



Faculty of Computers and Artificial Intelligence

Cairo University

Final Assessment Project

Project ID : PM-68

Course Title: Introduction to Software Engineering (CS251)

Semester: Second Semester

Date: May 2020

Instructors: Dr. Mohamed El-ramly

Prepared by:

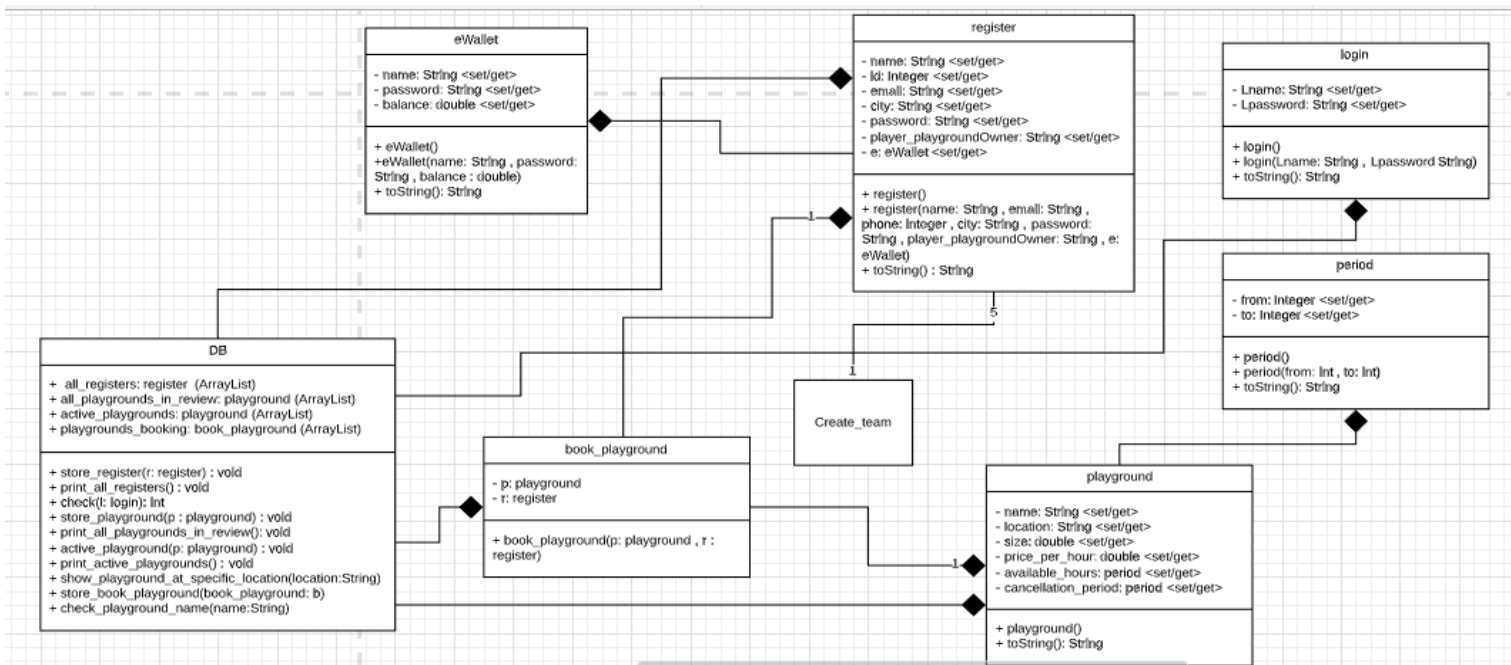
Student Names	Students IDs
Saeed Mohamed Ahmed Gooda	20180119
Mohamed Elsayed AbdElhammed	20180217

Option 1

Task 1 – System Diagram

Project Phase 2: System Design

Task 1 - Class Diagram:



Class Diagram Explanation:

In this diagram we make more classes to help the player to book playground and the playground owner to make his playground booked easily.

1- **Register class:** In this class any user can register as player or playground owner by enter his information (name , email , phone , city , password , player_playgroundOwner (that determine if he is player or playground owner) , e (variable from **eWallet class** that enable people of adding their wallet) so make between register and eWallet a composition relation. After user add all information , we will add it to ArrayList called "all_registers" in **DB class**.

2- **eWallet class:** In this class any user can add his eWallet information (name , password and balance).

- 3- Login class:** In this class any user can login with his name and password(that he register with them) then we will call "check" function (that exist at DB class) that take variable from login class and check if the name and password exist or not.
- 4- playground class:** In this class any playground owner can add his playground information (name , location , size , price per hour , available hours , cancellation period) . available hours and cancellation are a variable taken from **period class**. After playground owner add all information we will add it to ArrayList called "all_playgrounds_in_review" in **DB class**.
- 5- Period class:** In this class you can add specific time period (from , to).
- 6- Create team class :** any player can create team from 5 players so we make a association relation.
- 7- book_playground class :** In this class any player can book a playground. So this class take the variable from playground and from register(users).
- 8- DB class:** contain all users , playgrounds and book playgrounds data.
- 1- all_registers ArrayList : contain all users information and there is a function that enable to store the information called "store_register" and take variable from register class as a parameter.
 - 2- Check(variable of login class) : this function check if the username and password of the login is exist or not.
 - 3- all_playgrounds_in_review ArrayList : contain all playgrounds in review that will approve by administrator and also there is a function that store the playground called "store_playground" and take variable from playground as a parameter.
 - 4- Active_playgrounds ArrayList : contain all active playgrounds.
 - 5- show_playground_at_specific_location Function that take a location parameter and showing all active playgrounds in this location.
 - 6- playgrounds_booking ArrayList : contain the playground and the user who book it.

Task 2 Sequence Diagrams:

1- Register new user:

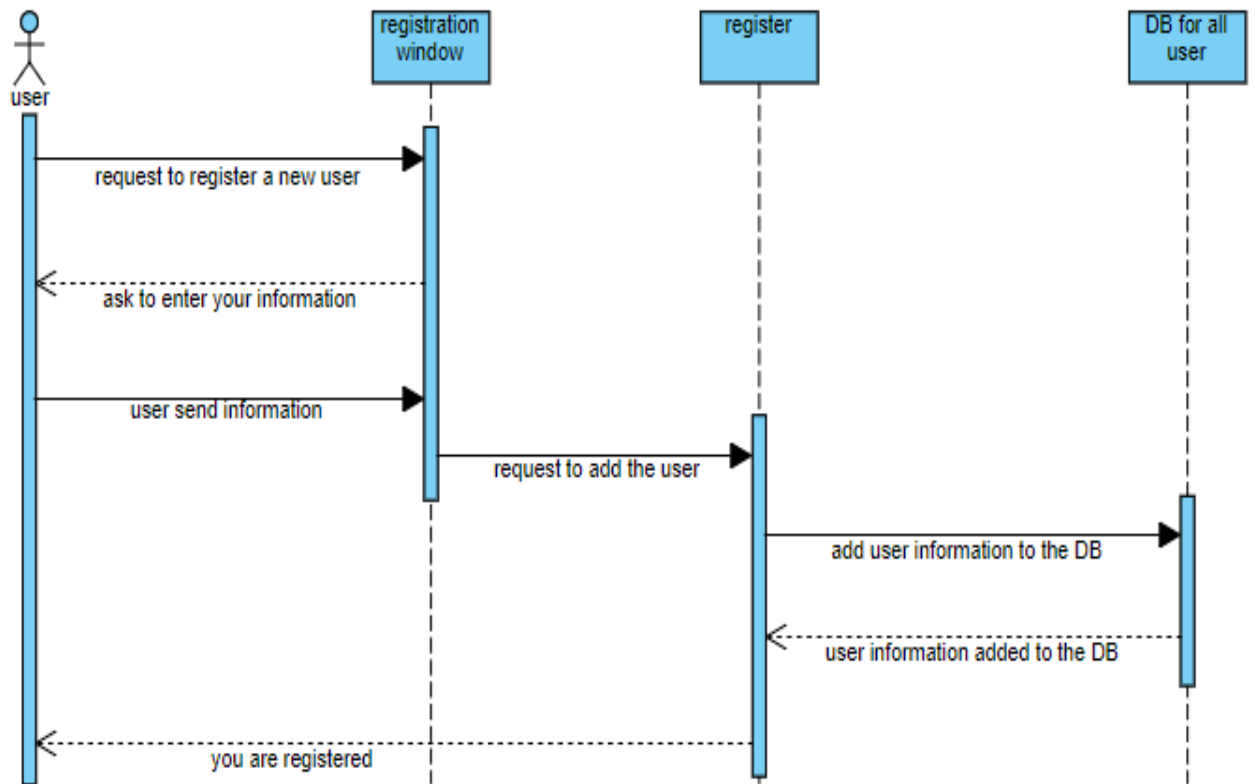


Diagram explanation:

In this sequence diagram user request to register (click on register button) then registration window will return a form ask him to enter his information then user will click on register button it will request to add the user and add him at the database (In this project we use ArrayList) and it will send to him that the information added to the database and return to the user that " you are registered".

2- User Login:

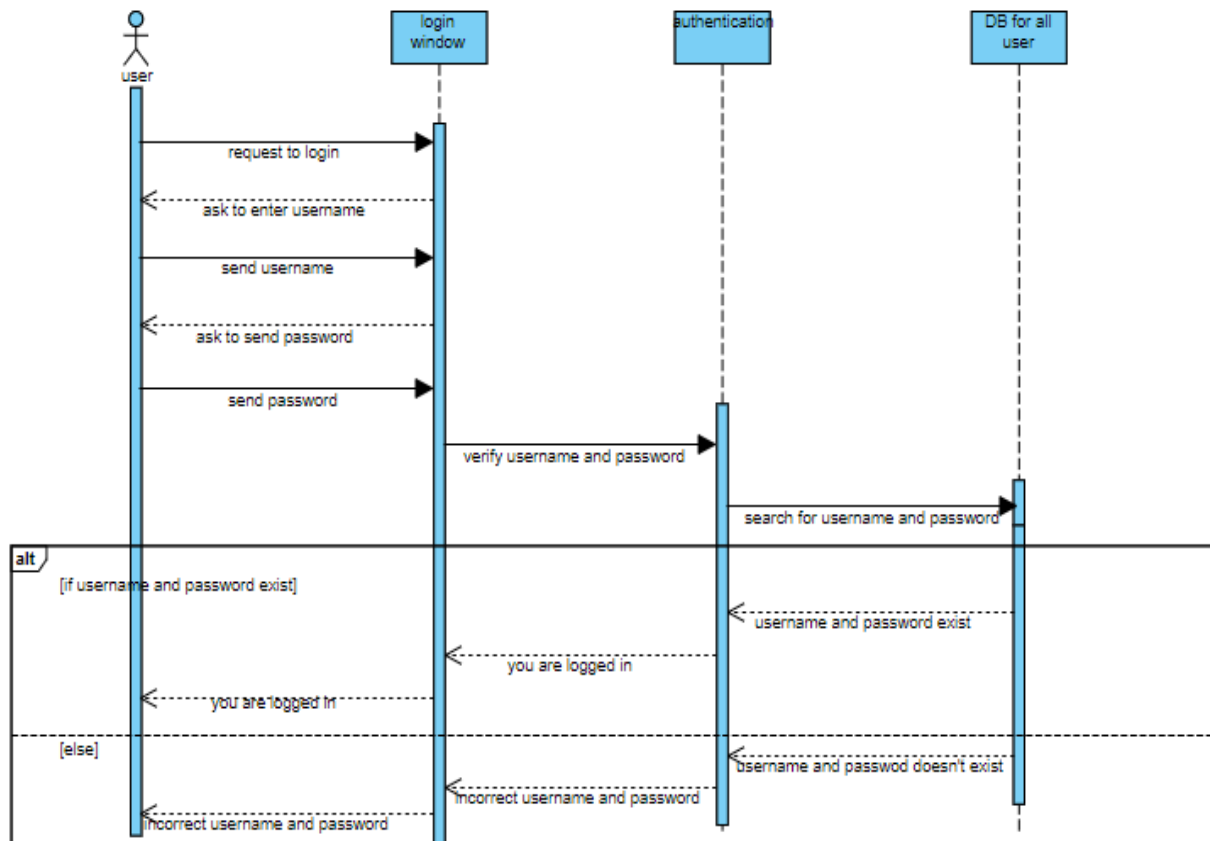


Diagram explanation:

In this sequence diagram user request to login (click on login button) , then login window request him to send username then user send it , then login window request to send password then user send it , then click on login (login window will send the information to authentication to verify it) , then it will search about username and password at the database(ArrayList) to know if it is exist or not. To know if it exist or not we will make alt (if condition) if (username and password exist at the database) it will return "You are logged in" else it will return "Incorrect username or password".

3- Add Playground(by playground owner):

