Software Requirements Specification

Version 1.0

<<Annotated Version>>

JUNE 18, 2020

Project for managing and organizing the car rental process in

EUROPE CAR COMPANY

Waad Ahmed

Razan AL-hameed

Submitted in partial fulfillment

Of the requirements of

CS 310 Software Engineering

<<Any comments inside double brackets such as these are *not* part of this SRS but are comments upon this SRS example to help the reader understand the point being made.

Refer to the SRS Template for details on the purpose and rules for each section of this document.

This work is based upon the submissions of the Spring 2020 CS 310. The students who submitted these team projects were Waad Ahmed ,Razan AL-hameed. >>

# Table of Contents

Table of content………………………………………………………………………………………i

List of Figures………………………………………………………………………………………...ii

1. Introduction………………………………………………………………………………………......
   1. Purpose……………………………………………………………………………………......
   2. Scope of Project……………………………………………………………………………….
   3. Glossary……………………………………………………………………………………......
   4. References……………………………………………………………………………………...
   5. Overview of Document………………………………………………………………………..
2. Overall description……………………………………………………………………………...

2.1 System Environment………………………………………………………………………………...

2.2 Functional Requirement Specification………………………………………………………………

2.2.1 Record vehicle data………………………………………………………………………….

Use Case: Record vehicle data…………………………………………………………………

2.2.2 Vehicle maintenance movement…………………………………………………………….

Use Case: Vehicle maintenance movement…………………………………………………….

2.2.3 Record customer data………………………………………………………………………..

Use Case: Record customer data………………………………………………………………..

2.2.4 Stop customer data ………………………………………………………………………….

Use Case: Stop customer data…………………………………………………………………..

2.2.5 Register and manage a reservation process………………………………………………...

Use Case: Register and manage a reservation process………………………………………...

2.2.6 Give authority to system users by administrator…………………………………………

Use Case: Give authority to system users by administrator…………………………………

2.3 User Characteristics………………………………………………………………………………...

2.4 Non-Functional Requirements……………………………………………………………………..

3.0 Requirements Specification..................................................................................................

3.1 External Interface Requirements………………………………………………………………….

3.2 Functional Requirements………………………………………………………………………….

3.2.1 Record vehicle data………………………………………………………………………..

3.2.2 Vehicle maintenance movement……………………………………………………………

3.2.3 Record customer data………………………………………………………………………

3.2.4 Stop customer data…………………………………………………………………………

3.2.5 Register and manage a reservation process……………………………………………….

3.2.6 Give authority to system users by administrator…………………………………………..

3.3 Detailed Non-Functional Requirements………………………………………………………….

### 3.3.1 Logical Structure of the Data……………………………………………………………

3.3.2 Security………………………………………………………………………………….

3.3.3 Easy and Speed……………………………………………………………………………

# List of Figures

Figure 1 - System Environment…………………………………………………………………………….

Figure 2 - Logical Structure of the Rental and greet car……………………………………………………

# 1.0. Introduction

## 1.1. Purpose

## 1.2. Scope of Project

## 1.3. Glossary

|  |  |
| --- | --- |
| **Term** | **Definition** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

## 1.4. References

## 1.5. Overview of Document

# 2.0. Overall Description

2.1 System Environment

There is a group of employees in this company with specific responsibilities and different tasks, it is better to equip the company with a computer network in addition to programs that take into account the multiplicity of branches and the responsibilities of each employee.

There is a general supervisor whose job is to general management of the company and monitor its branches, and another to maintenance, and there are those who receive and deliver cars from customers and many others ....

Where these responsibilities are integrated with the aim of achieving the required service from the company.

In addition to having a special server for storing the ***mysql*** database on which it stores k data for cars, customers and customers, it is connected to the system through the same computer network, which has specific permissions for each of the company’s departments, In addition to the system’s dealings with banks in order to pay via the bank to the customer, the system can deal with SMS messages (welcome - guidance - offers) for all customers.

Rental&Receipt car department

Maintenance department

manger

Mysql DB

Maintenance

Rental&Receipt

Administration

Accounting department

Mobile Network

Bank

#### Europe car System

accounting

Figure - System Environment

2.2 Functional Requirements Specification

## 2.3 User Characteristics

.

## 2.4 Non-Functional Requirements

# 3.0. Requirements Specification

## 3.1 External Interface Requirements

The only link to the external system is the mysql database in order to enter customer data on the one hand and auto data on the other hand in order to conduct the rental process correctly and accurately by entering the necessary data at each stage of the rental phase, for example we need to add a record for each A customer in the database and the author of the (triple name - profession - address - age - gender - nationality - the bank card information of the customer, id of the car that he rented) As for cars, they are already added to the database the moment they enter the company and their data is (name The car-id - type - engine number - manufacturer - part numbers ......)

