Cairo University  
Faculty of Computers and Artificial Intelligent

**CS251 - Software Engineering I**

Garage System

Software Requirements Specifications (SRS)

Team Names

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 4](#_Toc101814805)

[Requirements 4](#_Toc101814806)

[Functional Requirements 4](#_Toc101814807)

[Non Functional Requirements 4](#_Toc101814808)

[System Models 5](#_Toc101814809)

[Use Case Model 6](#_Toc101814810)

[Use Case Tables 7](#_Toc101814811)

[Ownership Report 14](#_Toc101814812)

# Team

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Email** | **Mobile** |
| 20200110 | Basel Mohammed | 11410120200110@stud.cu.edu.eg | 01113677601 |
| 20200050 | Ahmed Mohammed Hany | 11410120200050@stud.cu.edu.eg | 01154951688 |
| 20200315 | Abd El Rahman Tarek | 11410120200315@stud.cu.edu.eg | 01020888206 |
| 20200431 | Mohammed Ayman | 11410120200431@stud.cu.edu.eg | 01150371671 |

# Document Purpose and Audience

- This document explains what the system can do and how to does it.

- garage owner is expected to read this document.

- This document explains how our user will use our system.

- This document is for my Doctor and my TA

# Introduction

## Software Purpose

-making it easy for vehicle owners to reserve slots for their vehicles and making it easy for the owner of the garage to be in control.

## Software Scope

What the system can do:

-choosing an empty slot for the vehicle owner based on configuration chose by

Garage owner.

-take cash money.

What the system cannot do:

- it cannot deal with credit cards.

-it cannot decide the configuration for parking by itself.

-no online services are available.

**Definitions, acronyms, and abbreviations:**

-Best fit: find the slot with the minimum dimension to hold the vehicle.

-First Come: first free slot available from the parking garage slots

# Requirements

## Functional Requirements

- Recording arrival and departure time

- Calculate total income

- Calculate total fees

- Display available parking slot.

- Choose best fit slot.

- System pick free slot.

- System calculates the number of vehicles in it.

## Non-Functional Requirements

- Usability: user able to park without prior registration

- Reliability: Crashing system does not make data leakage for information's of cars

- Supportability: The system supports cars in any size of trucks, cars, motorcycles

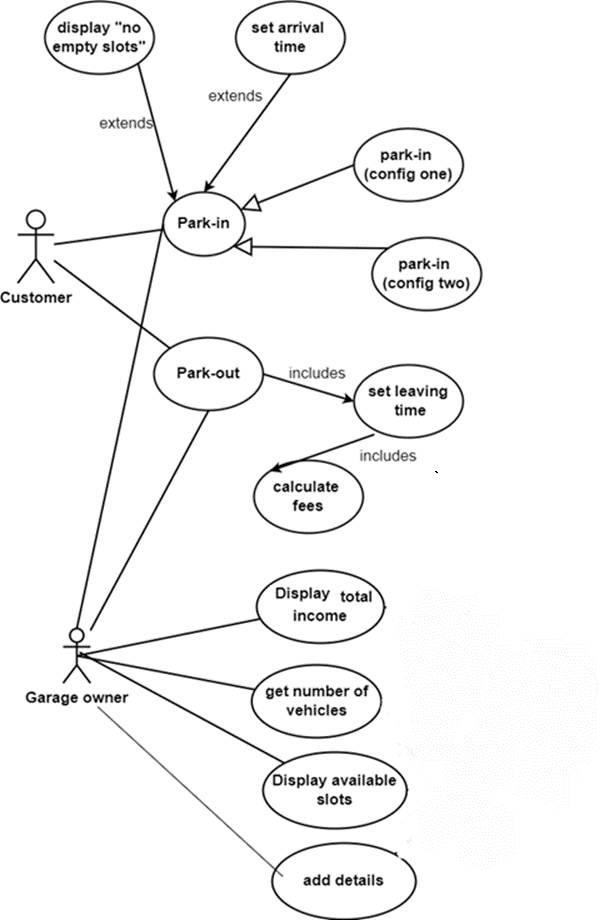
- Availability: The system is available daily from 7 am to 1 am.

- performance: choosing an empty slot takes less than 3 seconds.

