Cairo University  
Faculty of Computers and Artificial Intelligent

**CS251 - Software Engineering I**

Parking System

Software Requirements Specifications (SRS)

Mennatallah Mohammed Ibrahim El Sharawy

Alaa Ashraf Youssef

Toka Hamdy Mohammed

Omar Magdy Adel

May/2022

Contents

[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** |
| 20201048 | Toka Hamdy Mohammed | doudyhamdy.th@gmail.com | 01140640015 |
| 20201178 | Mennatallah mohammed Ibrahim | menahmohamed2002@gmail.com | 01068924028 |
| 20200329 | Alaa Ashraf Youssef |  | 01006873341 |
| 20200355 | Omar Magdy Adel |  | 01124273671 |

# Document Purpose and Audience

* **This document is responsible for the software requirements specifications where it contains the functional and non-functional requirements which describes what does the system mainly do and how does this system actually works so that the audience of the document could understand what the system is and also contains the use case diagram model with use case descriptions**

**Which gives as a brief about the main functionalities of the system and the participating actor in that system. It achieves the requirements for both, the customer and the garage owner.**

* **The audience for this document is the customer (Garage Owner) who will be using this system.**

# Introduction

## Software Purpose

* **The purpose of that software is that the parking system should provide the customer with the facility of parking his vehicle easily and should show the owner the progress of the system and the income that comes from the parking vehicles and to use the different functionalities of the system to let the customer park his vehicle.**

## Software Scope

* **This software application manages a parking space for a certain number of vehicles**
* **The system should let the customer park in the available parking slots**
* **It also lets the customer to park out from the parking garage easily and this can be done by the garage owner who will use the system to do so and calculates the parking fees for the customer to checkout.**
* **It also calculates the total income of all the vehicles that are parking at the parking slots and the total number of the vehicles which is important for the garage owner to know.**

## Definitions, acronyms, and abbreviations

NONE

# Requirements

## Functional Requirements

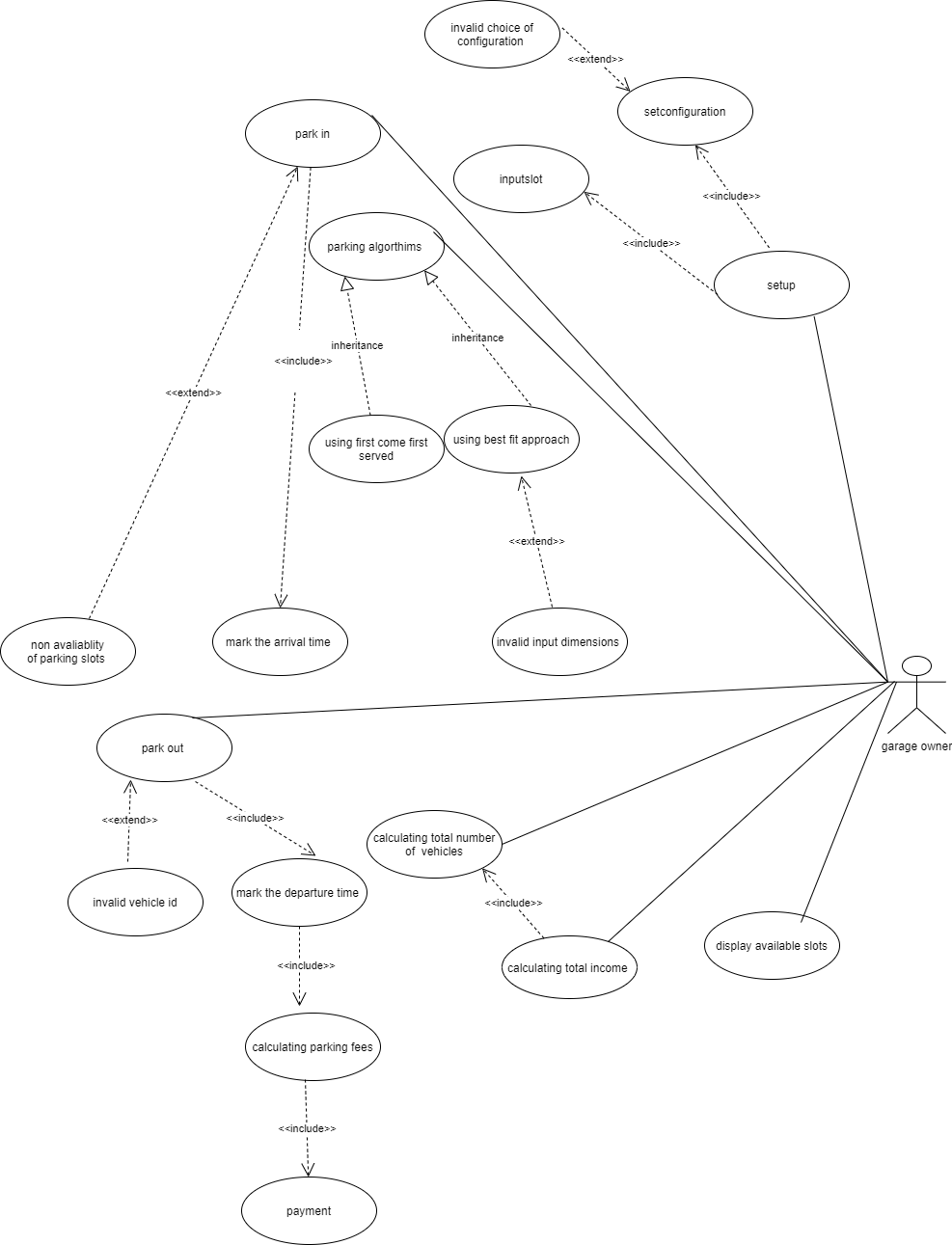
* **Car owner has the ability to park his car easily through the available parking slots (spaces) and this can be done by entering the vehicles information into the system.**
* **Car owner will also park out his vehicle easily through giving the unique id of the vehicle to the system.**
* **Garage owner has the ability to choose one of the two different configurations which are provided by the system at first while starting using the application and then he can change the configuration he has selected at first after that.**
* **Garage owner could choose the first configuration to park the vehicle of the car owner with, where this configuration is first come first served, if the garage owner selected this method, the system will choose the first free or available slot in the parking spaces.**
* **Also garage owner could select the other configuration instead which is best fit approach, if this configuration is selected by the garage owner, system will find the slot with the minimum dimension to hold the vehicle of the customer.**
* **Garage owner has also the ability to display the available slots which are found at the parking space.**
* **System also provides two functionalities which are calculating the total income and total number of the vehicles which are parking at the parking slots.**