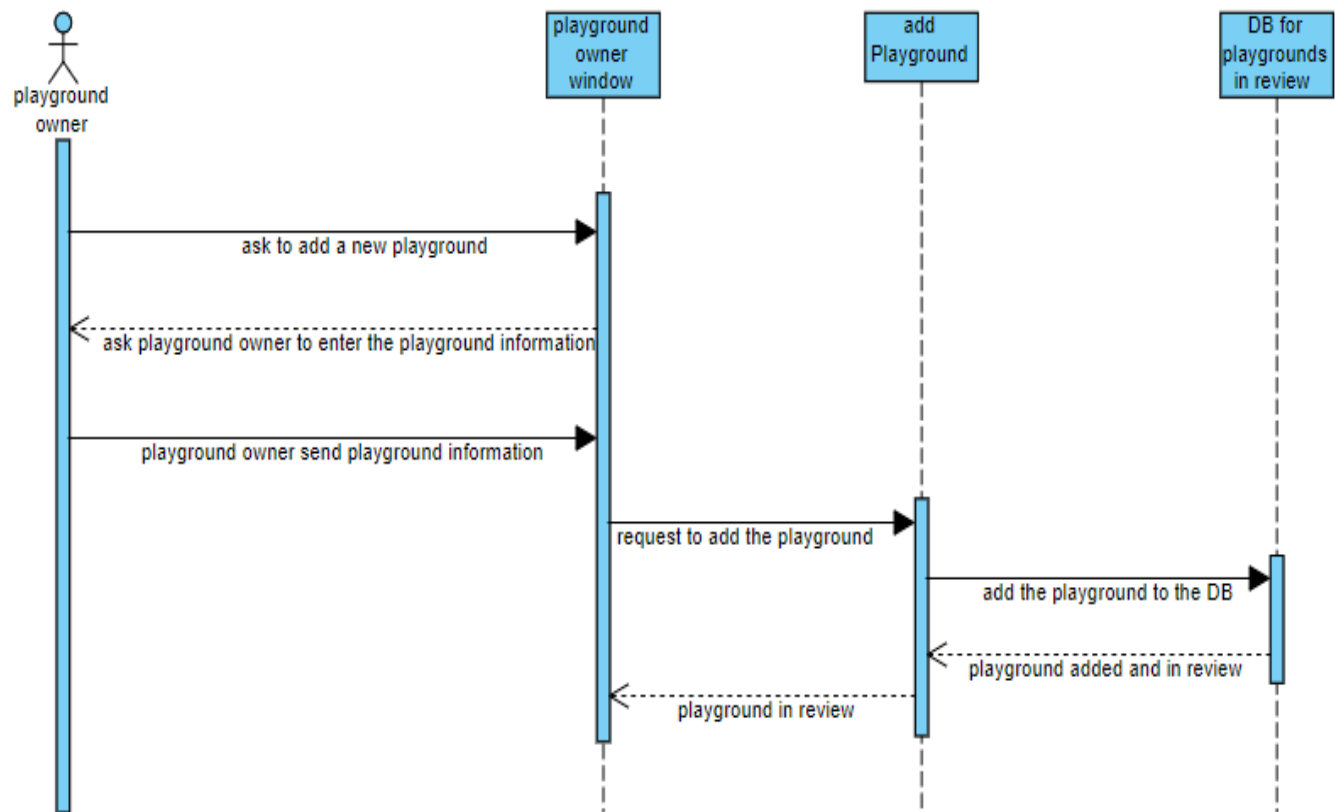


Diagram explanation:

In this diagram any playground owner can add his playground. First : he request to add the playground then playground owner window will ask him to add the playground information (name , location , size , price per hour , available hours and cancellation period) then request to add the playground then it will add to the playgrounds in review database (ArrayList) and it will return to the playground Owner that playground is added and in review.

4- Approve Playground:

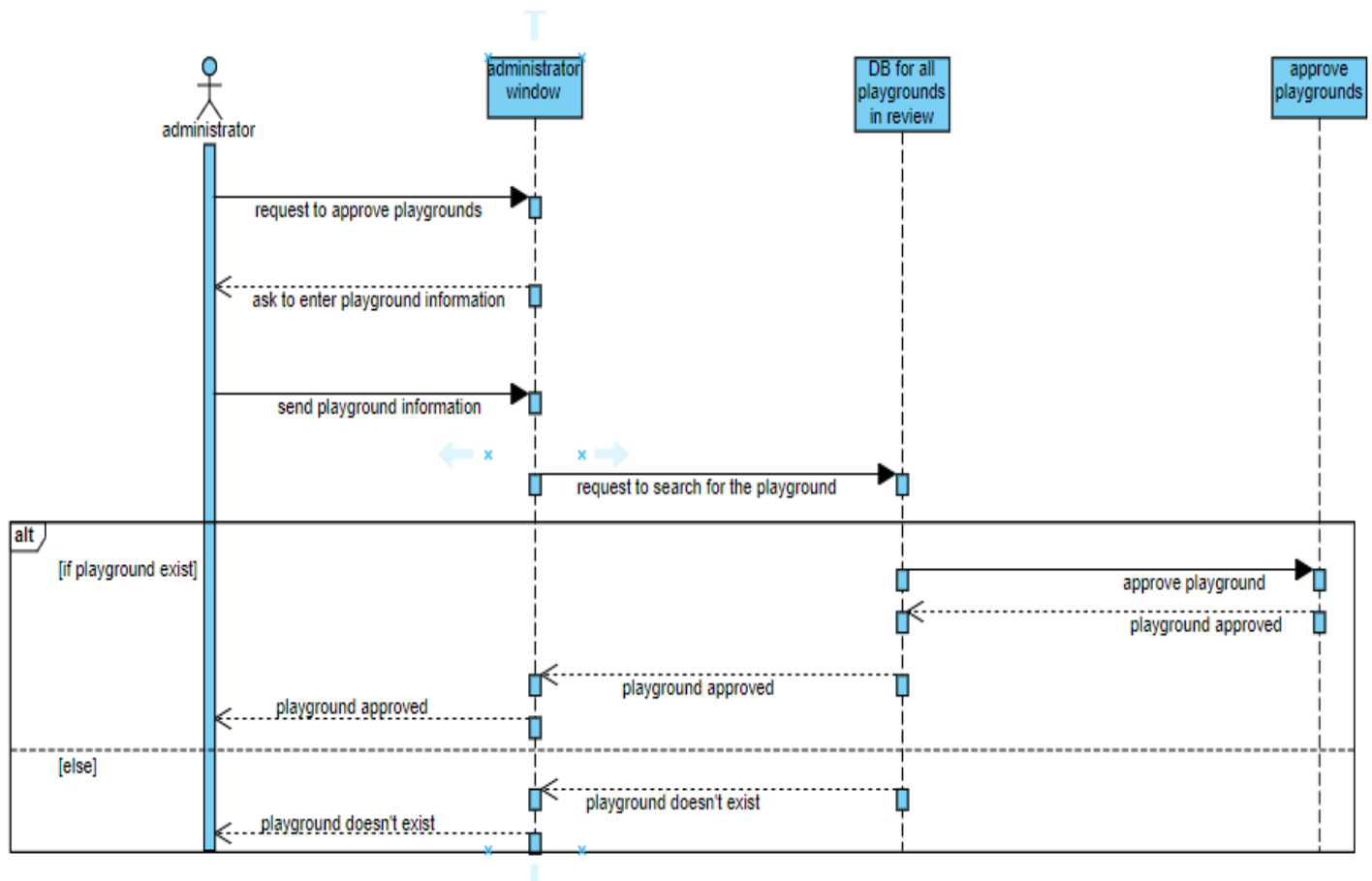


Diagram explanation:

In this diagram we need to know how administrator approve playgrounds that in review. First, administrator request to approve playground then administration menu will ask him to enter the playground information(name) that you want to approve then administrator send the information then administration menu will request to search for the playground in DB if it exist , it will approve it and send to the administrator that playground approved, if it doesn't exist , it will send to the administrator that playground information incorrect and playground doesn't exist.

