Cairo University  
Faculty of Computers and Artificial Intelligent

**CS251 - Software Engineering I**

Project Name

Software Requirements Specifications (SRS)

|  |
| --- |
| **Team Names** |
| Sama Hussien Abo Elala |
| Eman Fathy Abo Alhassan |
| Sara Ahmed Sayed |
| Mohamed Ayman |

June&2022

Contents

[Instructions [To be removed] 3](#_Toc101814799)

[Team 3](#_Toc101814800)

[Document Purpose and Audience 3](#_Toc101814801)

[Introduction 3](#_Toc101814802)

[Software Purpose 3](#_Toc101814803)

[Software Scope 3](#_Toc101814804)

[Definitions, acronyms, and abbreviations 3](#_Toc101814805)

[Requirements 4](#_Toc101814806)

[Functional Requirements 4](#_Toc101814807)

[Non Functional Requirements 4](#_Toc101814808)

[System Models 4](#_Toc101814809)

[Use Case Model 4](#_Toc101814810)

[Use Case Tables 5](#_Toc101814811)

[Ownership Report 6](#_Toc101814812)

[Policy Regarding Plagiarism: 6](#_Toc101814813)

# 

# Team

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Email** | **Mobile** |
| 20200232 | Sama Hussien Abo Elala | 11410120200232@stud.cu.edu.eg | 01129552444 |
| 20200105 | Eman Fathy Abo Alhassan | 11410120200105@stud.cu.edu.eg | 01111970653 |
| 20200214 | Sara Ahmed Sayed | 11410120200214@stud.cu.edu.eg | 01154688992 |
| 20200432 | Mohamed Ayman | 11410120200432@stud.cu.edu.eg | 01095891283 |

# Document Purpose and Audience

This SRS describes the specifications of the project.

The document is intended for requirements engineer, domain expert, developer, tester and project manager.

# Introduction

## Software Purpose

easy to make park in garage operation. parking management software is used to optimize parking space, manage the influx of cars, make the safety of cars and people and calculate the fees for each driver park in the garage. Software is used to calculate the total income and the number of vehicles which the admin wants to know and used to know the available slots.

## Software Scope

Software help drivers to park in garage and park out and enter the vehicle details and can update vehicle details also.

Software help garage owner or the administrator to manage the income, the number of vehicle park in the garage, display available slots and to manage the number of slots and the dimensions of each slot for first time only.

## Definitions, acronyms, and abbreviations

# Requirements

## Functional Requirements

1. The system should search for the available slots.
2. The system should pick a free slot based on two configurations

(i)first come first served.

(ii)best-fit approach.

