Cairo University  
Faculty of Computers and Artificial Intelligent

**CS251 - Software Engineering I**

Parking System

Software Requirements Specifications (SRS)

Kevin Marco marcory

Michael Maher Nagi

Mariam Tarek Galal

Hagar Abd-Elhamed Mohamed Ali

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** |
| 20200401 | Kevin Marco marcory | kevinmarco9000@gmail.com | 01202914566 |
| 20200576 | Michael Maher Nagi | [Michael.maher.nagi@gmail.com](mailto:Michael.maher.nagi@gmail.com) | 01285382191 |
| 20200522 | Mariam Tarek Galal | mmaarriiaammttaarreekk@gmail.com | 01006056040 |
| 20210558 | Hagar Abd-Elhamed Mohamed Ali | Habdelhameed585@gmail.com | 0115206902 |

# Document Purpose and Audience

* **This document is a software requirements specification for a parking garage system and the main purpose is to facilitate the finding of Slots and payment process.**
* **The target audience to read this document is the garage owner and the developers.**

# Introduction

## Software Purpose

* **This software is automated garage parking system that handles parking and managing garage slots.**
* **Facilitating the payment process by using cash and the system allows the change to be returned in case of excess.**

## Software Scope

* **This system allows the driver to park-in with selected configuration (first come first served slots OR best-fit) Selected by the Owner and the system allows the owner To Sets Dimensions of each slot in the garage and the Dimensions of Garage.**
* **And the system searches for available slots with some functions and display it with different color.**
* **The system allows to the driver to park-out from the garage after pays the required money.**
* **The system calculates the fees by knowing entry and departure Time of the car from the garage.**

## Definitions, acronyms, and abbreviations: None

# Requirements

## Functional Requirements

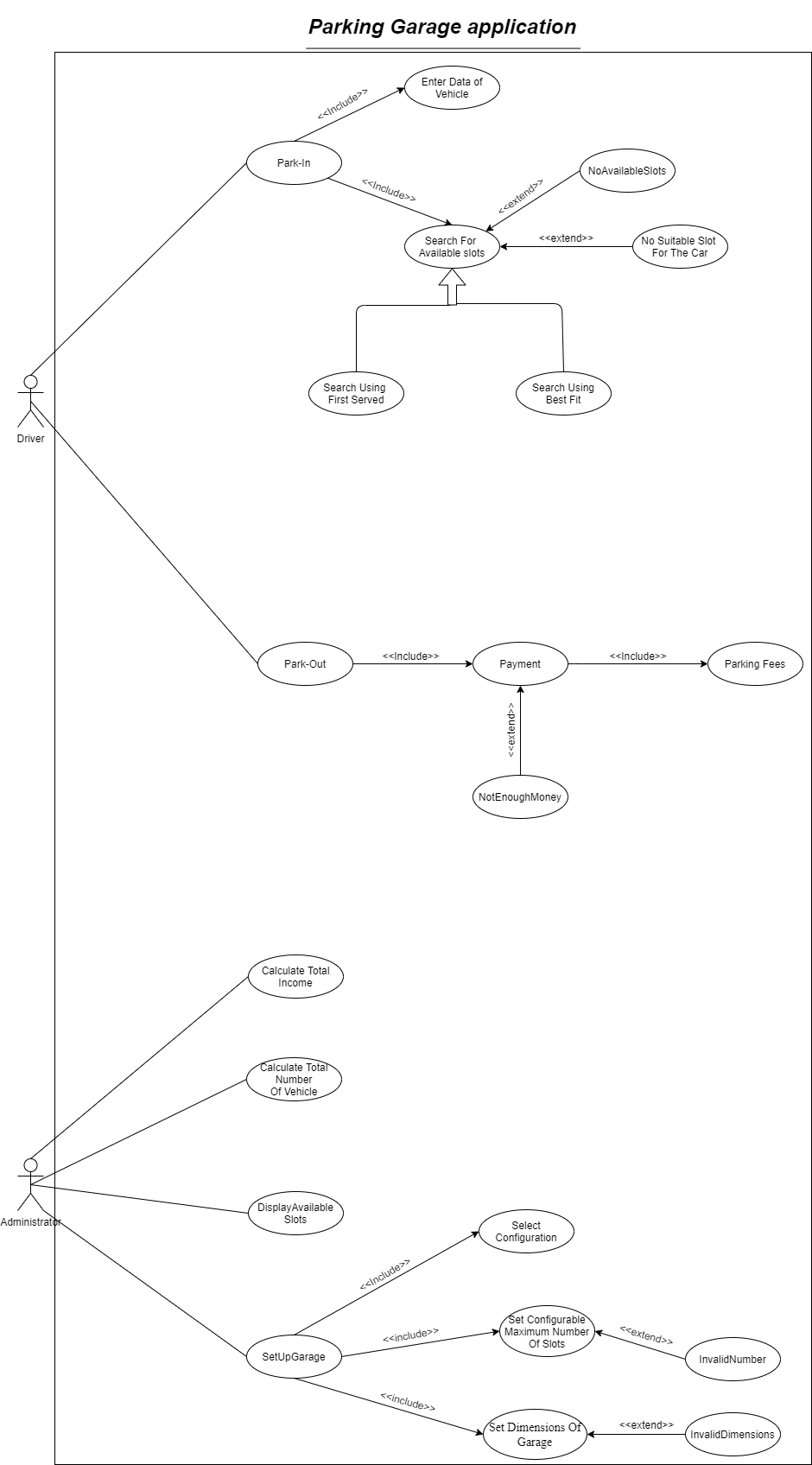
* **Park in**
* **Park out**
* **Parking fees**
* **Total income**
* **Total number of vehicles**
* **Display available slots**
* **Search For available slots**
* **Configuration**
* **Enter Data of Vehicle**
* **Payment**
* **Setup Garage**
* **Set Numbers of Slots**
* **Set Dimensions of Garage**
* **Set Configurable Maximum Number of Vehicles**