## 3.2 Functional Requirements

The Logical Structure of the Data is contained in Section 3.3.1.

3.2.1 Record vehicle data

|  |  |
| --- | --- |
| Use case name | Record vehicle data |
| XRef | Section 2.2.1, Record vehicle data |
| Trigger | New car entry to the company |
| Precondition | The company requested a car from the supplier |
| Basic Path | 1. The responsible employee requests to add a new car to the system (Through the interface concerned with adding car data to the database). 2. The system displays the interface that contains specific fields representing the data required for each new car entering the company (name The car-id - type - engine number - manufacturer - part numbers ......). 3. The employee fills in the required fields and whoever confirms that data to be saved in the database. |
| Alternative path | The employee who has certain powers will enter the car data after the relevant interface appears and if he enters a duplicate ID (already present in the database) the system will send a message alerting that with the request to re-enter again, the system Scans and checks all data Then the operator knows that the data is correct and on the record. |
| Postcondition | The system will save the data of the new car in the database. |
| Exception path | The employee may undo saving the new car data before confirming the data |
| Other | The process of validating the data that is entered by the employee and which the system performs according to specific rules and models defined in the database tables |

3.2.2 Vehicle maintenance movement

|  |  |
| --- | --- |
| Use case name | Vehicle maintenance movement |
| XRef | Section 2.2.2, Vehicle maintenance movement |
| Trigger | New car underwent maintenance |
| Precondition | A vehicle has been involved in an accident or needs to be replaced due to damage or other external factors |
| Basic Path | 1. The maintenance department employee who deals with the system interfaces that are related to car maintenance and periodic checks have entered the identifier of the vehicle, name, and kind of it and that he wants to Maintains it. 2. The employee chooses the maintenance status (accident, periodic check, replacement parts). 3. The system displays the corresponding interface for each case, which includes the appropriate data fields for each maintenance operation. 4. The employee fills in the fields that appear in the interface corresponding to the maintenance status and confirms the entry process. 5. After confirming the entry process, the system generates a report that includes the information that the employee entered, including the date of the report, and sends a copy to the administration and keeps a copy of it on the employee's device. |

|  |  |
| --- | --- |
| Alternative path | The system shows the maintenance employee an interface requesting the ID, type and name of the vehicle that will be maintained after entering that data by the employee. The system will verify if the vehicle is present in the company by searching for it through an ID in the database (sending a query to the database Either if it is present, it will move to step 2, or if it is not available, a message will appear to the employee stating that this car does not exist. In step 2, if the employee chooses the maintenance status is an accident, the system will ask him for information regarding the date and time of the accident, the damage caused by it, the name of the tenant with whom the car was, and the fees that the tenant must pay. Then we go to step 5……. |
| Postcondition | The system generates the report with the entered information and sends a copy to the administration and a copy that remains on the employee's machine. |
| Exception path | The employee can tell the system that he does not want to have a copy of the report on his device in addition to that the maintenance process can be undone before confirming it. |
| Other | Maintenance cases and information corresponding to each case based on templates and information recorded by maintenance workers. |

|  |  |
| --- | --- |
| Use case name | Record customer data |
| XRef | Section 2.2.3, Record customer data |
| Trigger | Rent a customer for a car |
| Precondition | A new customer comes to the company and chooses a car to rent |
| Basic Path | 1. The receptionist asks the system to add a new customer (where the employee has authority for this process). 2. The system displays an interface that contains data fields that must be filled in by the employee (he asks the customer) and these fields are (triple name - profession - address - age - gender - nationality - bank card information for the customer). 3. The employee confirms the data he entered. 4. The system installed the data in a database. |

3.2. 3 Record customer data

|  |  |
| --- | --- |
| Alternative path | The system displays an interface that includes fields to collect information about the customer who wants to rent a car that the employee enters with the help of the customer and then he confirms it. The system then checks the validity of these entered data if it is correct, we move to step 4, or if it is not correct, it displays A letter to the employee regarding the fields that must be re-entered |
| Postcondition | The system installs customer data in the database. |
| Exception path | It is possible for the job to cancel the addition process, but before confirming the data. |
| Other | Customer data fields are determined based on the information required in the lease contracts and according to the schedule of customers in the database |

|  |  |
| --- | --- |
| Use case name | Stop customer data |
| XRef | Section 2.2.4, Stop Customer data |
| Trigger | Car customer delivery |
| Precondition | The customer's rental period ends, the car is delivered, and the due fees are paid, if any |
| Basic Path | 1. An employee in the delivery department takes the lease number that is given with the delivery receipt to the customer from a customer and enters it to the interface that requests it. 2. The system sends a query to the database, including the number of the entered contract. 3. The system displays an interface that requires the employee to confirm the suspension of customer data related to this contract. 4. The employee confirms the Suspension. |