## Non Functional Requirements

|  |  |
| --- | --- |
|  | **Details** |
| **Performance** | * **The parker should have a place to park in with 30 second.** * **System rejects the invalid input.** * **A successful will displayed that the process has been done.** |
| **Usability** | * **Garage owner will not click more than 5 clicks to do any function.** |
| **Reliability** | * **An Error message will be displayed if owner entered invalid input** |
| **Robustness** | * **System maintains available even if the Owner has input invalid input.** |
| **Availability** | * **System is down not more 5 times a week.** |
| **Security** | * **Customers can not access other customer’s info.** |

# 

# System Models :Use Case Model

****

## Use Case Tables

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 1 | |
| Use Case Name: | setup | |
| Actors: | Garage owner | |
| Pre-conditions: | Garage owner should start opening the application and setup the configuration that should be used and should consider the number of slots that will be on the parking garage | |
| Post-conditions: |  | |
| Flow of events: | **User Action** | **System Action** |
| 1-garage owner start to set a configuration for the system. |  |
|  | 2- System will save the configuration that is chosen by the garage owner and will park the vehicles according to it unless the garage owner has set it up again. |
| 3- Garage owner will also choose the number of the slots to be in the parking garage and will enter the depth and width for each slot. |  |
|  | 4- System will save the data of the slots which is entered by the garage owner and will set it as the default one unless the garage owner has set it up again |
|  |  |
| Exceptions:  In case the garage owner has entered an invalid choice of configuration | **User Action** | **System Action** |
| 1- garage owner entered an invalid choice for the configuration |  |
|  | 2- system will display an error message saying that the input choice is invalid  3-And the system will let the garage to enter a choice again until it’s valid. |
| Includes: | Setconfiguration, inputslots | |
| Notes and Issues: | Usability: •Parker or Garage owner will not click more than 5 clicks to do any function.  Reliability: •An Error message will be displayed if owner entered invalid input | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 2 | |
| Use Case Name: | Park in | |
| Actors: | garage owner | |
| Pre-conditions: | The parker should enter valid information about his vehicle and there are free slots available to let the parker park in | |
| Post-conditions: | The parker has successfully parked his vehicle in one of the slots available | |
|  | **User Action** | **System Action** |
| 1- garage owner will enter the whole information about the parker’s vehicle |  |
|  | 2- the system will get this data |
| 3- garage owner will ask to display available slots |  |
|  | 4- System will display the available slots |
|  |  |
|  | 5-according to the chosen configuration at first, the system will check first whether there are free slots or not  6- system will then save data of the vehicle  7- system will display a successful message for the parker to make sure that he can finally park his vehicle  8-system will detect the arrival time of the vehicle |
| 9-the parker will start to park his vehicle in the slot |  |
|  | 10-system will then set the slot off (the parker has parked in) |
| Exceptions:  In case there is no free slots available | **User Action** | **System Action** |
| 1-the parker will not be able to park in |  |
|  | 2-system will display an error message to the garage owner that he can’t park the parker’s vehicle at that time. |
|  | 3-**parker can come at another time to park where may at that time there are free slots** |  |
| Includes: | Mark the arrival time | |
| Notes and Issues: | Performance: • The parker should have a place to park in with 30 second. | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 3 | |
| Use Case Name: | First come first served | |
| Actors: | Garage owner | |
| Pre-conditions: | Garage owner wants to the change the configuration which was selected before. | |
| Post-conditions: | The Vehicle has been parked using First come first served configuration | |
| Flow of events: | **User Action** | **System Action** |
| 1- Garage owner selects First come first served approach to park the vehicle. |  |
|  | 2- system will choose this parking configuration as the parking configuration for all next vehicles  3-system will then choose the first available slot.  4- A successful message will be displayed by the system that the configuration has been chosen |
| 4-parker will park his vehicle in the chosen slot |  |
| Exceptions:  None | None | None |
| Includes: | None | |
