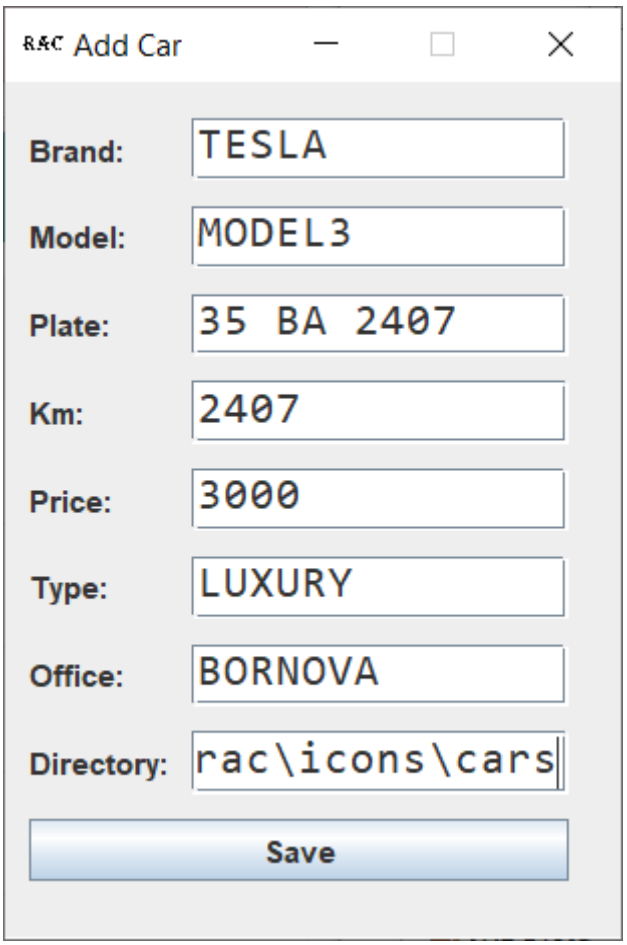
OK | Cancel

GLA200
35 TT 1232
1550
LUXURY
ALSANCAK
Select

BMW
X5
35 TA 771
2250
LUXURY
ALSANCAK
Select

BMW
520I
35 AAC 630
1450
LUXURY
MENEMEN
Select

ALPA:DEC | Sort | Prev | Next


R&C Add Car
Brand:
TESLA
Model:
MODEL3
Plate:
35 BA 2407
Km:
2407
Price:
3000
Type:
LUXURY
Office:
BORNOVA
Directory:
rac\icons\cars
Save

R&C Contracts

PEUGEOT
2008
35 CD 1711
BORNOVA
2500
17/05/2022
22/05/2022
Cemil Dalar

BMW
520I
35 AAC 630
MENEMEN
7250
14/05/2022
19/05/2022
Hasan Korkmaz

AUDI
A8
35 FC 6305
BUCA
3000
08/05/2022
13/05/2022
Halil Shapuk

TESLA
ROADSTER
35 ABC 123
BUCA
10500
19/05/2022
22/05/2022
Hasan Korkmaz

RENAULT
CLIO
35 FP 4481
ALSANCAK
1600
21/05/2022
26/05/2022
Cemil Dalar

PEUGEOT
301
35 SB 152
MENEMEN
330
15/05/2022
16/05/2022
Halil Shapuk

Prev Next

2.4 Use Cases

Use Case #1: Checking Car Status

Primary Actor : Customer

Goal in Context : To see if available car

Preconditions : If a car is available for selected reservation then customer can rent a car..

Trigger : The system knows the current information of cars and reservations.

Scenario:

1. Staff: Logs onto Car Rental Management system. (Enters Username/ Password)
2. Select the car that the customer wants.
3. Selects available cars.

Exceptions:

1. Car Rental Management System Username/Password incorrect: error message.

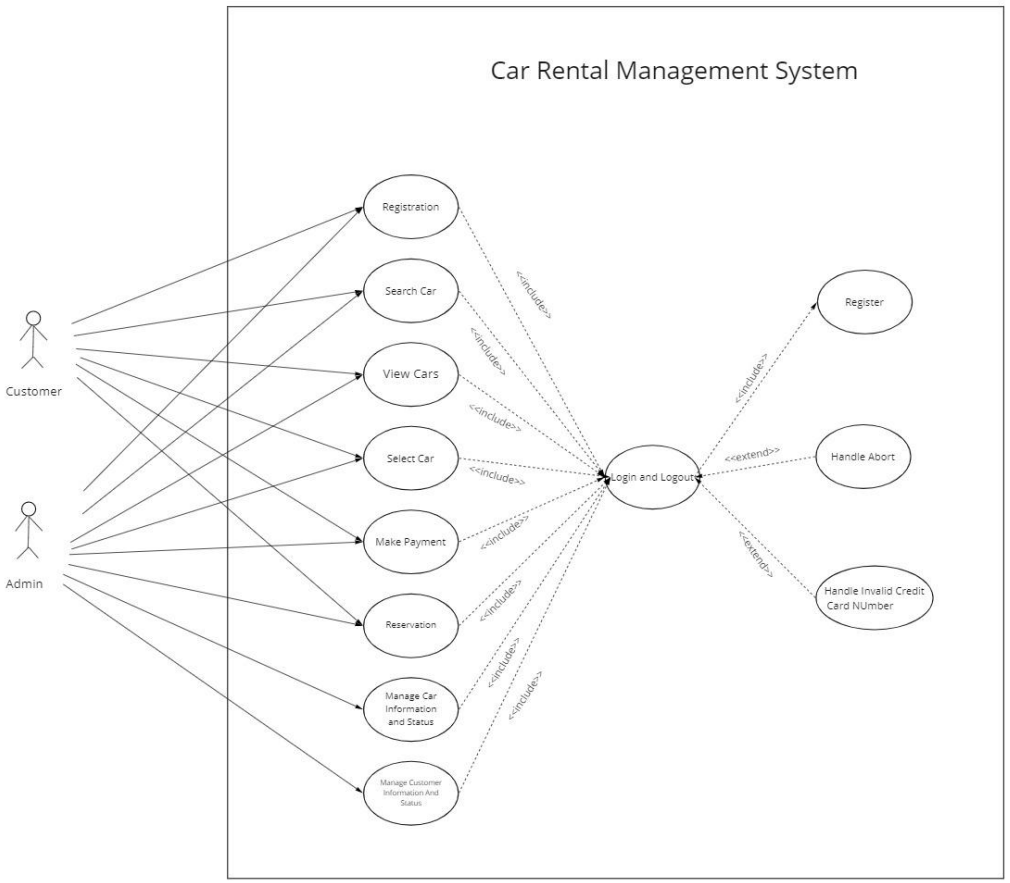
Priority: Essential, must be implemented

When available: Tickets can be sold.

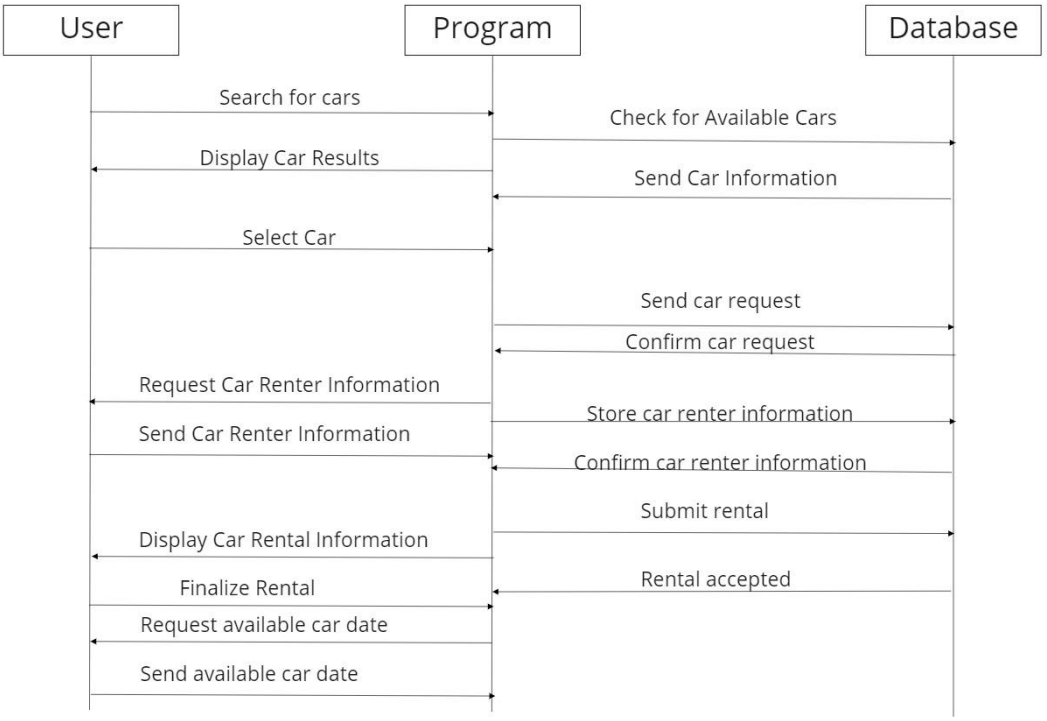
Frequency of use: Every time

Channel to actor: Car Rental Management System

2.5 Models


Car Rental Management System
Customer
Admin
Registration
Search Car
View Cars
Select Car
Make Payment
Reservation
Manage Car Information and Status
Manage Customer Information And Status
Login and Logout
Register
Handle Abort
Handle Invalid Credit Card NUmber
<<include>>
<<extend>>
miro

Sequence Diagram



CHAPTER THREE

3.Project Plan

3.1 Task Description

Design Models and Mockups

Designing the models and mockups help to ensure clarity in view of the project as well as how it works. Stakeholders are to sit through this process as drawings are created.

