

**Cairo University**

**Faculty of Computers and Artificial Intelligence**



# **CS251**

## **Intro. to Software Engineering**

### **Gofu project**

### **Software Requirements Specifications**

**Version 1.0**

**Team Names and Emails and Phone of**

**Team Leader**

**Month & Year**

**Content**



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

Team.....	2
Document Purpose and Audience.....	3
Introduction.....	3
Software Purpose .....	3
Software Scope.....	3
Definitions, acronyms, and abbreviations.....	4
Requirements .....	4
Functional Requirements .....	4
Non Functional Requirements .....	6
System Models .....	6
Use Case Model.....	6
Enriched User Stories .....	8
System Navigation Map .....	51
Tools .....	53
Ownership Report .....	53

### Team

ID	Name	Email	Mobile
----	------	-------	--------



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

20190141	Peter Essam Maguid Isaac	<a href="mailto:Pessam33@gmail.com">Pessam33@gmail.com</a>	01288333014
20190304	Abdallah Ragab Abdallah	abdallah.ragab.1432001@gmail.com	01121095927
20190818	Shehab eldin Khaled mohamed	<a href="mailto:Shehabgad2@gmail.com">Shehabgad2@gmail.com</a>	01066899613

### Document Purpose and Audience

### Introduction

#### Software Purpose

This system is a football playground reservation system. It benefits both the teams/players and the playground owners where:

- A team can book the playground they find it good for them and fits them in terms of their budget, free time slots and nearby playground locations.
- A playground owner can rent his playground with a suitable price for him, whenever he wants and for a suitable team as he can accept and decline team requests.

#### Software Scope

- The system makes the player access to a large number of playground until he chooses what he wants.
- It makes players to find teams and join it.
- There is the administrator to prevent any fraud from the players or the playground owner.
- There is an e-wallet that enables the players to pay money to the playground owner.



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

### Definitions, acronyms, and abbreviations

- **Administrator** : someone who oversees the overall operations of the system and ensures that no fraud takes place, he can also add or delete or suspend any playground owner if the owner does some fraudulent activities like double booking or if the playground gets a lot of complaints from the players.
- **E-wallet**: it is a type of electronic card, player pays the money from his e-Wallet to the e-Wallet of the owner to the booking will be accepted
- **Player**: someone who books a playground to plays football with his friends.

## Requirements

### Functional Requirements

- Administrator can add or remove a playground owner by one click on buttons (add) or (remove). - If he clicks add , a menu appears to enter all information of a playground owner including: his name, phone number, his e-Wallet registration, address of the playground , and price per hour. •  
Administrator can view all bookings between playground owners and players by clicking on (bookings) button.
  - when he clicks (bookings) button, a menu appears with all bookings' information including: which playground and which player is the booking between, time slots of bookings, money paid.
  - Administrator can see if the owner does some fraudulent activities like double booking and he can delete completely or activate the account of the owner again by tapping on the owner's account and then tap on either (delete permanently) or (deactivate) or (activate)
  - If the administrator taps on delete permanently, the owner's account will be deleted and can't be activated again.
  - If the administrator taps on deactivate, the owner's account will be deactivated and the (deactivate) button will turn into (activate) button so the administrator can activate the owner's account again.
- Administrator can view e-wallets of both playground owners and players inside each booking to ensure that no fraud takes place.



# CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

- Administrator can view all the reviews and complaints of each playground so he can delete or suspend/deactivate any bad playground.
- Player can register in the system and creates a profile it include his name, his email, his address and his phone number.
- Player can display the playgrounds near to him or display it in a specific location on specific dates.
- Player can filter the playground by the hours and date he selects.
- Player can book a time slot of 1 or more hours if it is available.
- When a player wants to book he will check from available playgrounds and time slots, selects the free time slot he want and calculates the total price.
- When booking is accepted , the money will be paid from his e-Wallet to the e-Wallet of the owner.
- The system updates the status of the booked slot

If it is accepted or denied he can remove it from the tab so it is not shown again.

If it is pending he can cancel the request.

- Player can create his favorite team and store their names and emails.
- Player can also cancel a booking if within the cancellation period.
- Player can enter to the playground's details and see it like the playground is natural grass or artificial grass .
- If the team is not complete the player can join another team so it can be complete and the previous request is canceled and the owner can see that in his request tab.
- The playground owner can register himself in the system and create a profile. This profile includes the playground name, ID, password, email, phone and default location.
- The system should allow the playground owner to register his e-wallet. This e-wallet will be used to make the player transfer money to the owner.
- The playground owner can requests registering a playground. When registering a playground the playground owner should add all the playground data. This data includes the playground name, location, size, available hours, price per hour, photo of the playground and the cancelation period. A playground is not active until the administrator approve it. Then this request get sent to the administrator, the administrator then can approve this request. Once the request is approved the playground becomes active.



# CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

- The system should allow the playground owner to display and edit any of his data. this data includes how many playgrounds he has, how many of them are available to book, what days and the exact time the playgrounds are available and a photo of the playground.
- The system should allow the playground owner to display all the requests sent to him by a player to rent a certain playground. a request includes the playground to be rented, the exact time and duration it will be rented, the player who sent the request, and if the player team is complete or not. Then the playground owner can accept or deny any of these requests.
- The system should allow the playground owner to display a list of all of his books. This list includes what playgrounds has been booked, to which player, what is the exact time and duration of the book, what playgrounds have not been booked yet, and at what time a playground has not been booked yet.

## Non Functional Requirements

	Details
Performance	<ul style="list-style-type: none"><li>• Activation/deactivation/deleting playground owner operations will be done within 3 seconds</li></ul>

## System Models

### Use Case Model

#### Actors:

Player: this the player who rent playgrounds for a specific duration from the playground owner

Playground owner: this is the person who owns one or multiple playgrounds and want to rent them to the player for a specific price