- Scalability: the system handles one user at a time

# System Models

## Use Case Model



## Use Case Tables

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 1 | |
| Use Case Name: | Park-in (Config one) – First Come | |
| Actors: | Vehicle Owner | |
| Pre-conditions: | none | |
| Post-conditions: | A slot has been reserved | |
| Flow of events: | **User Action** | **System Action** |
| 1- User Enters model name, Model year and vehicle dimensions. |  |
|  | 2- system the first free slot available.  Config 2: find the slot with the minimum dimension to hold the vehicle |
|  | 3- if there is a slot the system automatically stores the arrival time and tells the user his unique ID number. |
| Exceptions: | **User Action** | **System Action** |
| 1- User Enters model name, Model year and vehicle dimensions. |  |
|  |  | 2- system cannot find any suitable slot. |
|  |  | 3- system rejects order. |
| Includes: | None. | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 2 | |
| Use Case Name: | Park-in (Config two) – Best Fit | |
| Actors: | Vehicle Owner | |
| Pre-conditions: | none | |
| Post-conditions: | A slot has been reserved | |
| Flow of events: | **User Action** | **System Action** |
| 1- User Enters model name, Model year and vehicle dimensions. |  |
|  | 2- system finds the slot with the minimum dimension to hold the vehicle |
|  | 3- the system automatically stores the arrival time and tells the user his unique ID number. |
| Exceptions: | **User Action** | **System Action** |
| 1- User Enters model name, Model year and vehicle dimensions. |  |
|  |  | 2- system cannot find any suitable slot. |
|  |  | 3- system rejects order. |
| Includes: | None. | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 3 | |
| Use Case Name: | Park-out | |
| Actors: | vehicle Owner | |
| Pre-conditions: | none | |
| Post-conditions: | Vehicle owner takes Vehicle and pays money. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User Enters vehicle ID. |  |
|  | 2- system stores current time and calculates fees. (an hour costs 5 EGP) |
| 3- user pays money and leave. |  |
|  |  | 4- system updates total income. |
| Exceptions: | **User Action** | **System Action** |
| 1- User Enters vehicle ID. |  |
|  | 2- INVALID ID.  3- System rejects order. |
| Includes: | Set leaving time | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 4 | |
| Use Case Name: | Display available slots | |
| Actors: | Garage Owner | |
| Pre-conditions: | None | |
| Post-conditions: | Empty slots are displayed | |
| Flow of events: | **User Action** | **System Action** |
| 1- garage owner selects the "Display available slots " option. |  |
|  | 2- system displays all empty slots IDs and dimensions. |
| **User Action** | **System Action** |
| Includes: |  |  |
| Notes and Issues: |  |  |
|  |  |
|  |  |
|  |  |

# 

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 5 | |
| Use Case Name: | Display total income | |
| Actors: | Garage Owner | |
| Pre-conditions: | None | |
| Post-conditions: | Total income is displayed | |
| Flow of events: | **User Action** | **System Action** |
| 1- garage owner selects the "Calculate total income " option. |  |
|  | 2- the system retrieves total income and prints it. |
|  |  |  |
| Includes: | none | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 6 | |
| Use Case Name: | Get number of vehicles | |
| Actors: | Garage Owner | |
| Pre-conditions: | None | |
| Post-conditions: | Number of vehicles is displayed | |
| Flow of events: | **User Action** | **System Action** |
| 1- garage owner selects the "Total number of vehicles " option. |  |
|  | 2-system retrieves total number of vehicles at that time and displays it. |
| Exceptions: | **User Action** | **System Action** |
| Includes: |  | |
| Notes and Issues: | Note that anyone is considered usual vehicle owner until he logs in. | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 7 | |
| Use Case Name: | Add Details | |
| Actors: | Garage Owner | |
| Pre-conditions: | none | |
| Post-conditions: | New slots are added. | |
| Flow of events: | **User Action** | **System Action** |
| 1- garage owner starts the program. |  |
|  | 2-system asks for Details (his name, number of slots and dimensions of each one). |
| 3- owner submits Details. |  |
|  | 4-System accepts it. |
| Includes: |  | |
| Notes and Issues: |  | |

# Ownership Report

|  |  |
| --- | --- |
| **Item** | **Owners** |
| Requirements, Introduction, software purpose, scope. | *Abd El-Rahman Tarek* |
| Use case models and tables | *Ahmed Mohammed Hany* |