Cairo University  
Faculty of Computers and Artificial Intelligence



**CS251**

**Software Engineering I**

GoFo

Software Design Specifications

Version 2

Abd-Elrahman Mohamed Ahmed Morsy

abdomorsy389@gmail.com

June & 2021

Contents

[Instructions [To be removed] 3](#_Toc72827671)

[Team 3](#_Toc72827672)

[Document Purpose and Audience 3](#_Toc72827673)

[System Models 3](#_Toc72827674)

[I. Class Diagram(s) 3](#_Toc72827675)

[II. Class Descriptions 5](#_Toc72827676)

[III. Sequence diagrams 5](#_Toc72827677)

[Class - Sequence Usage Table 6](#_Toc72827678)

[IV. State Diagram 7](#_Toc72827679)

[Tools 7](#_Toc72827680)

[Ownership Report 7](#_Toc72827681)

# Team

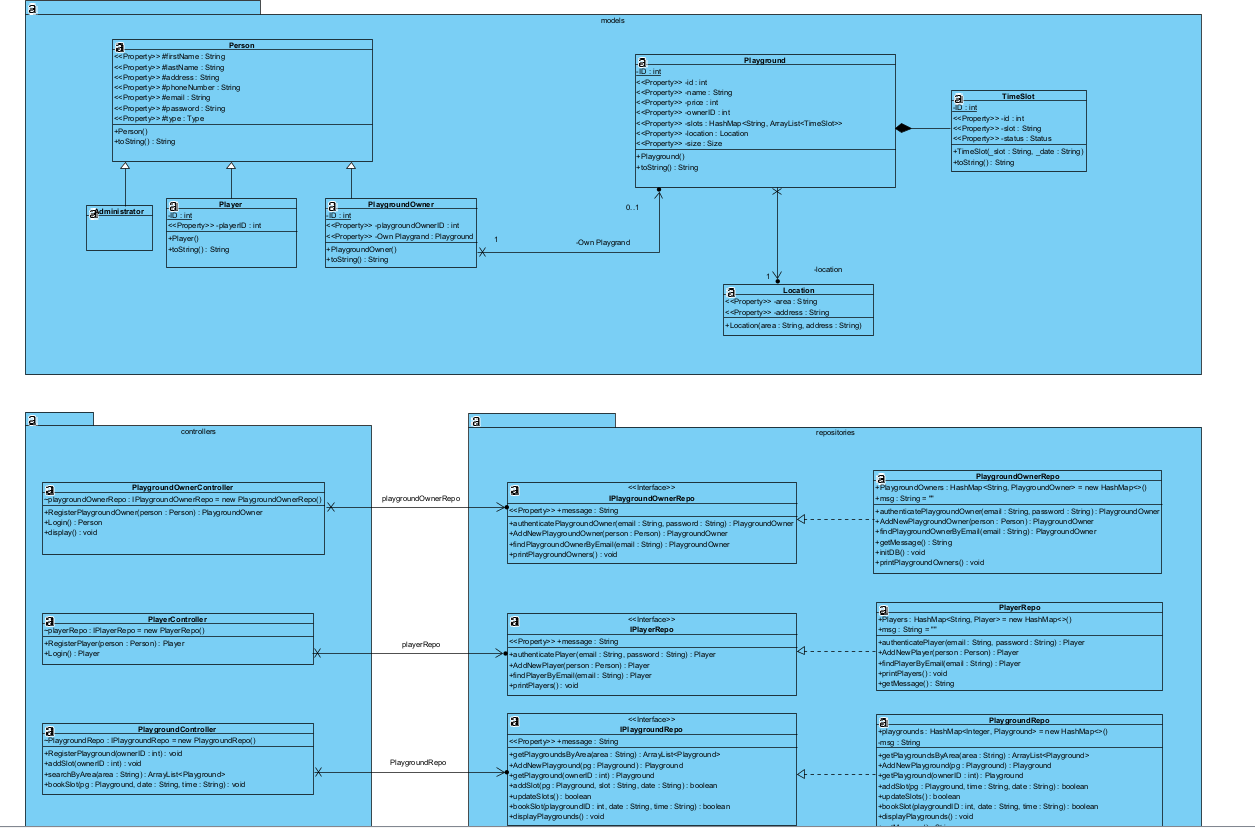
|  |  |  |  |
| --- | --- | --- | --- |
| ID | Name | Email | Mobile |
| 20170149 | Abdelrahman Mohamed Ahmed | abdomorsy389@gmail.com | 01140299729 |