## Non-Functional Requirements

|  |  |
| --- | --- |
|  | **Details** |
| **Performance** | * **(Time response): The system should display the available slots in 3 sec as maximum.** * **Park out should take place no more than 7 seconds.** |
| **Reliability** | * **The system should check the money entered by the user is enough for the total number of hours user stays within an hourly rate. if the entered money is less than the total fees the system will ask the driver to enter money again, else if the entered money is greater than the needed money the system will return for the user the rest of the money** * **The system should check if there is an available slot or not if there are no available slots the system will display for the driver a message and the process will cancel** * **The system should check the dimensions, number of slots, cancel negative numbers and display appropriate exception for the user (“Negative Numbers not allowed”).** * **The system should allow only one of two configurations (best-Fit or first-Served) and display appropriate exception for the user (“It is not allowed to enter any configuration except best-Fit or first-Served”).** |

# System Models

**Use Case Model**

****

## 

## Use Case Tables

* **Park-In**
* **Search for available slots**
* **Park-Out**
* **Payment**

## Park-IN Use Case

|  |  |  |
| --- | --- | --- |
| Use Case ID: | * UC01 | |
| Use Case Name: | * Park-In | |
| Actors: | * Driver | |
| Pre-conditions: | * None | |
| Post-conditions: | * Driver parked successfully | |
| Flow of events: | **User Action** | **System Action** |
| 1. The Driver enters the garage. 2. The Driver requests to Park-In. |  |
|  | 1. The system will ask for Description of the car (model name, Model year and vehicle dimensions). |
| 1. The Driver enters description of the car. |  |
|  | 1. The system activates “search for available slots” function according to the chosen configuration by the owner. 2. The system will mark the arrival time of that vehicle if there is available slot |
| 1. The driver takes the identification number. 2. The Driver goes to the slot and park the car. |  |
| Exceptions: | **User Action** | **System Action** |
| none | |
| Includes: | * Search For Available slots | * Enter Data of Vehicle |
| Notes and Issues: | * None | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | * UC02 | |
| Use Case Name: | * Search for available slots | |
| Actors: | * None | |
| Pre-conditions: | * None | |
| Post-conditions: | * Find available slots | |
| Flow of events: | **User Action** | **System Action** |
|  | 1. The system searches for available slots using best-fit approach according to the configuration. 2. The system will find the slot with the dimensions equal to that specified by the driver if not found the system will find the slot with the closest dimension. 3. The system returns the available slot. |
| Alternative path: |  | 1. The system searches for available slots using first-served approach. 2. The system will find the first slot that fits with the specified dimensions. 3. The system returns the available slot. |
| Exceptions: | **User Action** | **System Action** |
| **E1: No available slots in garage.** | |
|  | 1. The system finds that all slots are occupied by vehicles. 2. The system notifies the user that there are no available slots. 3. The system cancels the process. |
| 1. The Driver leaves the garage. |  |
| **E2: No suitable slots for the vehicle.** | |
|  | 1. The system finds that there is no suitable slot for the dimensions of the vehicles. 2. The system notifies the user that there are no available slots. 3. The system cancels the process. |
| 1. The Driver leaves the garage. |  |
| Includes: | * None | |
| Notes and Issues: | * The system should check there is an available slot or not if there are not available slots the system will display for the driver a message and the process will cancel: **Reliability** | |