5- Book Playground:

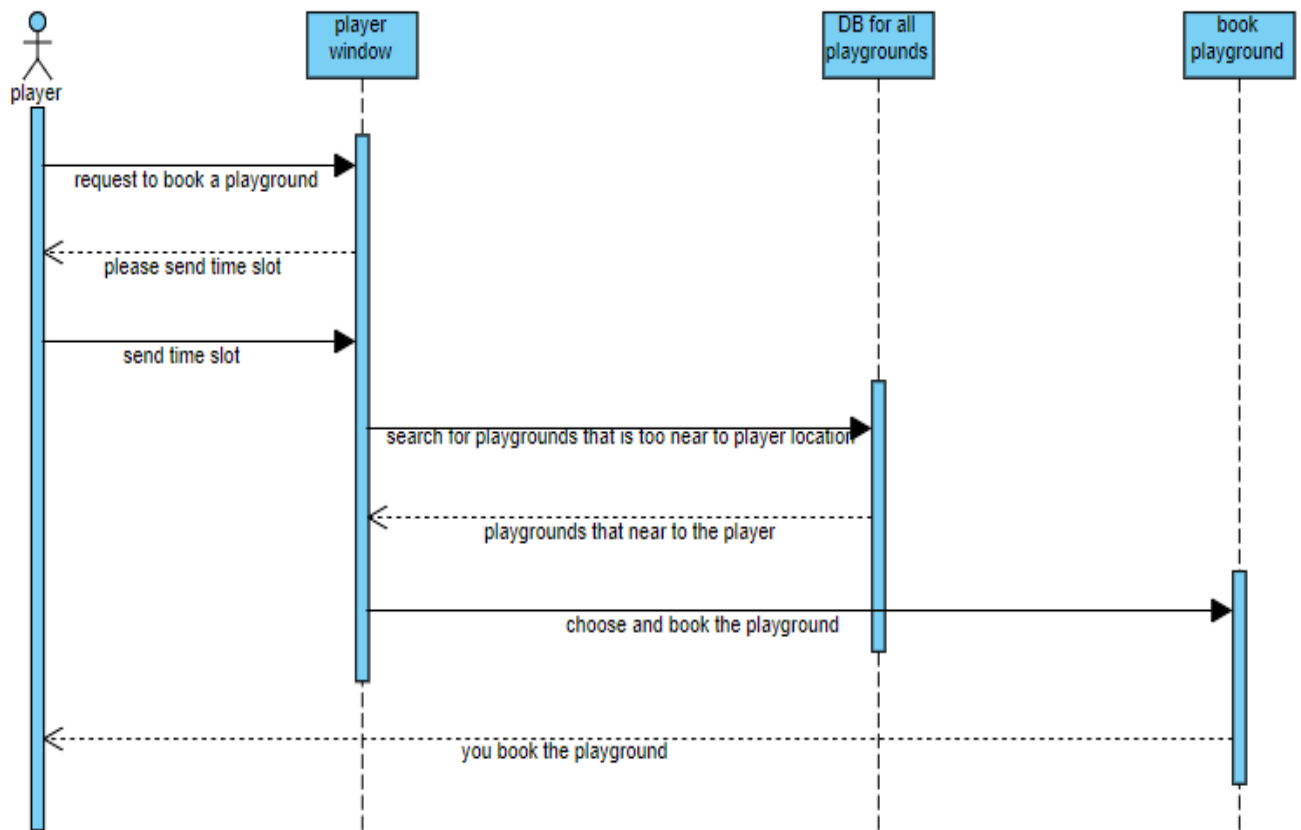


Diagram explanation:

In this diagram we need to know how player book the playground. First, player request to add the playground , then player window will ask him to send time slot(from , to) , then it will ask the active playgrounds database(ArrayList) to send all playgrounds that have the same location , then it will send all playgrounds to him and ask him to choose the appropriate playground ,then book the playground and return to the player that playground is booked.

6- Create Team:

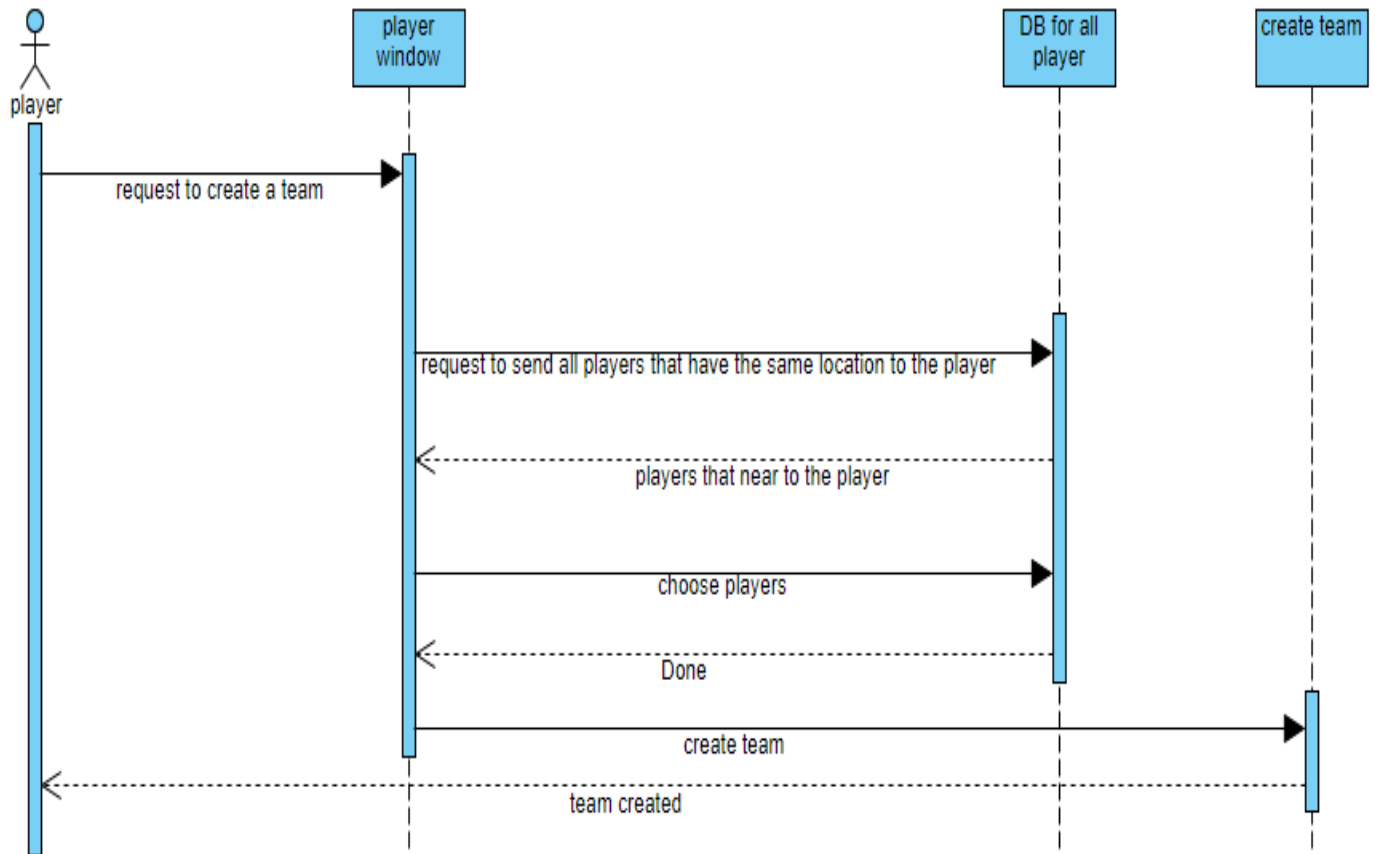
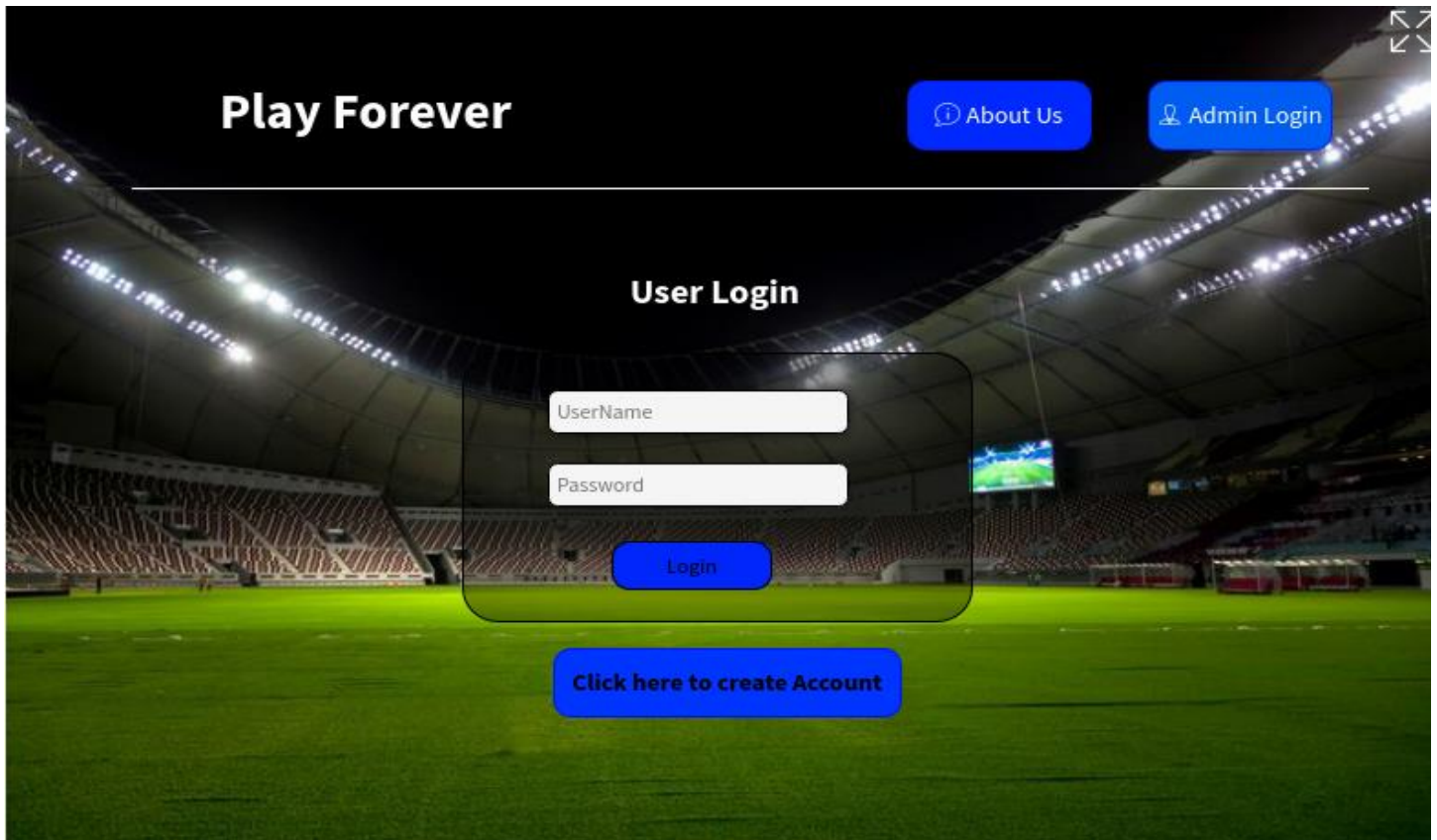


Diagram explanation:

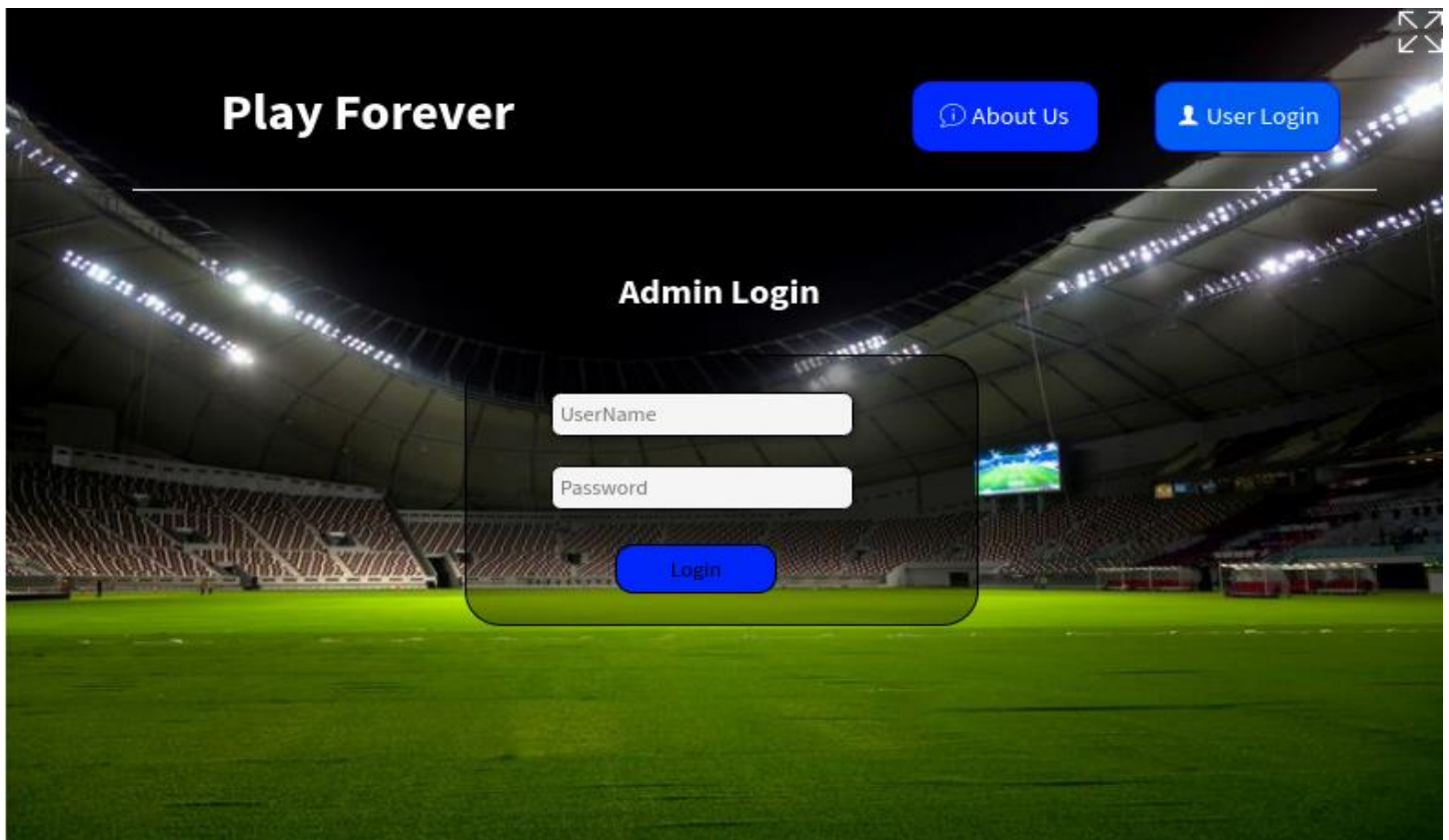
In this diagram we need to know how player can create a team. First, player request to create a team. Player window will return him all players in the database(ArrayList) that have the same location then player will choose the player that he want and send them then request to create the team then it will return to him that team is created.