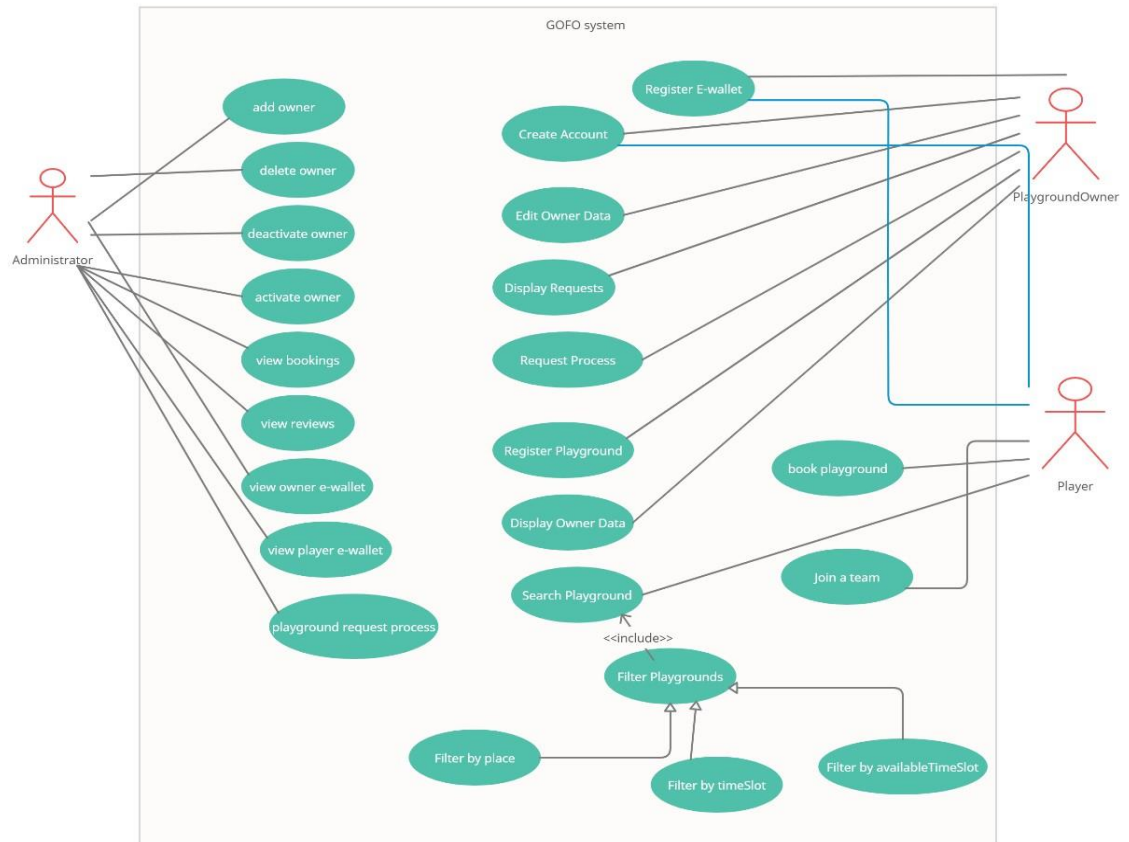
Administrator: someone who oversees the overall operations of the system and ensures that no fraud takes place, he can also add or delete or suspend any playground owner if the owner does some fraudulent activities like double booking or if the playground gets a lot of complaints from the players.



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications





CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

### Enriched User Stories

- User Story #1

User Story ID	US #1
User Story Name	Register e-wallet
Actors	playground owner
Description	<p><b>As</b> a playground owner</p> <p>I <b>like</b> to be able to register my e-wallet</p> <p><b>So</b> I can collect money from the players who rent my playgrounds</p>
Per condition	Playground owner has a profile and signed in the system
Post condition	Playground owner e-wallet is registered into the system
Acceptance Criteria	<p><b>Given</b> I'm signed in the system and in the home page</p> <p><b>When</b> I click register my e-wallet button and fill my e-wallet information</p> <p><b>Then</b> the system register my e-wallet information.</p>

- Scenarios

#### Normal Scenario

Actor Action	System Response
1- playground owner clicks register my e-wallet button	
	2- system displays a form to the playground owner to fill the e-wallet information.
3- playground owner fill the e-wallet information.	
4-playground owner clicks submit	
	5- system register and save the playground e-wallet information





CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

- **Screen Design #1 e-wallet form**

E-wallet number

Enter your e-wallet number

SUBMIT

- **Data Dictionary:**

Element Label	Type/Length	Data Validation / Business Rule
e-wallet number	String/30	Mandatory: Y, updatable: Y/must be a valid e-wallet number and exists in the e-wallet service provider



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

- User Story #2

User Story ID	US #2
User Story Name	Create account
Actors	playground owner
Description	<p><b>As</b> a playground owner</p> <p>I <b>like</b> to be able to create an account in the system</p> <p><b>So</b> I can register my playgrounds and be able to rent them to the players</p>
Per condition	Playground owner is in the sign up page
Post condition	Playground owner account is created and his profile is complete
Acceptance Criteria	<p><b>Given</b> I'm in the sign up page of the system</p> <p><b>When</b> I enter my name, ID, password, email, phone , location and click create account</p> <p><b>Then</b> the system creates an account for me and redirect me to the home page</p>

- Scenarios

### Normal Scenario

Actor Action	System Response
1- playground owner visits the sign up page of the system	
	2- the system displays to the playground owner a form to enter the information needed to create an account.
3- playground owner enters his name 4- playground owner enters his ID 5- playground owner enters his password 6- playground owner enters his email 7- playground owner enters his phone number 8- playground owner enters his default location	



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

9- playground owner clicks sign up button	
	<p>10 – system verifies the data the playground owner entered and the data is valid.</p> <p>11-system verifies if the account ID, email, or phone number already exists in the system or not. and the account doesn't exist in the system</p> <p>11- system creates the account for the user and redirects him to the home page.</p>

### Exceptional Scenario

Actor Action	System Response
1- playground owner visits the sign up page of the system	
<p>3- playground owner enters his name</p> <p>4- playground owner enters his ID</p> <p>5- playground owner enters his password</p> <p>6- playground owner enters his email</p> <p>7- playground owner enters his phone number</p> <p>8- playground owner enters his default location</p> <p>9- playground owner clicks submit</p>	<p>2- the system displays to the playground owner a form to enter the information needed to create an account</p>
	<p>10- system verifies the data the playground owner entered and the data is valid.</p> <p>11- system verifies if the account ID, email, or phone number already exists in the system or not. and the account already exist in the system</p> <p>12- System displays an error message to the user that the account already exists</p>



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

	in the system and request him to re-enter his data.
--	---

Actor Action	System Response
1- playground owner visits the sign up page of the system	
3- playground owner enters his name 4- playground owner enters his ID 5- playground owner enters his password 6- playground owner enters his email 7- playground owner enters his phone number 8- playground owner enters his default location 9- playground owner clicks submit	2- the system displays to the playground owner a form to enter the information needed to create an account
	10- system verifies the data the playground owner entered and the data is valid. 11- System displays an error message to the user and request him to re-enter his data.



## CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

- **Screen Design #2** playground owner sign up form

Sign up

You are a ☒ Playground owner ☐ Player

Name  
Enter your name

ID  
Enter your ID

Email address  
Enter your email address

Password  
Enter your password

Phone number  
Enter your phone number

Location  
Enter your default location

**SIGN UP**

- **Data Dictionary:**

Element Label	Type/Length	Data Validation / Business Rule
Type of account	Radio button	Mandatory: Y, updatable: N LOV: <ul style="list-style-type: none"> <li>Playground owner</li> <li>Player</li> </ul>
name	String/20	Mandatory: Y, updatable: Y
ID	String/20	Mandatory: Y, updatable: N/ must be unique and not alike any
Email address	String/7	Mandatory: Y, updatable: Y/must follow email format
Password	String/20	Mandatory: Y, updatable: Y



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

Element Label	Type/Length	Data Validation / Business Rule
Phone number	String/20	Mandatory: Y, updatable: Y
Location	String/20	Mandatory: Y, updatable: Y

### • User Story #3

<b>User Story ID</b>	US #3
<b>User Story Name</b>	Display owner data
<b>Actors</b>	playground owner
<b>Description</b>	<p><b>As</b> a playground owner</p> <p>I <b>like</b> to be able to display my data</p> <p><b>So</b> I can know how many playgrounds I have , how many of them are available to rent and what days and exact time they are available and a photo of the playground</p>
<b>Per condition</b>	Playground owner is signed in and in the home page
<b>Post condition</b>	System displayed all playground owner data
<b>Acceptance Criteria</b>	<p><b>Given</b> I'm signed in the system and in the home page</p> <p><b>When</b> I click display my data button</p> <p><b>Then</b> the system shows me all my data.</p>

### • Scenarios

#### Normal Scenario

Actor Action	System Response
1- playground owner clicks display my data button	
	2- system displays all the playground owner data. this data includes : how many playgrounds he has , how many of them are available to rent and what days and exact time they are available and a photo of the playground



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

- **Screen Design #3** playgrounds data display

Playground name1

Available to rent

Time available :

19/5/2021 8:00am to 9:00am

20/5/2021 7:00am to 8:30am

20/5/2021 10:00am to 11:00am

**EDIT**



Playground name2

Available to rent

Time available :

19/5/2021 8:00am to 9:00am

20/5/2021 7:00am to 8:30am

20/5/2021 10:00am to 11:00am

**EDIT**



Playground name3

not Available to rent

**EDIT**





CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

- **User Story #4**

<b>User Story ID</b>	US #4
<b>User Story Name</b>	Edit owner data
<b>Actors</b>	playground owner
<b>Description</b>	<b>As</b> a playground owner I <b>like</b> to be able to edit my data <b>So</b> I can change my data whenever necessary
<b>Per condition</b>	Playground owner is signed in and in home page
<b>Post condition</b>	
<b>Acceptance Criteria</b>	<b>Given</b> I'm signed in the system and in the home page <b>When</b> I click edit my data button <b>Then</b> the system displays all of my data and allow me to edit them.

- **Scenarios**

### Normal Scenario

Actor Action	System Response
1- playground owner clicks display my data button	
	2- system displays all the playground owner data. this data includes : how many playgrounds he has , how many of them are available to rent and what days and exact time they are available and a photo of the playground and system allow the user to edit this data.
3-playground owner clicks edit button in any of his listed playgrounds 4- playground owner adjusts the playground data and clicks the save button	
	5- system saves the playground owner data.






## CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

- **Screen Design #4** playground edit data screen

Playground name1  
Available to rent  
Time available :  
19/5/2021 8:00am to 9:00am  
20/5/2021 7:00am to 8:30am  
20/5/2021 10:00am to 11:00am



Playground name  
  
Is it available to rent ☐ Yes ☐ No  
Time available  
From  To

Playground name2  
Available to rent  
Time available :  
19/5/2021 8:00am to 9:00am  
20/5/2021 7:00am to 8:30am  
20/5/2021 10:00am to 11:00am



Playground name3  
not Available to rent



- **Data Dictionary:**

Element Label	Type/Length	Data Validation / Business Rule
Playground name	String/20	Mandatory: Y, updatable: Y
available to rent	Radio button	LOV: <ul style="list-style-type: none"> <li>• Yes</li> <li>• No</li> </ul>
Time available From	String/7	Mandatory: Y, updatable: Y
Time available To	String/7	Mandatory: Y, updatable: Y



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

- User Story #5

User Story ID	US #5
User Story Name	Register playground
Actors	playground owner
Description	<p><b>As</b> a playground owner</p> <p>I <b>like</b> to be able to register a playground I own</p> <p><b>So</b> it can be available in the system and players are allowed to to rent it.</p>
Per condition	Playground owner is signed in and in home page
Post condition	Playground owner registered his playground
Acceptance Criteria	<p><b>Given</b> I'm signed in the system and in the home page</p> <p><b>When</b> I click register my playground button and enters the playground data</p> <p><b>Then</b> the system registers the playground.</p>

- Scenarios

### Normal Scenario

Actor Action	System Response
1- playground owner clicks register a playground button	
	2- system displays a form to the user to enter the playground data. this data includes playground name, location, size, the available hours, the price per hour and the cancellation period.
3- playground owner enters the name, location, size, the available hours, the price per hour and the cancellation period of the playground in the form	



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

4- playground owner clicks submit	
	5- system send the registers the playground in the system and label it as inactive.  6- System send the registration request to the administrator who will approve or disapprove the request.