3.2.4 Stop customer data

3.2. 4 stop customer data

|  |  |
| --- | --- |
| Alternative path | The system takes the contract number from the employee and sends a query that includes it to the database in order to bring the data the customer who owns the number in case that customer does not find then he informs the employee by message and in the solution he found these data we move to step 3. |
| Postcondition | The system stops customer data. |
| Exception path | The employee cannot undo the deletion process, so caution must be exercised. |
| Other | ……………………………. |

3.2. 3 Record customer data

|  |  |
| --- | --- |
| Use case name | Register and manage a reservation process. |
| XRef | Section 2.2.5, Register and manage a reservation process |
| Trigger | Rent a customer for a car. |
| Precondition | Confirm customer data in the reception section and choose a specific vehicle. |
| Basic Path | 1. The employee requests to organize a lease contract from the lease. 2. The system displays the interface for the rental contract that contains the following fields (car name - model - number - payment type - customer name - rental date ......). 3. The employee enters the required data and confirms it. 4. The system links this customer to the contract he made with the company in the database and then prints a copy of this contract for the customer, in addition to printing a receipt for the selected car. |

3.2.5 Register and manage a reservation process

|  |  |
| --- | --- |
| Alternative path | The system organizes an electronic rental contract immediately upon the customer's request to do so, and then waits for data entry by the employee in order to confirm the rental process and make the necessary adjustments to the database such as linking the customer to the contract that was established with him through the number of this contract, which will be printed on the receipt receipt also, After entering the required data, the system will verify this data and after that we will proceed to step 4. |
| Postcondition | Printing the lease contract for both the customer and the company. |
| Exception path | The rental process can be undone but before confirming the information |
| Other | The information required by the system to regulate the lease contract based on the paper lease contract templates. |
| 3.2.6 Give authority to system users by administrator  Use case name | Give authority to system users by administrator | |
| XRef | Section 2.2.6, Give authority to system users by administrator | |
| Trigger | Grant or revoke authority from a specific user | |
| Precondition | The system administrator has requested to grant to revoke the authority of a specific user. ( The user must be the system administrator in order to take this action  ) | |
| Basic Path | 1. The system administrator, through the interface designated for the manager, enters the manager's password, which is given to the system upon installation. 2. The system verifies the password and then the administrator enters the interface, which contains a general administration for the entire system and the entire departments. The manager chooses to withdraw or grant the authority of an employee. 3. The system requires the manager to enter the name of an employee, the name of his department and the employee's ID .The system checks the data entered by the manager and then appears an interface that contains all the powers of this employee with the possibility of granting him a new authority. 4. The manager removes a certain authority by canceling her selection only or adding a new one by choosing it. | |

3.2.6 Give authority to system users by administrator

|  |  |
| --- | --- |
| Alternative path | The manager enters the special password and the system sends a query to the database to make sure of  This password belongs to the manager. If this password is not found, an alert will be sent to the manager that the password is not correct. Either way, the destination system for the manager will be opened and we will move on. Step 3. |
| Postcondition | Withdraw or grant the specified powers from the administrator to a specific user. |
| Exception path | The manager can undo the granting or withdrawal process at any moment |
| Other | The powers that the system provides to the manager are based on the powers of each department in the company |

## Detailed Non-Functional Requirements

### 3.3.1 Logical Structure of the Data

The logical structure of the data to be stored in the MySQL database is given below

rental

car

customer

greet

**Figure 2 - Logical Structure of the Rental and greet car**

The data descriptions of each of these data entities is as follows:

**Car data entity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| Name | Text | Name of the car |  |
| Kind | Text | Type of the car |  |
| id | Integer | Serial number for car |  |
| Engine serial Num | Integer |  |  |
| Manufacture company | Text | The company that launched the car |  |
| Production date | date | The date that car was launched |  |
| status | Boolean | Is the car rented or not? |  |
| Engine speed | float |  |  |
| Cutting number | Integer |  |  |

**Customer data entity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| Full Name | Text | Full Name of the customer |  |
| Work | Text | The customer profession |  |
| id | Integer | Serial number for customer |  |
| address | Text | Customer address(city, district, street name, number of building) |  |
| Age | Text | Customer age in years |  |
| Gender | Text | Male or female |  |
| Nationality | Text | Syrian or not? |  |
| Bank card Number | Integer | For payment process |  |
| Contract number | Integer | Lease contract Number |  |

3.3.2 Security

The server on which the MYSQL data base will have its own security to prevent unauthorized *write*, read And *delete* access.

The PC on which the administrator resides will have its own security. Only the mangers of departments and administrator will have physical access to the machines and the program on it.

3.3.3 Easy and Speed

The server that holds the database must be fast to perform the operations of reading, deleting and modifying quickly, and the system should also have an easy and beautiful interface for the concerned employees to work on the system with ease.