# Document Purpose and Audience

* The purpose of this document is to outline the design views of the **gofo** project which will satisfy functional requirements stated in the SRS Document.
* This document is serving as a guideline for the **developers** throughout development phase of the project.

# System Models

## I. Class Diagram(s)



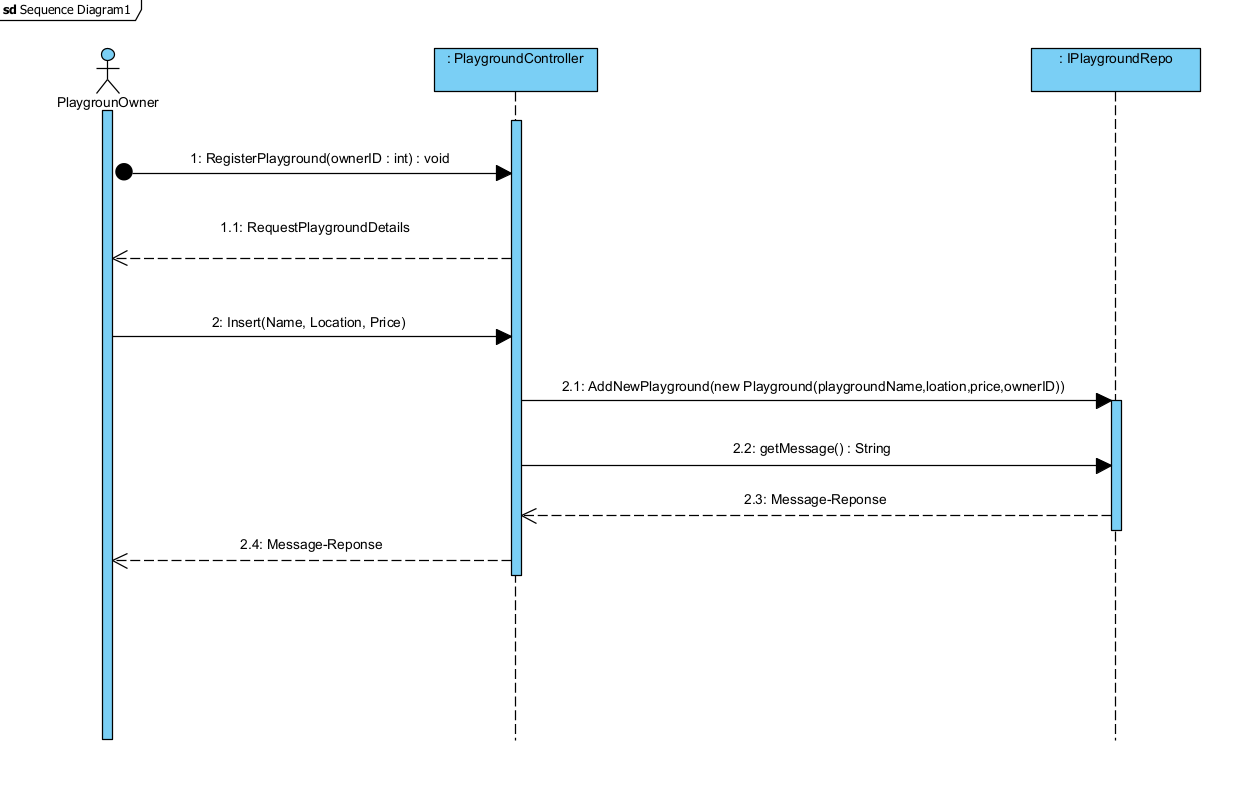
## II. Class Descriptions

|  |  |  |
| --- | --- | --- |
| **Class ID** | **Class Name** | **Description & Responsibility** |
| 1. | **Person** | This class for represent the person attributes in our app. |
| 2. | **PlaygroundOwner** | This class for represent the Playground-Owner (person) attributes in our app. |
| 3 | **Player** | This class for represent the Player (person) attributes in our app. |
| 4 | **Playground** | This class for represent the Playground attributes in our app. |
| 5 | **PlaygroundController** | This class responsible for treating with any request or anything will affect a playground object. |
| 6 | **PlaygroundOwnerController** | This class responsible for treating with any request or anything will affect a playground-owner object. |
| 7 | **PlayerController** | This class responsible for treating with any request or anything will affect a player object. |
| 8 | **PlayerRepository** | This class responsible for treating with CRUD operations or anything treats with the Players in the database. |
| 9 | **PlaygroundRepository** | This class responsible for treating with CRUD operations or anything treats with the Playgrounds in the database. |
| 10 | **PlaygroundOwnerRepository** | This class responsible for treating with CRUD operations or anything treats with the Playground-Owners in the database. |

