Cairo University  
Faculty of Computers and Artificial Intelligent

**CS251**

**Software Engineering I**

Project Name

Software Design

Hour Ahmed Mohamed Fathi

Rana Mohamed Ali

Nada Mostafa Mohammed Mahmoud Hefnawy

Hossam Ahmed Mahmoud Ali

8/5/2022

Contents

[Instructions [To be removed] 3](#_Toc101814919)

[Team 3](#_Toc101814920)

[Document Purpose and Audience 3](#_Toc101814921)

[System Models 3](#_Toc101814922)

[I. Class diagrams 3](#_Toc101814923)

[Important Algorithm 4](#_Toc101814924)

[II. Sequence diagrams 5](#_Toc101814925)

[Class - Sequence Usage Table 6](#_Toc101814926)

[Ownership Report 6](#_Toc101814927)

[Policy Regarding Plagiarism: 7](#_Toc101814928)

# Team

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Email** | **Mobile** |
| 20201058 | Hour Ahmed Mohamed Fathi | [Hourahmed11@gmail.com](mailto:Hourahmed11@gmail.com) | 01018525409 |
| 20200182 | Rana Mohamed Ali | Ranamkk55@gmail.com | 01148459501 |
| 20200594 | Nada Mostafa Mohammed Mahmoud Hefnawy | Nadaelhefnawy2002@gmail.com | 01141971413 |
| 20200148 | Hossam Ahmed Mahmoud Ali | Hossamsultan11@gmail.com | 01141971413 |

# Document Purpose and Audience

**Purpose**

The purpose of the Software Design Document is to provide a description of the design of a system fully enough to allow for software development to proceed with an understanding of what is to be built and how it is expected to built.

The Software Design Document provides information necessary to provide description of the details for the software and system to be built.

**Audience**

This document is intended to inform the project's software team of the design.

So the audience of this document is the most important is the software team of this project and its manager and those working on the software system