- **Screen Design #5** register a playground screen

Register a playground

Playground anme

Location

Size

Availble hours

Price per hour

cancellation period

**SUBMIT**



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

- Data Dictionary:**

Element Label	Type/Length	Data Validation / Business Rule
Playground name	String/20	Mandatory: Y, updatable: Y
Location	String/20	Mandatory: Y, updatable: N
Size	String/15	Mandatory: Y, updatable: N
Available hours	String/35	Mandatory: Y, updatable: Y
Price per hour	String/10	Mandatory: Y, updatable: Y
Cancellation period	String/10	Mandatory: Y, updatable: Y

- User Story #6**

<b>User Story ID</b>	US #6
<b>User Story Name</b>	Display requests
<b>Actors</b>	Playground owner
<b>Description</b>	<p><b>As</b> a playground owner</p> <p><b>I like</b> to be able to see all the requests players on the system have mad</p> <p><b>So</b> I can accept or reject these requests.</p>
<b>Per condition</b>	Playground owner is signed in and in home page
<b>Post condition</b>	Playground viewed all the player requests.
<b>Acceptance Criteria</b>	<p><b>Given</b> I'm signed in the system and in the home page</p> <p><b>When</b> click player requests button</p> <p><b>Then</b> the system displays all the player requests to me.</p>

- Scenarios**

**Normal Scenario**

Actor Action	System Response
--------------	-----------------



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

1- user clicks <u>player requests</u> button.	
	3- system displays in a form of a list all player request made to any playground the playground owner has. Each item in the list contain the player who made the request, the playground to be rented, exact time and duration it will be rented and if the player has a complete team or not. in addition to this information there are two buttons [ACCEPT] and [REJECT] a playground owner can accept or reject the request through these two buttons.
5- playground owner can view the requests and can accept or reject a certain request	
	6- the system labels the request as accepted or rejected and inform the player who made the request.



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

- **Screen Design #6** players requests screen

Player name1

Playground name1

From 8:00am to 945am

Team is complete

ACCEPT

REJECT

Player name2

Playground name2

From 8:00am to 945am

Team is not complete

ACCEPT

REJECT

Player name3

Playground name3

From 8:00am to 945am

Team is complete

ACCEPT

REJECT



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

- User Story #7

User Story ID	US #7
User Story Name	Display Books
Actors	Playground owner
Description	<p>As a playground owner</p> <p>I like to be able to see all the books</p> <p>So I can know what playgrounds have been booked, when, and to whom</p>
Per condition	Playground owner is signed in the system and in the homepage
Post condition	System displayed all the books to the owner
Acceptance Criteria	<p>Given I'm signed in and in the home page</p> <p>When I click my books button</p> <p>Then the system show me all books that have been made to any playground</p>

- Scenarios

### Normal Scenario

Actor Action	System Response
1- User click books button .	
	2- System displays to the user all the bookings that have been made and at what time and to which player, also it will displays all the player grounds that have not been booked yet, and at which time.



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

- **Screen Design #7 owner bookings screen**

### Playground name1

Booked to player player1

19/5/2021 From 8:00am to 9:30am

Booked to player player2

19/5/2021 From 10:30am to 11:30am

not booked

18/5/2021 From 8:00pm to 10:00pm

17/5/2021 From 8:35am to 9:50am

### Playground name2

Booked to player player3

19/5/2021 From 8:00am to 9:30am

not booked

18/5/2021 From 8:00pm to 10:00pm

17/5/2021 From 8:35am to 9:50am

### Playground name3

not booked

18/5/2021 From 8:00pm to 10:00pm

17/5/2021 From 8:35am to 9:50am





CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

- User Story #8

User Story ID	US #8
User Story Name	Add/Delete playground Owner
Actors	Administrator
Description	As an administrator. I like to be able to add or delete playground owners. So I can control the system and prevent any fraud actions.
Per condition	I am in the playground owners users' infos page
Post condition	Administrator added/deleted playground owner
Acceptance Criteria	Given I'm in the playground owners users' infos page. When I tap on add/delete owner button . Then owner user is added/deleted.

- Scenarios

### Normal Scenario

Actor Action	System Response
1- Administrator taps on <b>users info</b> button.	
	2- system displays either <b>playground owners</b> or <b>players</b> .
3- Administrator taps on <b>playground owners</b> button.	
	4- System displays list of playground owners information and <b>add</b> button.
5- Administrator taps on <b>add</b> button.	
	6- System displays a registration form for a playground owner.
7- Administrator enters playground owner name	



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

<p>8- Administrator enters playground owner ID</p> <p>9- Administrator enters playground owner password</p> <p>10- Administrator enters playground owner email</p> <p>11- Administrator enters playground owner phone number</p> <p>12- Administrator enters playground owner default location</p> <p>13- Administrator enters playground owner clicks submit</p>	
	<p>14- system verifies the data of playground owner entered and the data is valid.</p> <p>15- system verifies if the account ID, email, or phone number already exists in the system or not. and the account doesn't exist in the system</p> <p>16- system creates/adds the account and redirects the administrator to the home page.</p>

### Exceptional Scenario

Actor Action	System Response
1- Administrator taps on add button.	
	13- System displays a registration form for a playground owner.
<p>14- Administrator enters playground owner name</p> <p>15- Administrator enters playground owner ID</p>	



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

16- Administrator enters playground owner password 17- Administrator enters playground owner email 18- Administrator enters playground owner phone number 19- Administrator enters playground owner default location 20- Administrator enters playground owner clicks submit	
	21- system verifies the data the administrator entered and the data is valid. 22- system verifies if the account ID, email, or phone number already exists in the system or not. and the account <b>already exist in the system.</b> 23- System displays an error message to the user that the account already exists in the system and request him to re-enter his data.

Actor Action	System Response
1- Administrator taps on add button.	
	12- System displays a registration form for a playground owner.
13- Administrator enters playground owner name 14- Administrator enters playground owner ID	



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

15- Administrator enters playground owner password 16- Administrator enters playground owner email 17- Administrator enters playground owner phone number 18- Administrator enters playground owner default location 19- Administrator enters playground owner clicks submit	
	20- system verifies the data the Administrator entered and the data is invalid. 21- System displays an error message to the user and request him to re-enter his data.

Actor Action	System Response
1- Administrator taps on delete button.	
	2- System displays a list of owners.
3- Administrator chooses which owner he wants to delete. 4- Administrator taps on <b>Delete</b> button.	
	5- system deletes the data of the owner that the administrator has chosen. 6- System displays a confirmation message "owner has been deleted"



CS251: Phase 1 – Team Name  
Project: <Project Name>

## Software Requirements Specifications

- Screen Design #8 add/delete owner screen

The image displays three mobile application screens for user management:

- Users infos:** A screen titled "Users infos" showing a list of "Playground owners" (Owner1 to Owner10). It includes buttons for "ADD OWNER", "DEACTIVATED OWNERS", and "DELETE OWNER".
- Add Owner:** A screen titled "Add Owner" with input fields for Name, ID, Email address, Password, Phone number, and Location. It features an "Add" button at the bottom.
- Delete Owner:** A screen titled "Delete Owner" with a "SELECT ALL" button and a list of owners (Owner1 to Owner10) with checkboxes. It includes a "DELETE" button at the bottom.