| Notes and Issues: | Performance: • The parker should have a place to park in with 30 second. | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 4 | |
| Use Case Name: | Best-Fit Approach | |
| Actors: | Garage owner | |
| Pre-conditions: | Garage owner wants to the change the configuration which was selected before. | |
| Post-conditions: | The vehicle has been parked using Best-fit configuration | |
| Flow of events: | **User Action** | **System Action** |
| 1- Garage owner selects Best-fit approach to park the vehicle |  |
|  | 2- System will ask for width and depth of the vehicle to choose the best slot fits the vehicle |
| 3- Garage owner will enter the width and depth of the vehicle |  |
|  | 4- System choose the best slot fits the vehicle  5-A successful message will be displayed by the system that the slot has been chosen |
| 6-Driver will park his vehicle in the chosen slot |  |
| Exceptions:  Invalid input dimensions | **User Action** | **System Action** |
| 1- garage owner entered invalid input dimension of the vehicle |  |
|  | 2- System can not choose a slot correctly  3- An Error message will be displayed which says that the system could not choose a slot for the vehicle and the garage owner should enter valid dimensions(within a certain range ) |
| Includes: | None | |
| Notes and Issues: | Performance: • The parker should have a place to park in with 30 second. | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 5 | |
| Use Case Name: | park out | |
| Actors: | garage owner, parker | |
| Pre-conditions: | The parker wants to park out | |
| Post-conditions: | The parker parked out | |
| Flow of events: | **User Action** | **System Action** |
| 1- the garage owner asks the parker for his vehicle ID |  |
|  | 2- the parking system Verifies the parker ID. |
|  | 3- The parking system mark the departure time and save it to the parker’s vehicle information. |
|  | 4- The parking system calculate the vehicle fees and print it. |
| 5-the garage owner takes the money from the parker in cash |  |
| Exceptions: | **User Action** | **System Action** |
| 1- the garage owner asks the parker for his vehicle ID |  |
|  | 2-The vehicle ID is invalid!  3- The system rejects the ID and ask the parker to Enter the correct ID. |
| Includes: | Mark the departure time, calculate parking fees, payment | |
| Notes and Issues: | Performance: • System rejects the invalid input.  Reliability : •An Error message will be displayed if owner entered invalid input | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 6 | |
| Use Case Name: | Calculate parking fees. | |
| Actors: | garage owner (participate) | |
| Pre-conditions: | the parker is parking out. | |
| Post-conditions: | The parker is paying. | |
| Flow of events: | **User Action** | **System Action** |
|  | 1-The parking system search for the vehicle by its ID. |
|  | 2- The parking system select the arrival time and the departure time of the vehicle. |
|  | 3- The parking system calculates the total time did the vehicle stay. |
|  | 4- The parking system calculate the fees by multiply the total time (in hours) by 5. |
|  | 5- the system displays the parking fees. |
| Exceptions: | **User Action** | **System Action** |
|  |  |
|  |  |
| Includes: | Payment. | |
| Notes and Issues: | Performance: **A successful will displayed that the process has been done.** | |

6-

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 7 | |
| Use Case Name: | Calculate total income | |
| Actors: | Garage owner | |
| Pre-conditions: | Garage owner should select calculating the total income functionality | |
| Post-conditions: | The total income will be displayed for the garage owner | |
| Flow of events: | **User Action** | **System Action** |
| 1-garage owner selects calculating the total income functionality |  |
|  | 2- System will start calculating the total income  3- Where the system will get the parking fees for all the vehicles (which are parking at the parking slots) which were saved before by the system.  4- System calculates the total income by getting the sum of the parking fees of all vehicles which were parking in the parking slots. |
|  | 5-Then the system will display a message to the garage owner which represents the calculated total income |
|  |  |
|  |  |
| Exceptions:  NONE | **User Action** | **System Action** |
| NONE | NONE |
|  |  |
| Includes: | Calculate total number of vehicles | |
| Notes and Issues: | Performance: • A successful will displayed that the process has been done.  Usability: •Parker or Garage owner will not click more than 5 clicks to do any function. | |

# Ownership Report

|  |  |
| --- | --- |
| **Item** | **Owners** |
| Functional requirements, NON-Functional requirements, use case model and use case tables. | *All team have participated in doing it* |
|  |  |