**the Audience can be the**

**1-the team who made that system**

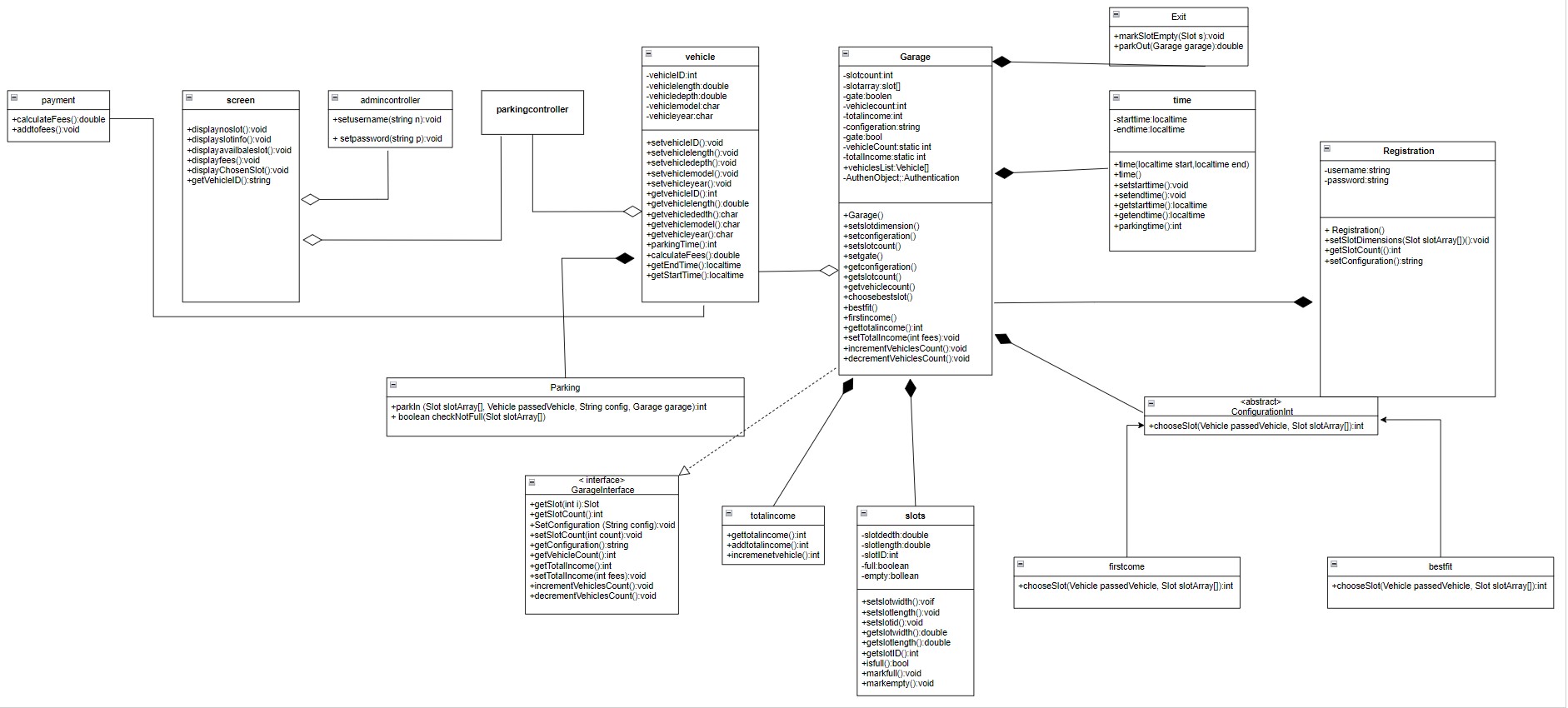
**2-the owner**

**3-the project manager**

**4- the vehicle driver**

# System Models

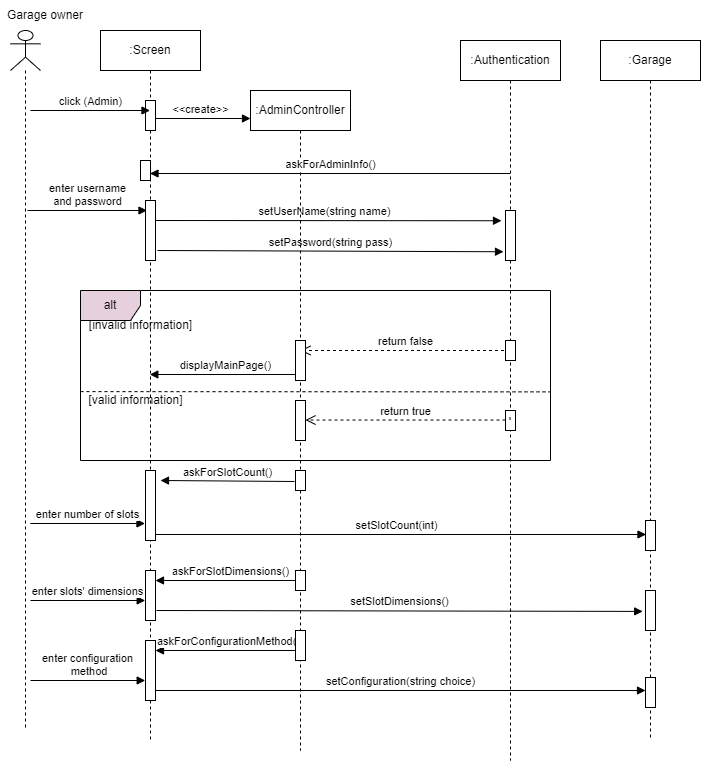
## Class diagrams



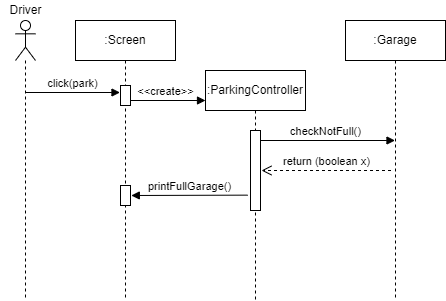
| **Class ID** | **Class Name** | **Description & Responsibility** |
| --- | --- | --- |
| 1 | Parking system | it is the super main class with which every other class are associated |
| 2 | payment | It is a cash payment method class that is associated with Parking system class |
| 3 | screen | It is the main Boundary class from which Actors can trigger the system.  its main responsibilities are  - presenting actor's options and choices  - taking inputs from actors.  - displaying System messages and warnings. |
| 4 | Admin controller | it is a controller class that manages the system reactions with the Admin (garage owner) |
| 5 | vehicle | it is an entity class that represents the vehicle. its main responsibilities are:  - storing main data about the vehicle such as length, width, customer name, both arrival and departure time.  - automatically generating a unique id for each vehicle.  - getting the parking time and calculating the parking fees for all parking time. |
| 6 | garage | it's an entity class that represents the garage that aimes to set up the garage its main responsibilities are:  - choosing the configuration method for finding a free slot either by best-fit approach or first come first served approach.  - counting the number of slots through a static counter.  - setting slot dimensions, and finding the best slot among available ones.  - Check if the garage is full.  - the main two methods are:  - park-in  -park-out  and calculating total income of the garage. |
| 7 | Authentication | it's more like a verifying class that checks if the administrator is the only one accessing garage private info and setting up garage . the authentication step is used almost before each step in garage setting and editing.  its main responsibilities are:  - setting the owner username and password for the first time.  - preventing multiple owners.  - verifying if the sent info belongs to the same owner(administrator) |
| 8 | Slots | it is an entity class that aims to store more detailed data about parking slots its main responsibilities are:  - saving all slots info  - get all the available slots |

