Cairo University  
Faculty of Computers and Artificial Intelligent

**CS251 - Software Engineering I**

Parking Garage

Software Requirements Specifications (SRS)

Team Names

Mohamed tarek fathi

Ahmed adel emad

Mohamed Youssef

Fady osama

Month & Year

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** |
| 20200794 | Mohamed tarek fathi |  |  |
| 20200025 | Ahmed adel emad |  |  |
| 20200487 | Mohamed Youssef sultan |  |  |
| 20200369 | Fady osama nasif |  |  |

# Document Purpose and Audience

purpose create system for owner of garage ,by the system owner can easy reserve a slot for cars;

Audience : owners of garages

document : simpley it explain what are steps of building projects like usecse its represent who will use the system and functions will be build in the project

# Introduction

## Software Purpose

purpose of software make easy way to make owners of garage to park cars belong to customers

## Software Scope

scope: system can park cars and calculate time of Entry , time of exit car ,calculate the cost of parking

## Definitions, acronyms, and abbreviations

# Requirements

## Functional Requirements

1-identify each vehicle by model name and model year

2-identify vehicle dimensions (width - depth)

3-make identification number for each vehicle

4-capture arrival time of vehicle automatically

5-check slots if there is available one (park in)

6-provide two configurations which garage owner will choose from:

i- first come first served slots

ii- best fit

7-capture departure time of vehicle from garage automatically

8-calculate total time of stay (park out)

9-calculate parking fees based on total time of stay

10-calculate total income of garage at any given point in time

11-calculate total number of cars used the garage at any given point in time

12-display and determine the available slots in the garage

13-display descriptive error message for each exception can happen

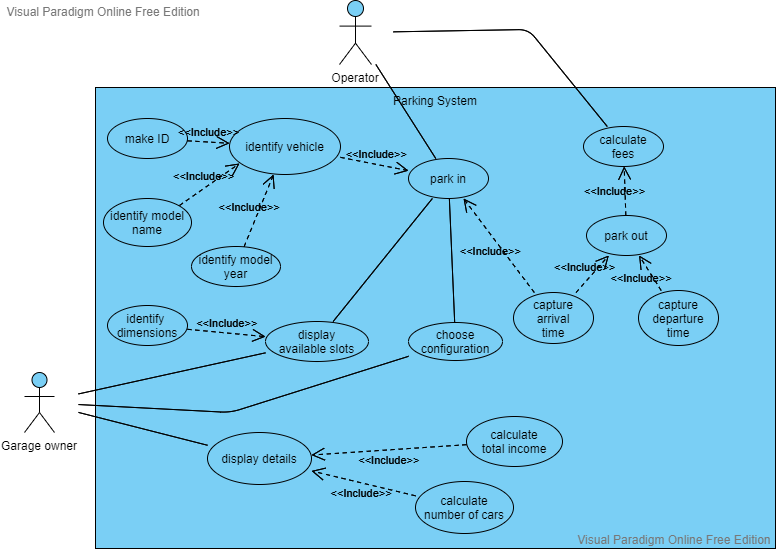
## Non Functional Requirements

|  |  |
| --- | --- |
|  | **Details** |
| **Usability** | **system easy to take dimensions of vehicle and display available slots clearly** |
| **Reliability** | **system able to recognize if given dimensions are wrong** |
| **Availability** | **the system must be available in all working hours of the garage and drop down less than 5 sec at once** |
| **Maintainability** | **we can add new models of cars and change number of slots and it’s dimensions without affect anything in the system** |
| **Adaptability** | **the system able to change the cost per hour or change the way to calculate fees** |
| **Performance** | **system must able to check time of stay, calculate fees, check available slots and display anyone of them each in 10 sec maximum** |