## Payment Use Case

|  |  |  |
| --- | --- | --- |
| Use Case ID: | * UC03 | |
| Use Case Name: | * Payment | |
| Actors: | * Driver | |
| Pre-conditions: | * Driver must have sufficient money | |
| Post-conditions: | * Payment process is succeeded | |
| Flow of events: | **User Action** | **System Action** |
| 1-Driver Enters the required money |  |
|  | 2- System checks the entered money  3- System asks for confirmation the process. |
| 4-Driver confirms. |  |
|  | 5- System displays that the process is succeeded.  6- System give the Driver a receipt that contains start time, end time and total fees. |
| 7- Driver will check out |  |
| Exceptions: | **User Action** | **System Action** |
| **E1: Driver enters money less than the total fees** | |
|  | 1-The system will display an error message which is contain notification that the driver did not enter enough money  2-The system would allow for the driver to enter money again |
| **E2 : Driver enters money greater than the total fees** | |
|  | 1 – The system will calculate the rest of the money that the driver should take by subtracting total entered money by total fees  2 – The system will return the rest of the money |
| Includes: | * Parking Fees | |
| Notes and Issues: | * The system should check the money entered by the user is enough for the total number of hours user stays within an hourly rate. If the entered money is less than the total fees the system will ask the driver to enter money again, else if the entered money is greater than the needed money the system will return for the user the rest of the money: **Reliability**. | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | * UC04 | |
| Use Case Name: | * Park-out | |
| Actors: | * Driver | |
| Pre-conditions: | * Already Parked In | |
| Post-conditions: | * Parked Out successfully | |
| Flow of events: | **User Action** | **System Action** |
| 1- Driver has checked out the slot. |  |
| 2-Driver wants to park-out. |  |
|  | 3- System marks departure time (by subtracting entry time from departure time  4- System displays the total hours the driver has spent  5- System calculates the total fees (by parking fees use case) by multiplying the total hours by 5  6- System shows the total fees the driver should pay. |
| 7- Driver should activate “Payment” Function. |  |
|  | 8- System displays that the process is succeeded |
| 9- Driver will leave out |  |
| Exceptions: | * None | |
| Includes: | * Payment process | |
| Notes and Issues: | * Park out should take place no more than 7 seconds: **Performance** | |

## Payment Use Case

## Setup Garage Use Case

|  |  |  |
| --- | --- | --- |
| Use Case ID: | * UC05 | |
| Use Case Name: | * Setup Garage | |
| Actors: | * Administrator | |
| Pre-conditions: | * None | |
| Post-conditions: | * Garage setup successfully | |
| Flow of events: | **User Action** | **System Action** |
| 1. The Administrator requests to setup the garage |  |
|  | 1. The system will ask the Administrator to Set Dimensions of Garage |
| 1. The Administrator Sets Dimensions of Garage |  |
|  | 1. The system will ask the Administrator to Set Configurable Maximum Number of Vehicles. |
| 1. The Administrator Sets Configurable Maximum Number of slots. |  |
|  | 1. The system will ask the Administrator to Set Dimensions of each slot in the garage. |
| 1. The Administrator Sets Dimensions of each slot in the garage. |  |
|  | 1. The system will ask the Administrator to Select Configuration (Best-Fit, First-Served). |
| 1. The Administrator Selects the Configuration. |  |
| Exceptions: | **User Action** | **System Action** |
| E1: Administrator enters negative dimensions | |
|  |  | 1. The system refuses the input dimensions. 2. Inform the Administrator and asks him to enter the dimensions again. |
|  | E2: Administrator enters negative Number of slots | |
|  |  | 1. The system refuses the input Number of slots. 2. Inform the Administrator and asks him to enter the Number of slots again |
|  | E2: Administrator enters wrong configuration | |
|  |  | 1. The system refuses the input Configuration. 2. Inform the Administrator and asks him to choose either best-Fit or first-Served only. |
| Includes: | * Select Configuration * Set Configurable Maximum Number of Vehicles * Set Dimensions of Garage | |
| Notes and Issues: | * The system should check the dimensions, number of slots, cancel negative numbers and display appropriate exception for the user (“Negative Numbers not allowed”). * The system should allow only one of two configurations (best-Fit or first-Served) and display appropriate exception for the user (“It is not allowed to enter any configuration except best-Fit or first-Served”). | |

# Ownership Report

|  |  |
| --- | --- |
| **Item** | **Owners** |
| * Software Requirements Specifications * Class Diagram * Search for Available Slots * Park In * Setup Garage | **Kevin Marco marcory** |
| * Software Requirements Specifications * Class Diagram * Search for Available Slots * Park In * Setup Garage | **Michael Maher Nagi** |
| * Software Requirements Specifications * Class Diagram * Search for Available Slots * Park Out * Payment | **Mariam Tarek Galal** |
| * Software Requirements Specifications * Class Diagram * Search for Available Slots * Park Out * Payment | **Hagar Abd-Elhamed Mohamed Ali** |

# Policy Regarding Plagiarism:

**Students have collective ownership and responsibility of their project. Any violation of academic honesty will have severe consequences and punishment for ALL team members.**

1. تشجع الكلية على مناقشة الأفكار و تبادل المعلومات و مناقشات الطلاب حيث يعتبر هذا جوهريا لعملية تعليمية سليمة
2. ساعد زملاءك على قدر ما تستطيع و حل لهم مشاكلهم فى الكود و لكن تبادل الحلول غير مقبول و يعتبر غشا.
3. أى حل يتشابه مع أى حل آخر بدرجة تقطع بأنهما منقولان من نفس المصدر سيعتبر أن صاحبيهما قد قاما بالغش.
4. قد توجد على النت برامج مشابهة لما نكتبه هنا أى نسخ من على النت يعتبر غشا يحاسب عليه صاحبه.
5. إذا لم تكن متأكدا أن فعلا ما يعد غشا فلتسأل المعيد أو أستاذ المادة.
6. فى حالة ثبوت الغش سيأخذ الطالب سالب درجة المسألة ، و فى حالة تكرار الغش سيرسب الطالب فى المقرر.