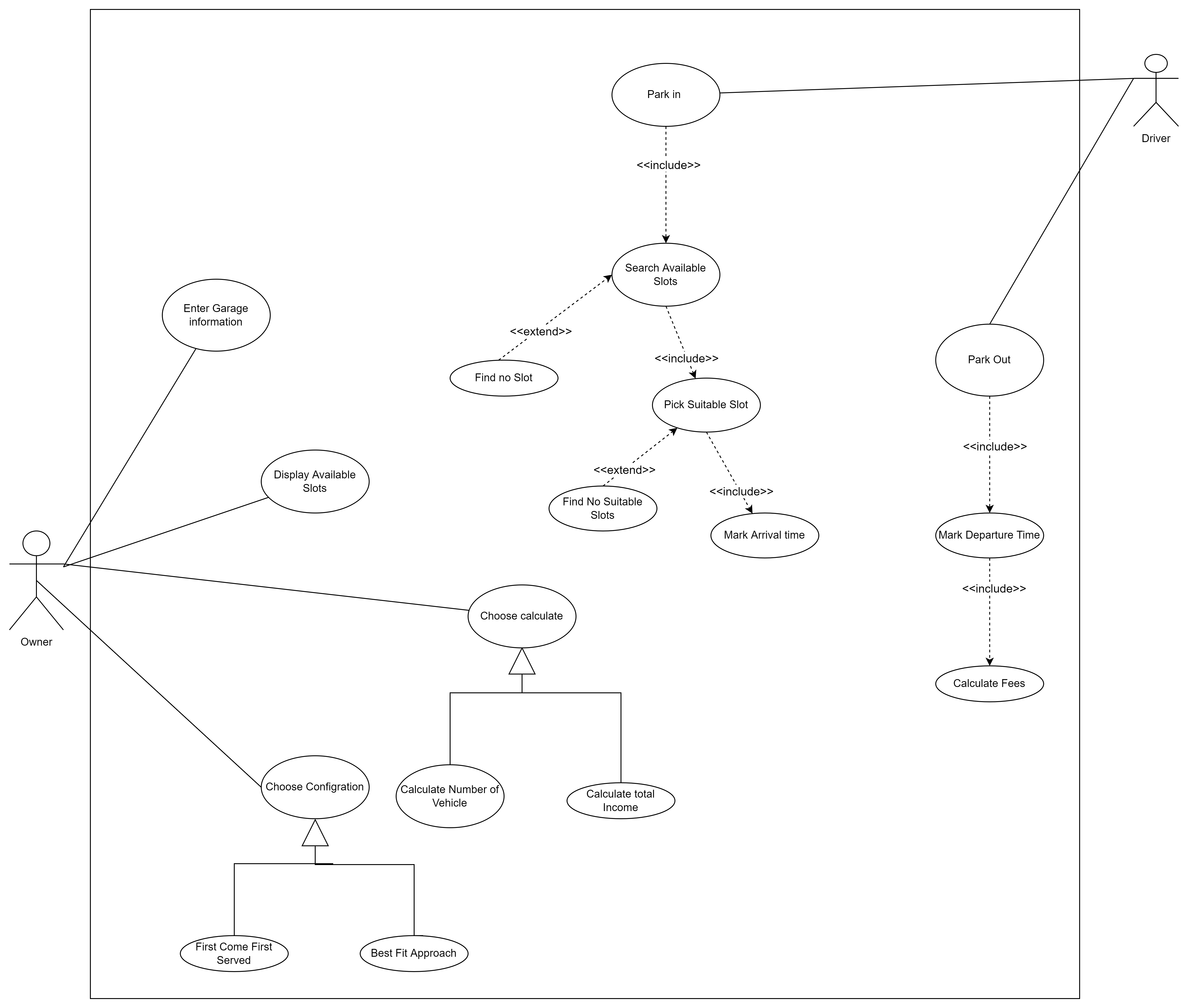
1. The system should display the suitable and free slot for the user to park in, otherwise display message to show that there is no empty slot.
2. The system should mark arrival time of the vehicle.
3. The system should mark Departure time of the vehicle from the garage.
4. The system should calculate time-of-stay from the arrival time and departure time.
5. The system should calculate parking fees based on the time-of-stay.
6. The system should display the available parking slots if the actor needs.

## Non Functional Requirements

|  |  |
| --- | --- |
|  | **Details** |
| **Performance** | Each operation will be done through 30 second whether park-in or park-out or any calculation or payment or sign up. |
| **Scalability** | The system should be able to support up to 30 simultaneous customers. |
| **Usability** | The customers can learn, operate, prepare inputs, and interpret outputs through interaction with the system. |
| **Reliability** | The system should perform the specified functions without failure. |
| **Survivability** | How well the system continues to function and recover in the presence of a system failure. |
| **Availability** | We can depend on the system's ability to function during normal operating time by the percentage of 90%. |