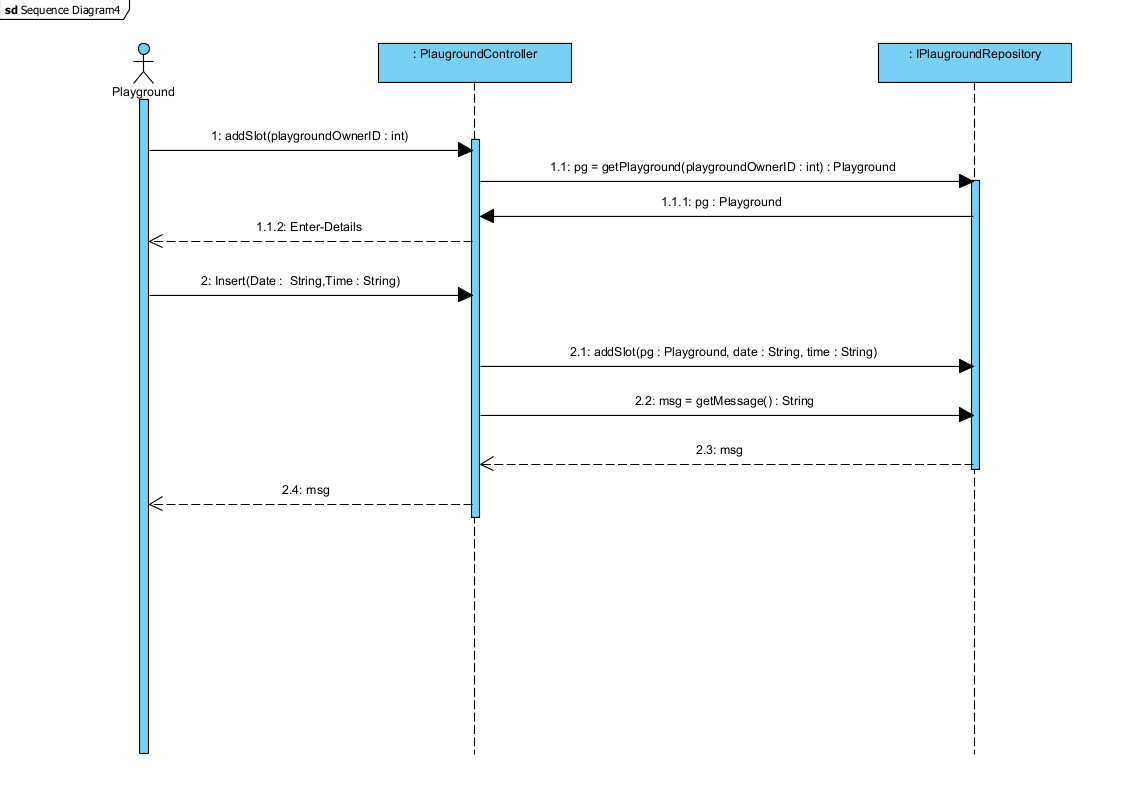
## 

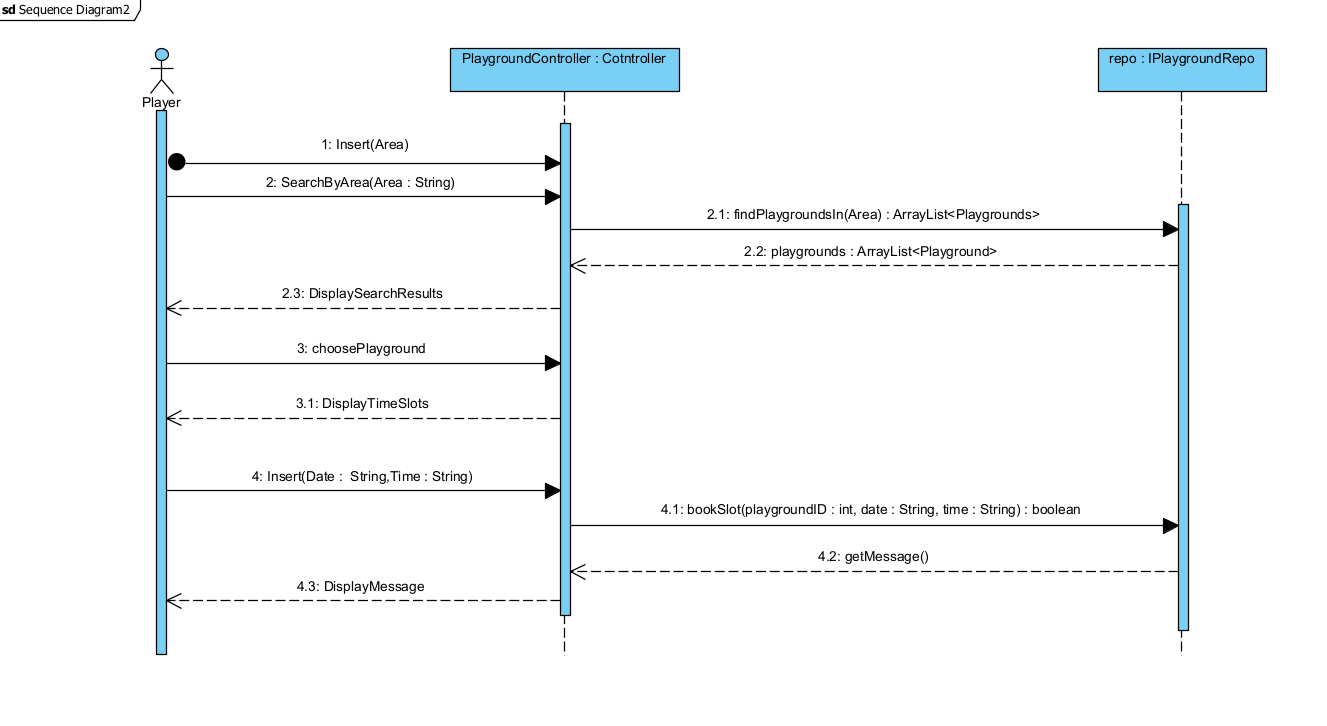
## III. Sequence diagrams

1. **Register new Playground**

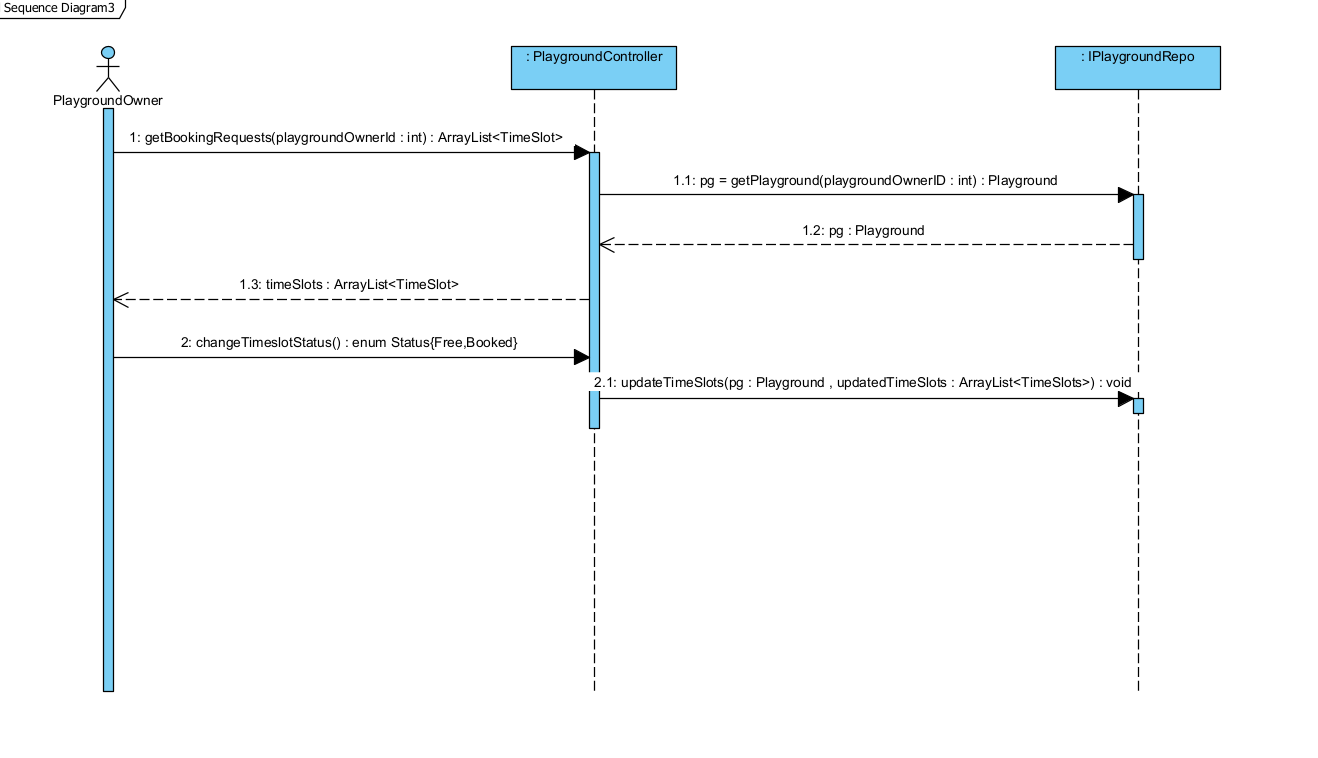
****

1. **Add slot**

****

1. **Book Slot  
     
   **

1. **Accept/Reject Booking**



### 

### Class - Sequence Usage Table

|  |  |  |
| --- | --- | --- |
| **Sequence Diagram** | **Classes Used** | **All Methods Used** |
| 1. Register New Playground | Class PlaygroundOwner  Class Playground  Class PlaygroundController  Class IPlaygroundRepository | RegisterPlayground(playgroundOwnerId : int) : void  AddNewPlayground(pg : playground) |
| 1. Adding Time-Slot | Class PlaygroundOwner  Class Playground  Class PlaygroundController  Class IPlaygroundRepository | getPlayground(playgroundOwnerId : int) : Playground  addSlot(pg : Playground, date : String, time : String)  getMessage() : String |
| 1. Book Time-Slot | Class Player  Class Playground  Class PlaygroundController  Class IPlaygroundRepository | searchByArea(area : String) : ArrayList<Playground>  getPlaygroundsByArea(area : String) : ArrayList<Playground>  bookSlot(pgID : int, date : String, time : String) : boolean  getMessage() : String |
| 1. Accept/Reject request | Class PlaygroundOwner  Class Playground  Class PlaygroundController  Class IPlaygroundRepository | getBookingRequests(playgroundOwnerId : int) : ArrayList<TimeSlot>  getPlayground(playgroundOwnerID : int) : Playground  updateTimeSlots ( pg : Playground , updatedTimeSlots : ArrayList<TimeSlots>) : void |

## IV. State Diagram

# Tools

* **Visual-Paradigm**

# Ownership Report