- Data Dictionary:



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

Element Label	Type/Length	Data Validation / Business Rule
name	String/20	Mandatory: Y, updatable: Y
ID	String/20	Mandatory: Y, updatable: N/ must be unique and not alike any
Email address	String/7	Mandatory: Y, updatable: Y/must follow email format
Password	String/20	Mandatory: Y, updatable: Y
Phone number	String/20	Mandatory: Y, updatable: Y
Location	String/20	Mandatory: Y, updatable: Y

### • User Story #9

<b>User Story ID</b>	US #9
<b>User Story Name</b>	Activate/Deactivate owner's account.
<b>Actors</b>	Administrator
<b>Description</b>	<p><b>As</b> an administrator.</p> <p>I <b>like</b> to be able to Activate or Deactivate playground owner's account.</p> <p><b>So</b> I can control the system and prevent any fraud actions.</p>
<b>Per condition</b>	I am in the playground owners users' infos page
<b>Post condition</b>	Administrator activated/deactivated playground owner's account
<b>Acceptance Criteria</b>	<p><b>Given</b> I'm in the playground owners users' infos page.</p> <p><b>When</b> I tap on activate/deactivate owner button .</p> <p><b>Then</b> owner user is activated/deactivated.</p>

### • Scenarios



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

### Normal Scenario

Actor Action	System Response
1- Administrator taps on <b>users info</b> button.	
	2- system displays either <b>playground owners</b> or <b>players</b> .
3- Administrator taps on <b>playground owners</b> button.	
	4- System displays list of playground owners.
5- Administrator chooses which owner he wants to deactivate.	
6-Administrator taps on <b>deactivate</b> button.	
	6- System asks the admin whether he wants to deactivate the owner's account <b>permanently</b> or <b>temporarily</b> .
7- Administrator taps on <b>permanently</b> button.	
	8- System deactivates the owner's account permanently and deletes owner's data. 9- System displays a confirmation messages " <b>owner has been deactivated permanently</b> "

### Exceptional Scenario

Actor Action	System Response
1- Administrator taps on <b>users info</b> button.	
	2- system displays either <b>playground owners</b> or <b>players</b> .
3- Administrator taps on <b>playground owners</b> button.	
	4- System displays list of playground owners.



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

5- Administrator chooses which owner he wants to deactivate.	
6-Administrator taps on <b>deactivate</b> button.	
	6- System asks the admin whether he wants to deactivate the owner's account <b>permanently</b> or <b>temporarily</b> .
2- Administrator taps on <b>Temporarily</b> button.	
	3- System deactivates the owner's account temporarily. 4- System displays a confirmation messages " <b>owner has been deactivated</b> "

Actor Action	System Response
1- Administrator taps on <b>users info</b> button.	
	2- system displays either <b>playground owners</b> or <b>players</b> .
3- Administrator taps on <b>playground owners</b> button.	
	4- System displays list of playground owners and " <b>deactivated accounts</b> " button
5- Administrator taps on <b>deactivated accounts</b> button.	
	6- System displays a list of deactivated accounts
7- Administrator chooses which account he wants to re-activate. 8- Administrator taps on <b>re-activate</b> button.	
	9-System re-activates the owner's account and restore owner's data.





CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

10-System displays a confirmation messages **“owner has been re-activated” permanently**”

- Screen Design #9 add owner/deactivate owner screen

The image displays three mobile application screens for user management:

- Users infos:** A screen titled "Users infos" showing a list of "Playground owners" (Owner1 to Owner10). It includes buttons for "ADD OWNER", "DEACTIVATED OWNERS", and "DELETE OWNER".
- Owner 1:** A screen showing the details for "Owner 1". It includes input fields for Name, ID (123), Email address (Owner1@gmail.com), Password (masked with asterisks), Phone number (0128333015), and Location (shobra-cairo). At the bottom, there are buttons for "Deactivate temporarily" and "Deactivate permanently".
- Deactivated Users:** A screen titled "Deactivated Users" showing a list of deactivated users (Owner1 to Owner10). It includes a "SELECT ALL" button and a "RE-ACTIVATE" button at the bottom.



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

- User Story #10**

<b>User Story ID</b>	US #10
<b>User Story Name</b>	View bookings
<b>Actors</b>	Administrator
<b>Description</b>	<p><b>As</b> an administrator.</p> <p>I <b>like</b> to be able to view all bookings</p> <p><b>So</b> I can control the system and see if fraud takes place</p>
<b>Per condition</b>	I am in the homepage
<b>Post condition</b>	Administrator viewed bookings
<b>Acceptance Criteria</b>	<p><b>Given</b> I'm in the administrator homepage.</p> <p><b>When</b> I tap on bookings button .</p> <p><b>Then</b> all bookings are viewed.</p>

- Scenarios**

**Normal Scenario**

Actor Action	System Response
1- Administrator taps on <b>Bookings</b> button in homepage	
	2- system displays all bookings that are already made between the owner and the player

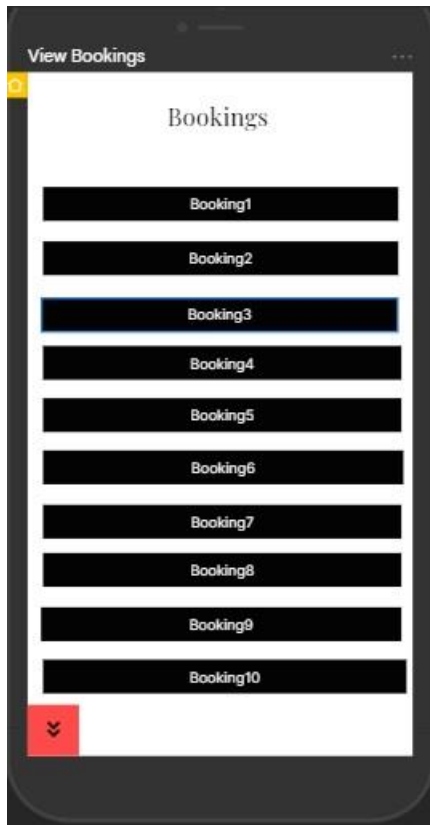


CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

- Screen Design #10 view bookings screen





CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

- User Story #11

User Story ID	US #11
User Story Name	View reviews
Actors	Administrator
Description	<p><b>As</b> an administrator.</p> <p>I <b>like</b> to be able to view all reviews.</p> <p><b>So</b> I can control the system and see if there is a complaint.</p>
Per condition	I am in the homepage
Post condition	Administrator viewed reviews
Acceptance Criteria	<p><b>Given</b> I'm in the administrator homepage.</p> <p><b>When</b> I tap on playgrounds button .</p> <p><b>Then</b> all playgrounds' reviews are viewed.</p>

- Scenarios

### Normal Scenario

Actor Action	System Response
1- Administrator taps on <b>Playgrounds</b> button in homepage	
	2- system displays all reviews on playgrounds.