Database Creation

Customer information is kept in the database, needed information is obtained by reading from there.

Software Creation

This software has been created using visual and textual models and requirements that users (customers) and administrators can easily use. This software has been developed to easily find and rent the cars you want in a short time.

Testing

Test cases may be used to guide and understand the basic actions of customers. Any bugs or errors that occur will be identified and resolved.

Finalization and Reports

All testing and function processes are finalized at this stage. Reports will be created to ensure all information and functionality is clear in order to make the user manual and to help ensure employees can use the software with ease.

3.2 Task Assignment

After the groups were created, each group chose its own project topic. Three people actively participated in the brainstorming on project selection. All the presented project ideas were evaluated considering their potential positive and negative aspects. After the project was determined, they worked together while creating all the models. Meetings were held on process planning, task sharing in the software, helping each other in the problems encountered, or changing tasks by meeting frequently with 3 group members.

3.3 Deliverables and Milestones

We had four major Milestones in this project:

1. Completion of Requirements Gathering.
2. Completion of Design and code.
3. Completion of Testing.
4. Completion of Demonstration.

These milestones were all completed on schedule and yielded a Deliverable at the end of each.

3.4 Project Schedule

The first month of the project start date (April) was mainly to determine the requirements and this took 25 hours for about 2 weeks. In the last two weeks of April and part of May, the design took 40 hours.After the design started, we wrote the progress report. At the same time, testing started and continued throughout the rest of the project. This total time was about 75 hours. We then created our final report, which took about 10 hours. Final adjustments and corrections also took 8 hours.

The percentage distribution was as follows:

Requirements: 25 hours - 15.82%

Design and code: 40 hours - 25.31%

Test: 75 hours - 47.46%

Manual and Final Report: 10 hours - 6.32%

Display and Adjustments: 8 hours - 5.06%

Total: 158 hours - 100%

CHAPTER FOUR

4.Testing

4.1 Features to be Tested

The features we tested were as follows:

- Log-ins worked efficiently and consistently
- To ensure that the application itself ran
- Availability of the rented car by another customer
- Checking page load time

4.2 Test Cases

The following are examples of test cases we implemented:

- Amount values should be displayed with correct currency symbols
- Check all pages for broken images
- Check all pages for broken links

4.3 Testing Schedule

The testing should begin right after the project itself begins. Keeping up on testing will ensure that any mistakes are caught early and corrected immediately.

CHAPTER FIVE

5.Conclusion

5.1 The Problem and Solution

As a result of the planning, there was no problem that could not be met or solved.

5.2 The Team and the SE Process

The Software Engineering process we used was the waterfall method. In this method, plan and schedule all of the activities before starting working on them (plan-driven process). The phases of the waterfall model are: Requirements, Design, Implementation, Testing, and Maintenance.

5.3 Engagement of Umbrella Activities

Software project tracking and control:

When plans, tasks, models all have been done then a network of our team's tasks that will enable us to get the job done on time will have to be created.

Risk management:

Risk management is a series of steps that help a software team to understand and manage uncertainty. It's a really good idea to identify it, assess its probability of occurrence, estimate its impact, and establish a contingency plan that– 'should the problem actually occur'.

Document preparation and production :

Formal technical Reviews - This activity was essentially implemented for peer review. Having new and fresh eyes to view code and ensure that everything met the requirements.

5.4 The Stakeholder's that Benefited

After the release of the product, all of our active stakeholders benefited from the software. This list includes customer, admin, company and development team.

5.5 The Organization's Benefits

UML diagrams were used first when designing this project. Thanks to these diagrams, the project resulted in efficient time and resulted. These diagrams were used to model the system, to determine the status of the objects and to determine the requirements.

CHAPTER SIX

6.User Manual

6.1 Software Description

This software allows our customers to choose the right car for them by applying the filters they want in the available cars.Customers can rent a car for a maximum of 7 days. The delivery date is automatically formed in the contract.

6.2 How to use the Software

By entering the user information, the selected car can be rented by clicking on the desired car as a result of the filters. Profit and other variables are updated after each lease is created. Contracts, cars, customer records can be accessed and modified by the admin at any time.

6.3 Troubleshooting Common Problems

All possibilities were evaluated and no errors were encountered during the project process.