## II. Sequence diagrams

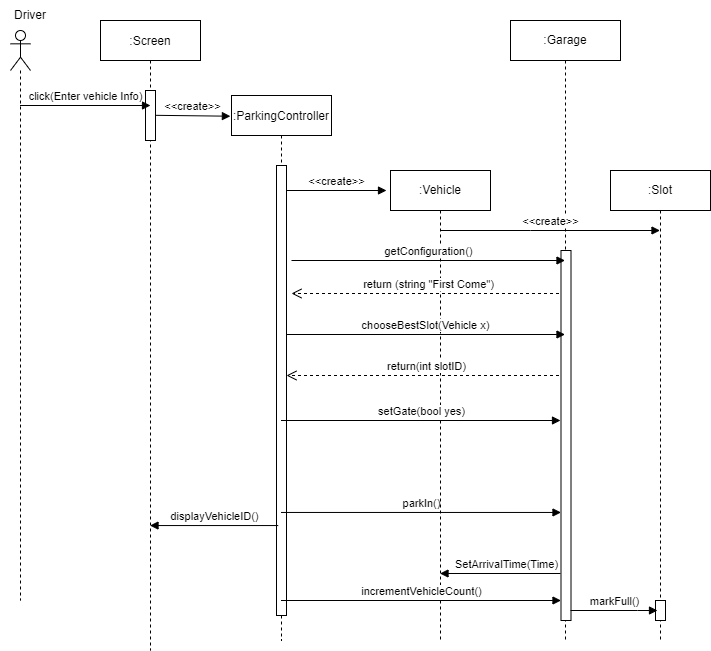
1-

****

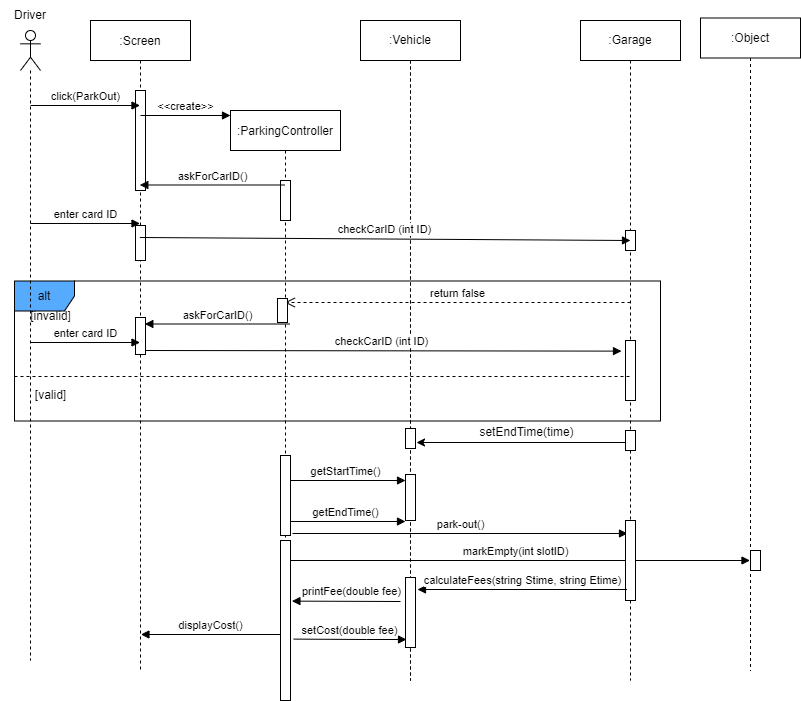
**2-**

****

**3-**

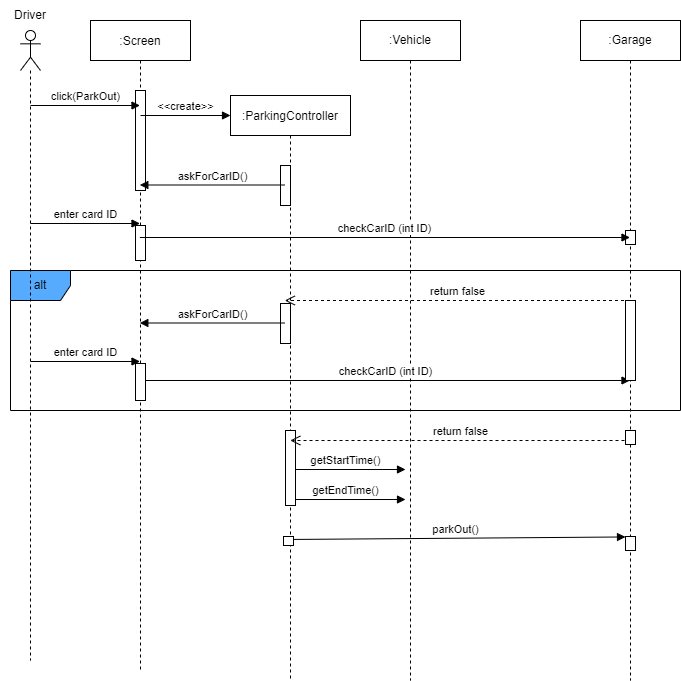
****

**4-**

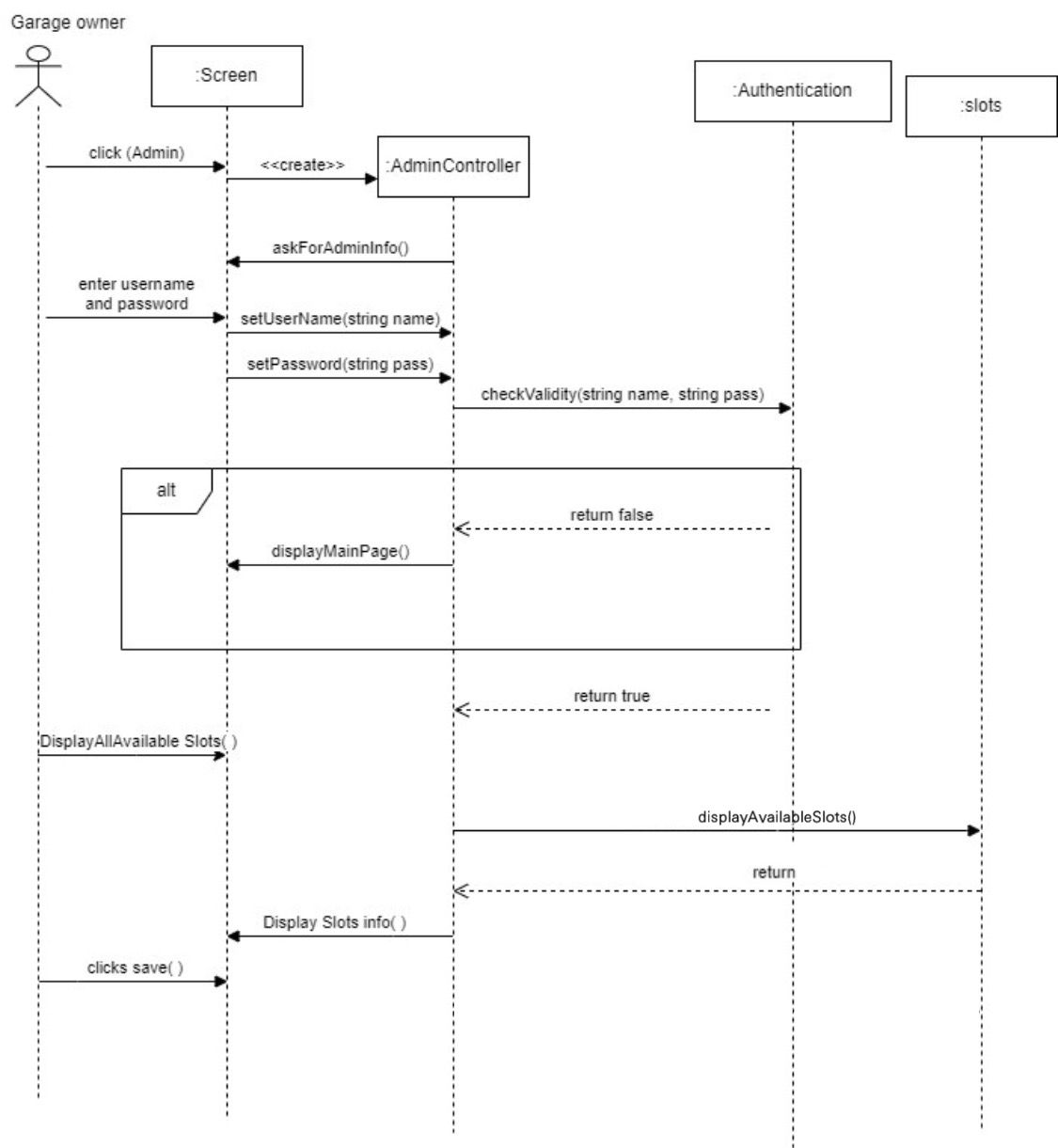
****

:Slots

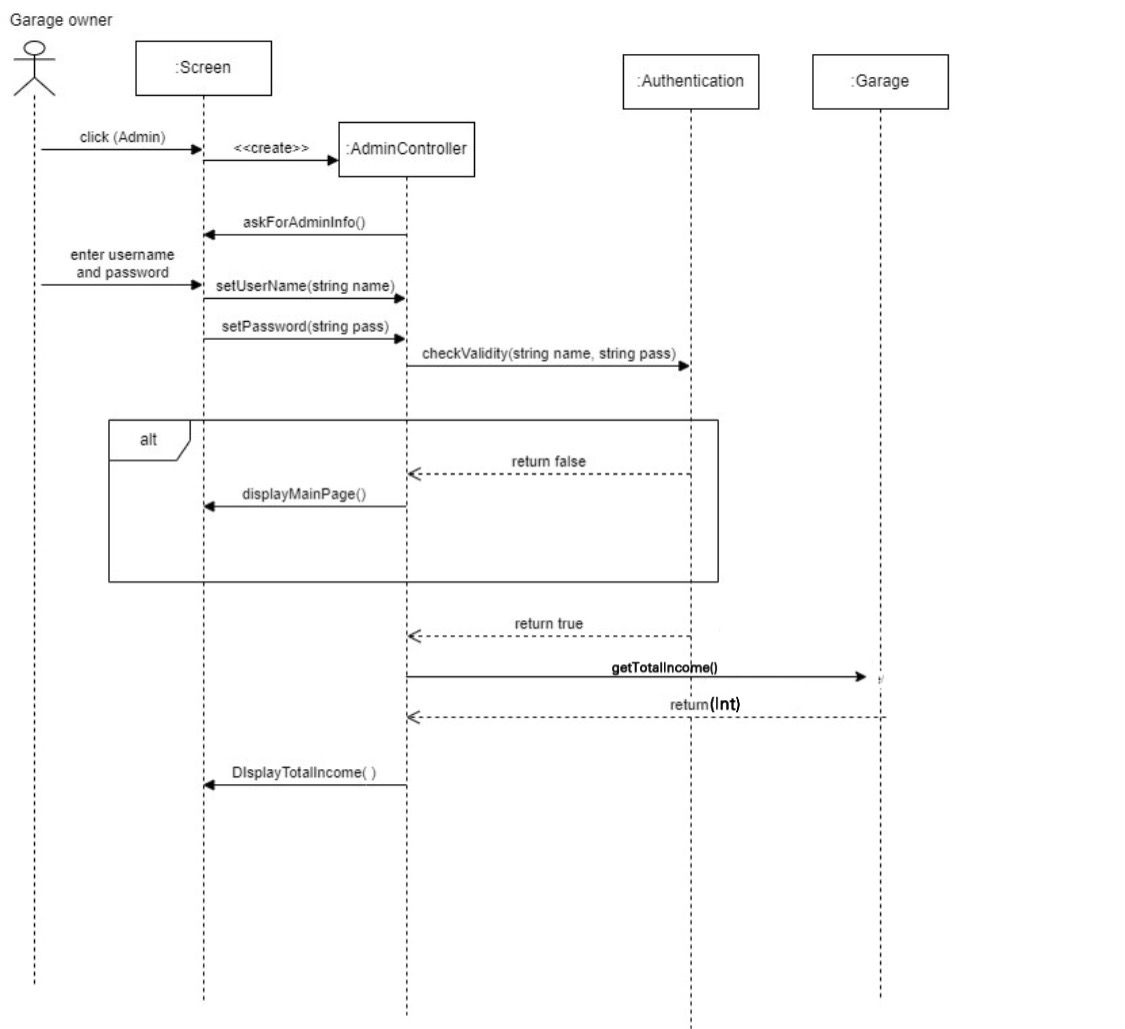
**5-**

****

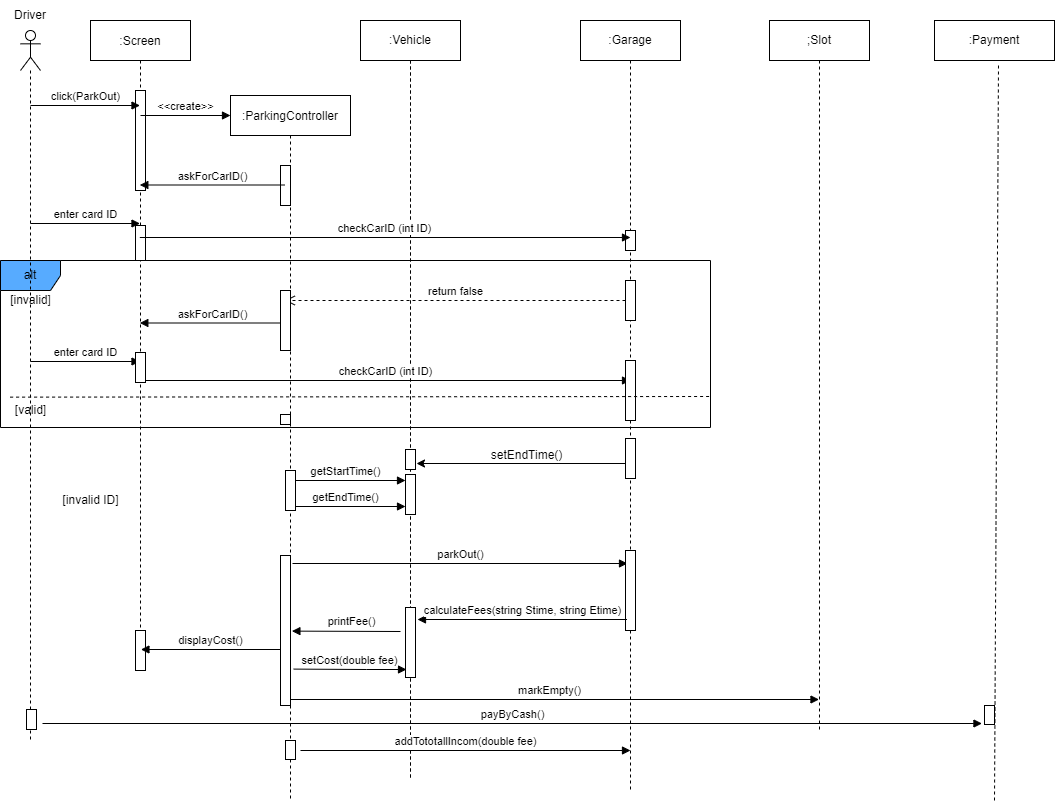
**6-**

****

**7-**

****

**8-**

****

### Class - Sequence Usage Table

| **Class Name** | **Sequence Diagrams** | **Overall used methods** |
| --- | --- | --- |