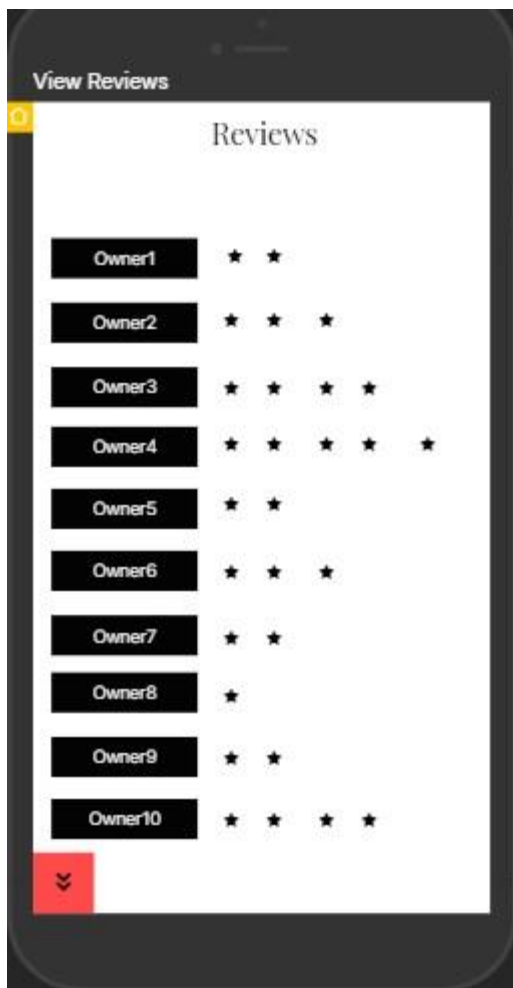


CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

- Screen Design #11 reviews screen





CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

- **User Story #12**

<b>User Story ID</b>	US #12
<b>User Story Name</b>	View player/owner e-wallet
<b>Actors</b>	Administrator
<b>Description</b>	<p><b>As</b> an administrator.</p> <p><b>I like</b> to be able to view player/owner's e-wallet.</p> <p><b>So</b> I can control the system and see if fraud takes place.</p>
<b>Per condition</b>	I am in the homepage
<b>Post condition</b>	Administrator viewed player/owner's e-wallet
<b>Acceptance Criteria</b>	<p><b>Given</b> I'm in the administrator homepage.</p> <p><b>When</b> I tap on e-wallets button.</p> <p><b>Then</b> e-wallets are viewed.</p>

- **Scenarios**

**Normal Scenario**

Actor Action	System Response
1- Administrator taps on <b>E-Wallets</b> button in homepage	
	2- system displays menu <b>includes player e-wallets</b> and <b>playground owner e-wallets</b> buttons.
3- Administrator taps on <b>player e-wallet</b> button.	
	4 – system displays all players
5 – Administrator chooses which player he wants to see his e-wallet	
	6 – system displays player's e-wallet information



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

### Exceptional Scenario

Actor Action	System Response
1- Administrator taps on <b>E-Wallets</b> button in homepage	
	2- system displays menu includes <b>player e-wallets</b> and <b>playground owner e-wallets</b> buttons.
3- Administrator taps on <b>playground owner e-wallet</b> button.	
	4 – system displays all players
5 – Administrator chooses which playground owner he wants to see his e-wallet	
	6 - system displays playground owner's e-wallet information

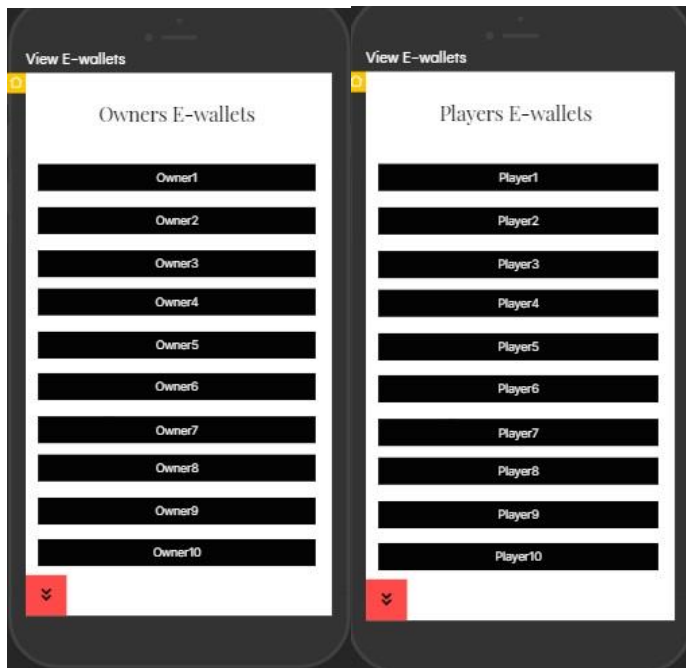


CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

- Screen Design #12 e-wallet screen



- User Story #13

User Story ID	US #13
User Story Name	Create account
Actors	Player
Description	As a player I like to be able to create an account in the system So I can find the playground, i need to play in it
Per condition	Player is in the sign up page
Post condition	Player account was created and his profile is complete
Acceptance Criteria	Given I am in the sign in page of the system When I enter my name, password, email, phone number and address and click create account





CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

**Then** the system creates my account and returns me to the home page

- Scenarios**

### Normal Scenario

Actor Action	System Response
1- Player visits the sign in page of the system	
	2- The system displays a form to the player to enter the information needed to create an account
3- Player enters his name 4- Player enters his password 5- Player enters his email 6- Player enters his phone number 7- Player enters his address 8- Player clicks submit	
	9- The system verifies that the data entered by the player is correct 10- System verifies if the account email and phone number already exists in the system or not. and the account doesn't exist in the system 11- System creates the account for the user and returns him to the home page.