# System Models

## Use Case Model



## 

## Use Case Tables

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 1 | |
| Use Case Name: | Enter garage information. | |
| Actors: | The owner. | |
| Pre-conditions: | The owner clicked on Garage management button. | |
| Post-conditions: | The owner will enter the configuration that he wants. | |
| Flow of events: | **User Action** | **System Action** |
| 1-The owner enters the number of slots that exist in his garage.  2-The owner enters the details of each slot. |  |
|  | 3-The system will store the information of each slot. |
| 4-The owner clicks on display available slots to ensure that the system store all the information about the slots. |  |
|  | 5-The system will display the available slots. |
| 6-The owner clicks on choose configuration. |  |
|  | 7-The system will ask him for choosing option such as first come first served slot or best-fit approach. |
| 8-The owner choose the first option which is first come first served. |  |
| Exceptions: | **User Action** | **System Action** |
| 1-The owner clicks on display the available slots. |  |
|  | 2-The system display the slots. |
| 3- the owner finds the slots that the system displayed differs from the slots that he inserted  4-The owner enter the details of each slot again. |  |
|  |  | 5-The system store them. |
|  | 6-The owner click display available slots |  |
|  |  | 7-The system display him the slots he enters. |
| Includes: | Enter garage info include display available slots. | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 2 | |
| Use Case Name: | Park in garage. | |
| Actors: | The driver | |
| Pre-conditions: | The driver wants to park in the garage | |
| Post-conditions: | The driver park in garage. | |
| Flow of events: | **User Action** | **System Action** |
| 1-The drive check the available slots. |  |
|  | 2-The system return there is available slots |
| 3-The driver enters his information (phone and number)  4- The driver enters his vehicle information |  |
|  | 5-The system returns a message to show his info and his car info are stored successfully. |
| 6-The driver clicks park-in button |  |
|  | 7-The system marks the arrival time. |
| 8- The driver clicks pick suitable button |  |
|  | 9-The system perform the first configuration which is first come first served, as the owner selected this configuration while entering the garage information  10-The system displays the slot ID that he should park in it. |
| Exceptions: | **User Action** | **System Action** |
| 1- the driver clicks check available slot button. |  |
|  | 2- System search available slot.  3- System show error message.  4-System show message come soon. |
| Includes: | Park in include check available slot, pick a suitable slot, mark arrival time. | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 3 | |
| Use Case Name: | Calculate total income. | |
| Actors: | The owner | |
| Pre-conditions: | The owner wants to know the total income and total number of vehicles. | |
| Post-conditions: | The owner knows the total income at specific period time. | |
| Flow of events: | **User Action** | **System Action** |
|  | 1- System ask the owner for choosing option such as calculate total income or calculate number of vehicles. |
| 2- The owner chooses to calculate the total income. |  |
|  | 3-System return total income. |
|  | 4- The owner chooses to calculate number of vehicles. |  |
|  |  | 5-System return the number of vehicles. |
| Exceptions: | **User Action** | **System Action** |
| 1-The owner chooses to calculate total income. |  |
|  | 2- System show error message about there is no income. |
| 3-The owner choose calculate number of vehicles. |  |
|  | 4- System show error message about there is no vehicles. |
| Includes: | Include choose to calculate total income or number of vehicles. | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 4 | |
| Use Case Name: | Park out | |
| Actors: | The driver | |
| Pre-conditions: | The driver wants to leave the garage. | |
| Post-conditions: | The driver park out of garage. | |
| Flow of events: | **User Action** | **System Action** |
| 1- the driver click park out button. |  |
|  | 2- System mark departure time.  3-System calculate the duration.  4- system calculate fees and display it to the driver. |
| Exceptions: | **User Action** | **System Action** |
| 1- the driver click park out button. |  |
|  | 2- System mark departures time.  3- System can't find the arrival time to calculate the duration  4-System display message to tell the driver that he whether is not park in our garage or he should close the system and open it again |
| 5-The driver close the program and open it again.  6-Driver click on park out button |  |
|  | 7- System mark departure time.  8--System calculate the duration.  9- system calculate fees and display it to the driver. |
| Includes: | Park out includes mark departure time, calculate fees | |
| Notes and Issues: |  | |

# Ownership Report

|  |  |
| --- | --- |
| **Item** | **Owners** |
| Functional, non-functional Requirements, use-case diagram | *Sara Ahmed* |
| Use case descriptions,introduction | *Eman Fathy* |