| Garage | 1,2,3,4,5,6,8 | setSlotCount(int),  setSlotDimensions(),  setConfiguration(string choice),  checkNotFull(),getConfiguration(),  chooseBestSlot(Vehicle x),  setGate(bool yes), parkIn(),  incrementVehicleCount(),  checkCarID (int ID), park-out(),  getTotalIncome() |
| Screen | 1,2,3,4,5,6,7,8 | askForAdminInfo(),  displayMainPage(),  askForSlotCount(),  askForSlotDimensions(),  askForConfigurationMethod(),  printFullGarage(),displayVehicleID(),  askForCarID(),displayCost(),  displayTotalIncome(),  displaySlotsInfo() |
| Vehicle | 3,4,5,8 | SetArrivalTime(Time),  setEndTime(time),  getStartTime(),getEndTime(),  setCost(double fee) |
| AdminController | 1,6,7 | setUsername(string name),  setPassword(string pass) |
| ParkingController | 2,3,4,5,8 | none |
| Payment | 8 | payByCash() |
| Authentication | 1,6,7 | setUserName(string name),  setPassword(string pass),  checkValidity(string name, string pass) |
| Slots | 4,8 | markEmpty(),markFull() |

# Ownership Report

|  |  |
| --- | --- |
| **Item** | **Owners** |
| ParkIn Function | *Hour Ahmed* |
| DisplayAvailableSlot Function | *Nada Mostafa* |
| Use Case Diagram | *Hour Ahmed* |
| Class Diagram | *Nada Mostafa,hossam ahmed* |
| Project details (scope, audience, etc) | *Hossam ahmed* |
| Functional Requirements | *Rana Mohamed, Nada Mostafa* |
| Non-Functional requirements | *Hour Ahmed* |
| Sequence Diagrams | *Hour Ahmed, Rana Mohamed, Nada Mostafa* |
| Use case description / table | *Hossam ahmed, Nada Mustafa* |
| Code | *Hour Ahmed, Hossam ahmed, Nada Mostafa, Rana Mohamed* |

# 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.