### Exceptional Scenario

Actor Action	System Response
1- Player visits the sign in page of the system	
	2- The system displays a form to the player to enter the information needed to create an account



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

3- Player enters his name 4- Player enters his password 5- Player enters his email 6- Player enters his phone number 7- Player enters his address 8- Player clicks submit	
	9- The system verifies that the data entered by the player is correct 10- System verifies if the account email and phone number already exists in the system or not. and the account doesn't exist in the system 11- System displays an error message to the user that the account already exists in the system and request him to re-enter his data



CS251: Phase 1 – Team Name  
Project: <Project Name>

## Software Requirements Specifications

- Screen Design #13 player sign up form

### Sign up

you are a ☐ playground owner ☒ player

Name

Email

Address

Phone number

Password

**SIGN UP**



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

- Data Dictionary:**

Element Label	Type/Length	Data Validation / Business Rule
Type of account	Radio button	Mandatory: Y, updatable: N LOV: <ul style="list-style-type: none"> <li>Playground owner</li> <li>Player</li> </ul>
name	String/30	Mandatory: Y, updatable: Y
Email	String/30	Mandatory: Y, updatable: Y/must follow email format
Address	String/40	Mandatory: Y, updatable: Y
Phone number	String/20	Mandatory: Y, updatable: Y
Password	String/20	Mandatory: Y, updatable: Y

- User Story #14**

<b>User Story ID</b>	US #14
<b>User Story Name</b>	Register E-wallet
<b>Actors</b>	Player
<b>Description</b>	<p><b>As</b> a player</p> <p>I <b>like</b> to be able to register my e-wallet</p> <p><b>So</b> I can paid money to the playground owner when I books the playground</p>
<b>Per condition</b>	Player has a profile and signed in the system
<b>Post condition</b>	Player E-wallet is registered into the system



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

### Acceptance Criteria

**Given** I'm signed in the system and in the home page  
**When** I click register my E-wallet button and fill my E-wallet information  
**Then** the system register my E-wallet information.

- Scenarios

#### Normal Scenario

Actor Action	System Response
1- Player clicks register my E-wallet button	
	2- System displays a form to the playground owner to fill the E-wallet information.
3- Player fill the E-wallet information. 4-Player clicks submit	
	5- System register and save the playground E-wallet information

- Screen Design #14 e-wallet form screen



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

E-wallet number

Enter your E-wallet number

SUBMIT

- Data Dictionary:**

Element Label	Type/Length	Data Validation / Business Rule
e-wallet number	String/30	Mandatory: Y, updatable: Y/must be a valid e-wallet number and exists in the e-wallet service provider

- User Story #15**

<b>User Story ID</b>	US #15
<b>User Story Name</b>	Booking playground
<b>Actors</b>	Player
<b>Description</b>	<p><b>As</b> a player</p> <p><b>I like</b> to be able to booking a playground</p> <p><b>So</b> I can research the best playground, i want to play in it</p>
<b>Per condition</b>	Player is signed in and in home page
<b>Post condition</b>	Player booked the playground
<b>Acceptance Criteria</b>	<p><b>Given</b> I am signed in the system and in the home page</p> <p><b>When</b> I click booking playground button and choose the playground</p>



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

**Then** the system books the playground for me.

- Scenarios**

### Normal Scenario

Actor Action	System Response
1- Player clicks booking playground button	
	2- The system displays all playgrounds available now
3- Player chooses the playground he wants to play in 5- Player clicks submit	4- The system displays the playground information
	6- System makes this playground unavailable for reservation 7- The system filters the playgrounds.



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

- Screen Design #15 playgrounds available screen

**Playgrounds available**

Playgroung name1

Avalaible in

20/5/2021 from 8:00pm to 10:00pm

Book

21/5/2021 from 6:00pm to 9:00pm

Book

22/5/2021 from 10:00am to 7:00pm

Book

Playgroung name2

Avalaible in

20/5/2021 from 11:00am to 5:00pm

Book

20/5/2021 from 9:00pm to 11:00pm

21/5/2021 from 1:00pm to 6:00pm

Book

21/5/2021 from 11:00pm to 1:00am

22/5/2021 from 3:00am to 9:00pm

Book

Playgroung name3

Avalaible in

20/5/2021 from 2:00pm to 8:00pm

Book

21/5/2021 from 6:00pm to 10:00pm

Book

22/5/2021 from 7:00pm to 9:00pm

Book





CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

- **User Story #16**

<b>User Story ID</b>	US #16
<b>User Story Name</b>	Join a team
<b>Actors</b>	Player
<b>Description</b>	<p><b>As</b> a player</p> <p>I <b>like</b> to be able to join a team</p> <p><b>So</b> I can play with another team</p>
<b>Per condition</b>	Player is signed in and in home page
<b>Post condition</b>	Player joined to team
<b>Acceptance Criteria</b>	<p><b>Given</b> I am signed in the system and in the home page</p> <p><b>When</b> I click join team button and chooses a team they need a player</p> <p><b>Then</b> the system makes me one of the team members</p>

- **Scenarios**

**Normal Scenario**

Actor Action	System Response
1- Player clicks join team button	
	2- The system displays all teams available now
3- Player chooses the team he wants to play with them	4- The system displays the team information like their names and phones
5- Player clicks submit	
	6- System registers me as a player on the team
	7- The system filters the teams