# 

# System Models

## Use Case Model

****

## 

## Use Case Tables

|  |  |  |
| --- | --- | --- |
| Use Case ID: | UC-1 | |
| Use Case Name: | identify vehicle | |
| Actors: | Operator | |
| Pre-conditions: | Park in | |
| Post-conditions: | Identification successful | |
| Flow of events: | **User Action** | **System Action** |
| 1-Driver park in |  |
|  | 2-System ask for identification info |
| 3-Operator identify model name |  |
| 4-Operator identify model year |  |
| 5-Operator make ID for the vehicle |  |
|  |  | 6-System acknowledge Identification successful |
| Exceptions: | **User Action** | **System Action** |
| 1-Driver park in |  |
|  | 2-No available slots  3- System rejects the vehicle |
| Includes: | UC-2: make ID UC-3: Identify model name UC-4: identify model year | |
| Notes and Issues: | Availability 🡪 the system must be available in all working hours of the garage and drop down less than 5 sec at once  Maintainability 🡪 we can add new models of cars and change number of slots and it’s dimensions without affect anything in the system | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | UC-5 | |
| Use Case Name: | Display available slots | |
| Actors: | Garage owner, Operator | |
| Pre-conditions: | Park in | |
| Post-conditions: | Display available slots | |
| Flow of events: | **User Action** | **System Action** |
| 1-Driver park in |  |
|  | 2-System request identify dimensions |
| 3-identify dimensions |  |
|  | 4-Check available slot |
|  | 5- Display available slots |
| Exceptions: | **User Action** | **System Action** |
| 1-Driver park in |  |
|  | 2-No available slots  3- System rejects the vehicle |
| Includes: | UC-6: identify dimensions | |
| Notes and Issues: | Usability 🡪 system easy to take dimensions of vehicle and display available slots clearly  Reliability 🡪 system able to recognize if given dimensions are wrong | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | UC-7 | |
| Use Case Name: | Choose configuration | |
| Actors: | Garage owner | |
| Pre-conditions: | Garage owner set configuration | |
| Post-conditions: | Configuration added successfully | |
| Flow of events: | **User Action** | **System Action** |
| 1-Set configuration |  |
|  | 2-System provide two configurations |
| 3- Select one configuration |  |
|  | 4-System applied configuration to the parking way |
|  | 5- Configuration added successfully |
| Exceptions: | **User Action** | **System Action** |
| 1-Set configuration |  |
|  | 2-system down 3- configuration not applied |
| Includes: | None | |
| Notes and Issues: | Maintainability 🡪 we can add new models of cars and change number of slots and it’s dimensions without affect anything in the system  Adaptability 🡪 the system able to change the cost per hour or change the way to calculate fees | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | UC-8 | |
| Use Case Name: | Park in | |
| Actors: | Operator | |
| Pre-conditions: | Park in | |
| Post-conditions: | parking successfully | |
| Flow of events: | **User Action** | **System Action** |
| 1-Driver park in |  |
|  | 2-System ask for identification info |
| 3-Operator identify model name 4-Operator identify model year 5-Operator make ID for the vehicle |  |
|  |  | 6-System acknowledge Identification successful |
|  | 7-identify dimensions |  |
|  |  | 8-Check available slot 9- Display available slots |
|  | 10-Park in | 11-capture arrival time |
| Exceptions: | **User Action** | **System Action** |
| 1-Driver park in |  |
|  | 2-No available slots  3- System rejects the vehicle |
| Includes: | UC-1: identify vehicle UC-11: capture arrival time | |
| Notes and Issues: | Usability 🡪 system easy to take dimensions of vehicle and display available slots clearly  Reliability 🡪 system able to recognize if given dimensions are wrong Availability 🡪 the system must be available in all working hours of the garage and drop down less than 5 sec at once | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | UC-9 | |
| Use Case Name: | Calculate fees | |
| Actors: | Operator | |
| Pre-conditions: | Park out | |
| Post-conditions: | Driver pay fees | |
| Flow of events: | **User Action** | **System Action** |
| 1-Driver park out |  |
|  | 2-capture departure time |
|  | 3-Calculate time of stay |
|  | 4-Calculate fees |
|  | 5- Driver pay fees |  |
| Exceptions: | **User Action** | **System Action** |
| 1-Driver park out |  |
|  | 2-capture departure time  3-Calculate time of stay  4-Calculate fees |
|  | 5-driver has no cash to pay |  |
| Includes: | UC-10: park out UC-11: capture arrival time UC-12: capture departure time | |
| Notes and Issues: | Performance 🡪 system must able to check time of stay, calculate fees, check available slots and display anyone of them each in 10 sec maximum Availability 🡪 the system must be available in all working hours of the garage and drop down less than 5 sec at once | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | UC-13 | |
| Use Case Name: | display details | |
| Actors: | Garage owner | |
| Pre-conditions: | Garage owner request details | |
| Post-conditions: | display details | |
| Flow of events: | **User Action** | **System Action** |
| 1-requset details |  |
|  | 2-calculate total income |
|  | 3- calculate number of cars |
|  |  | 4-display details |
| Exceptions: | **User Action** | **System Action** |
| 1-requset details |  |
|  | 2-No available details  3- don’t display details |
| Includes: | UC-14: calculate total income UC-15: calculate number of cars | |
| Notes and Issues: | Maintainability 🡪 we can add new models of cars and change number of slots and it’s dimensions without affect anything in the system  Adaptability 🡪 the system able to change the cost per hour or change the way to calculate fees Performance 🡪 system must able to check time of stay, calculate fees, check available slots and display anyone of them each in 10 sec maximum | |

# Ownership Report

|  |  |
| --- | --- |
| **Item** | **Owners** |
| requirments | *Mohamed tarek* |
| System Models | *Mohamed tarek* |