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

- Screen Design #16 Teams available to join screen

**Teams available**

Team name1

Join

Team name2

Join

Team name3

Join

Team name4

Join

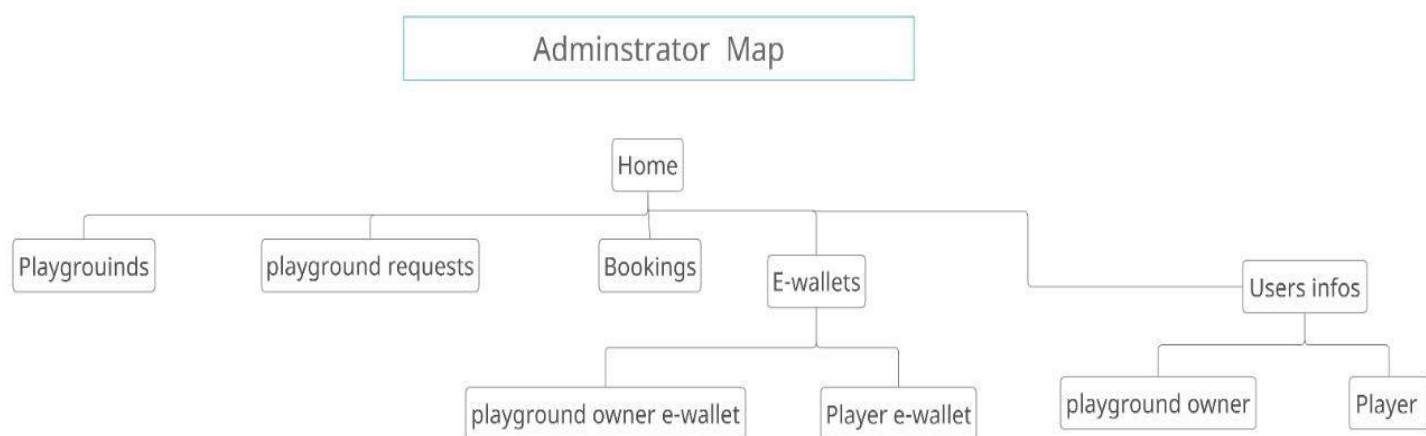


CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

### System Navigation Map

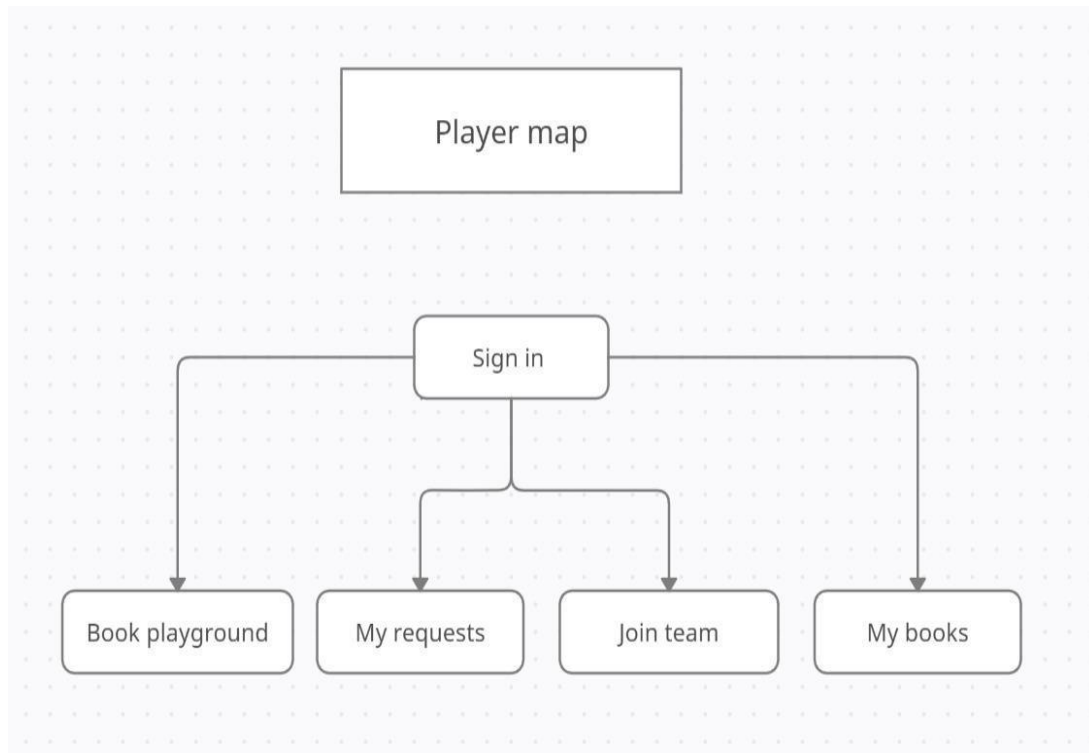




CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications

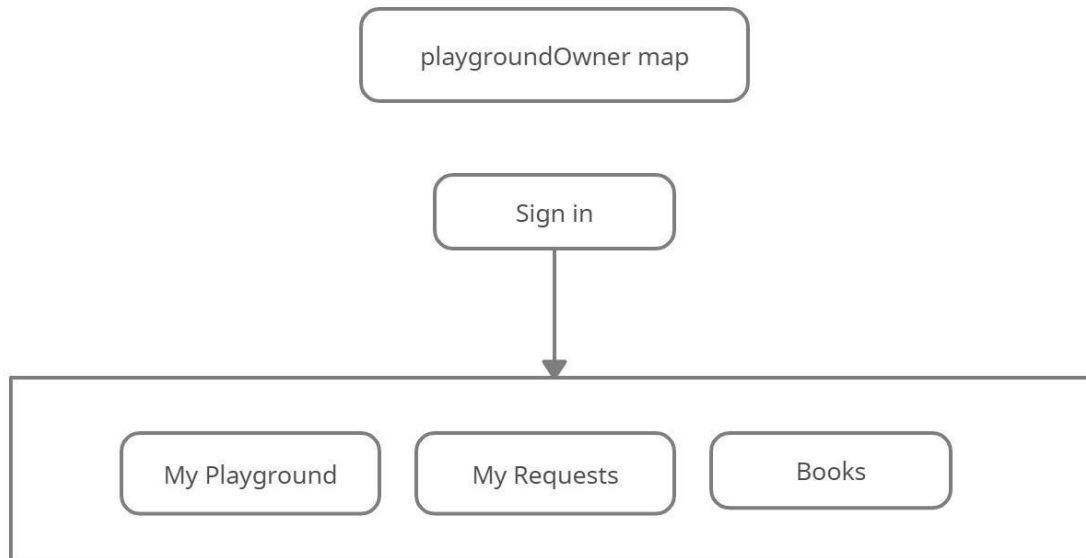




CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications



### Tools

- <https://createely.com/>
- <https://uizard.io/>

### Ownership Report

Student	Items he created
Peter Essam Maguid Isaac	<b>System purpose and Part of Use Case Model, functional &amp; Non-Functional Requirements, definitions, and system navigation map.</b> Enriched user story #8, #9, #10, #11, #12
Abdallah Ragab Abdallah	<b>System scope and Part of Use Case Model, functional Requirements, definitions, and system navigation map</b> Enriched user story #13, #14, #15, #16
Shehab eldin Khaled mohamed	<b>Part of Use Case Model, functional Requirements, and system navigation map</b> Enriched user story #1, #2, #3, #4, #5, #6, #7



CS251: Phase 1 – Team Name

Project: <Project Name>

## Software Requirements Specifications