Task 3 User Interface Design:

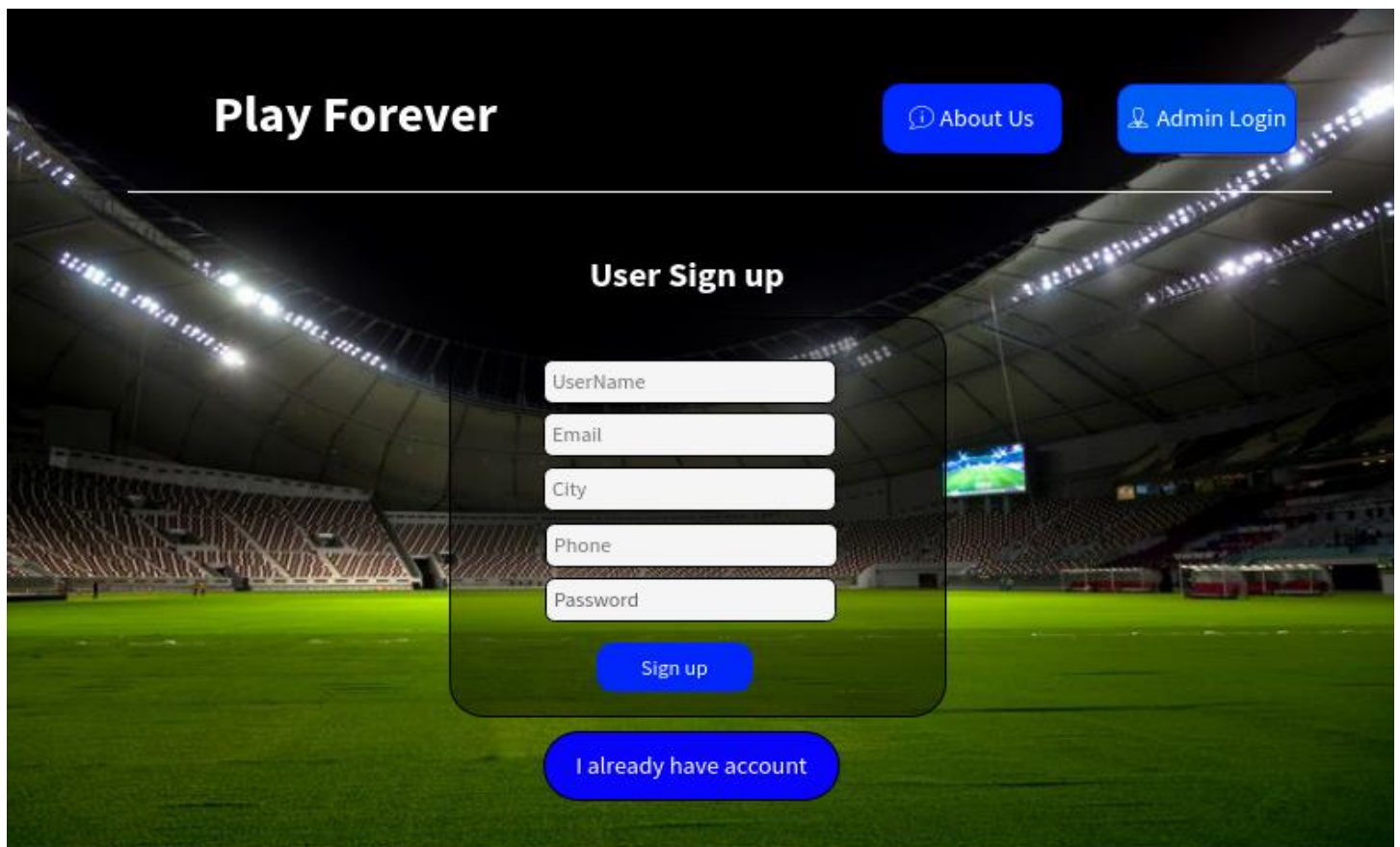
1- User Login:



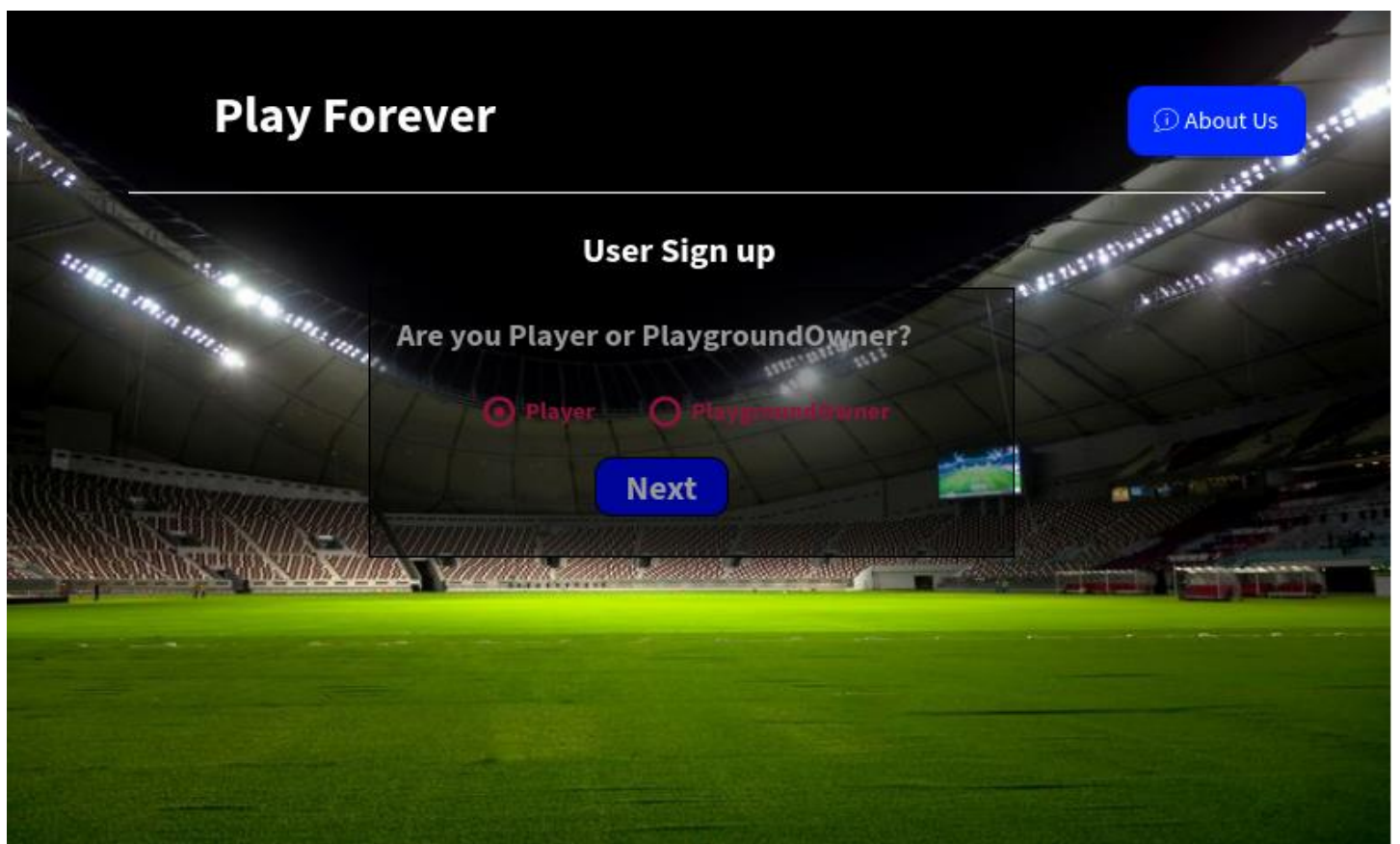
2- Admin login:



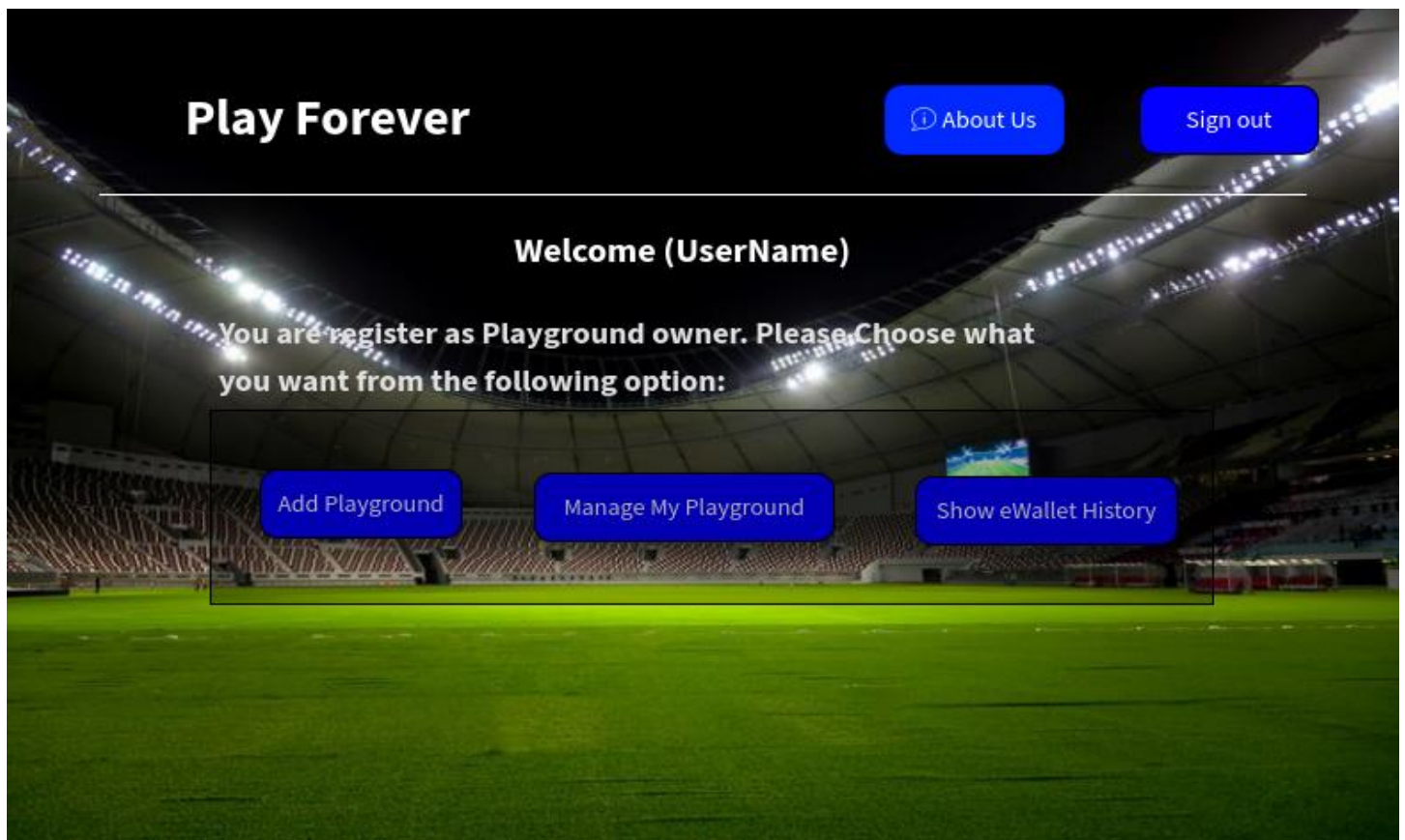
3- User Sign up:



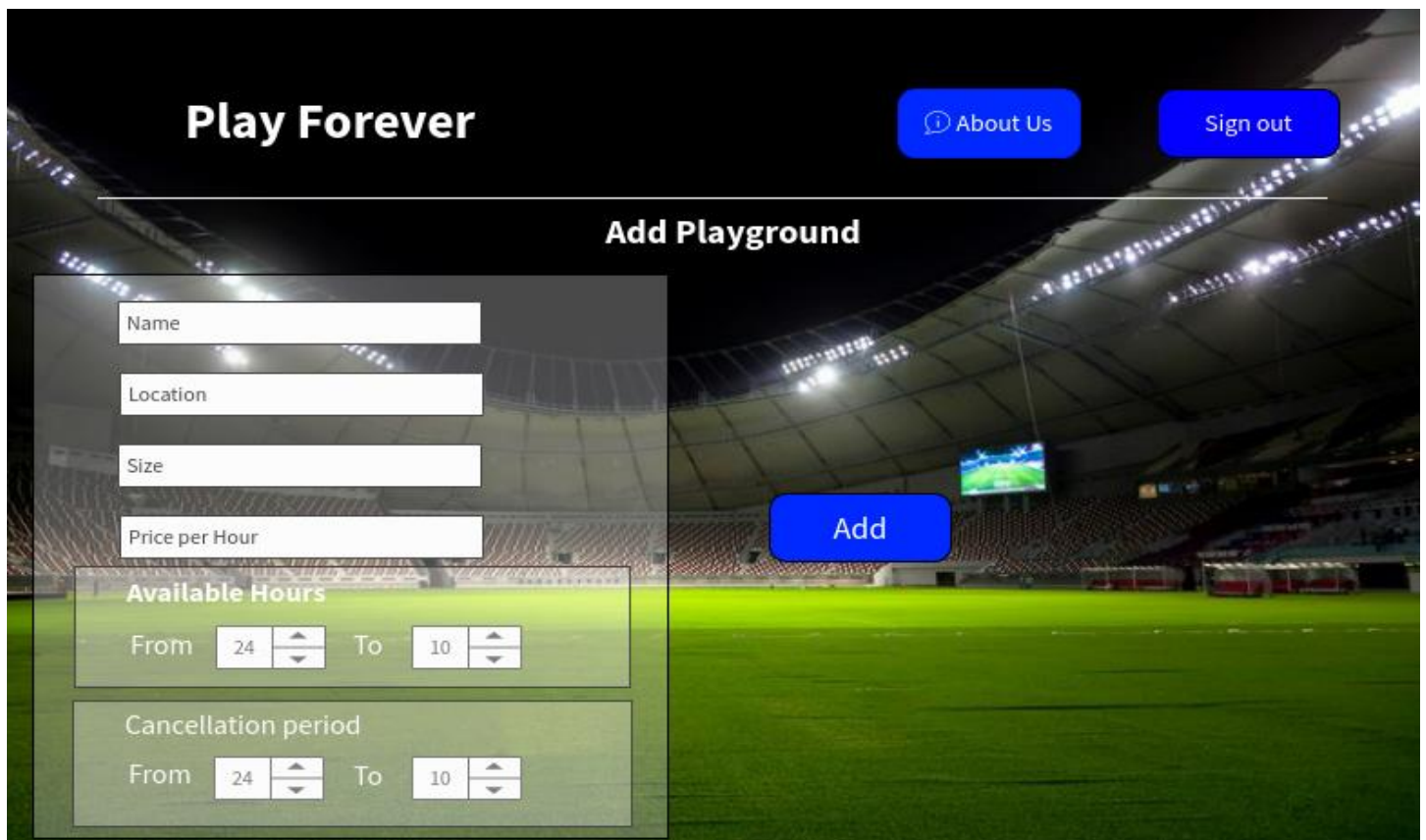
4- After click on sign up to know you are player or playground owner:



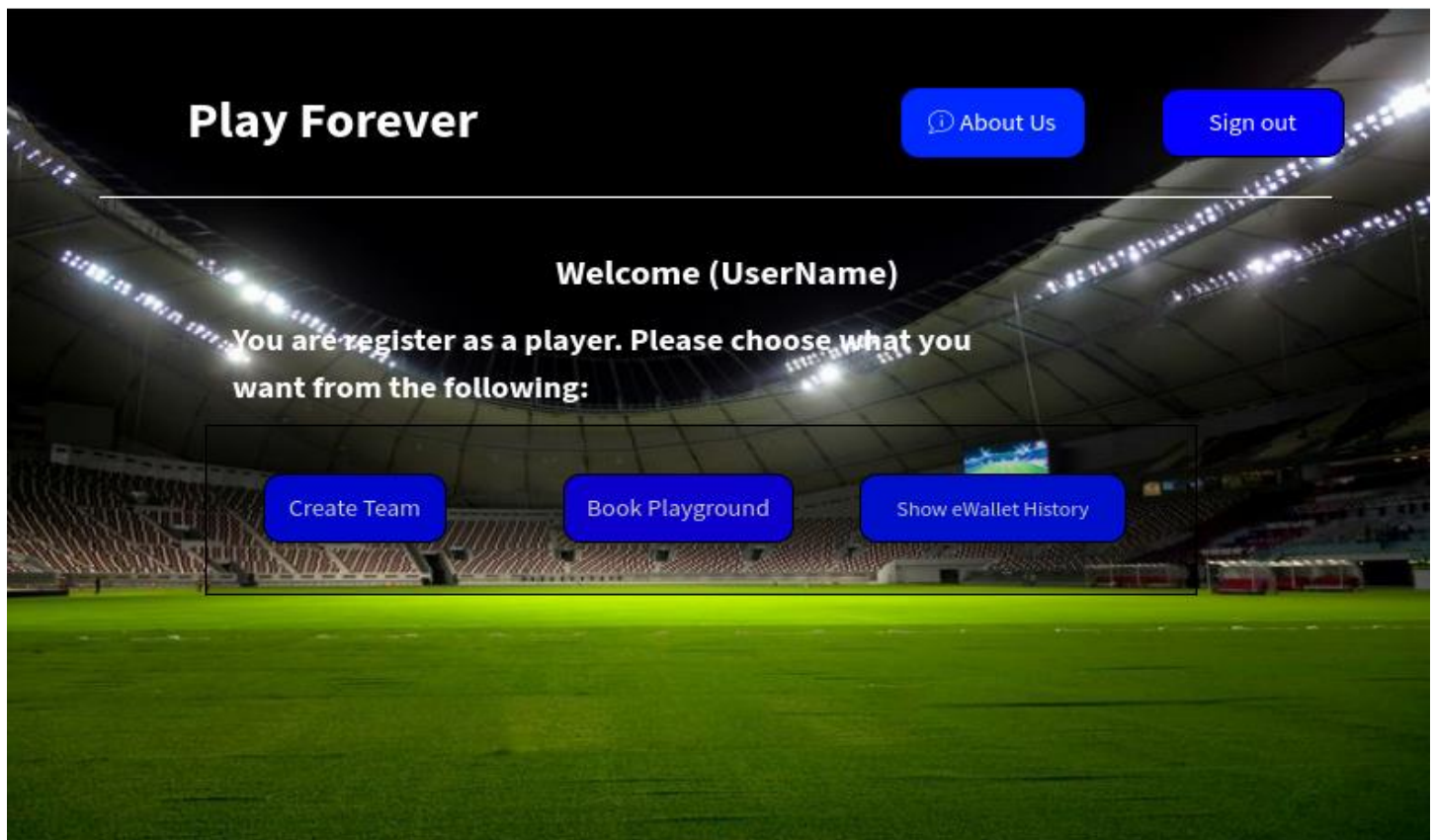
5- After register or login as playground Owner:



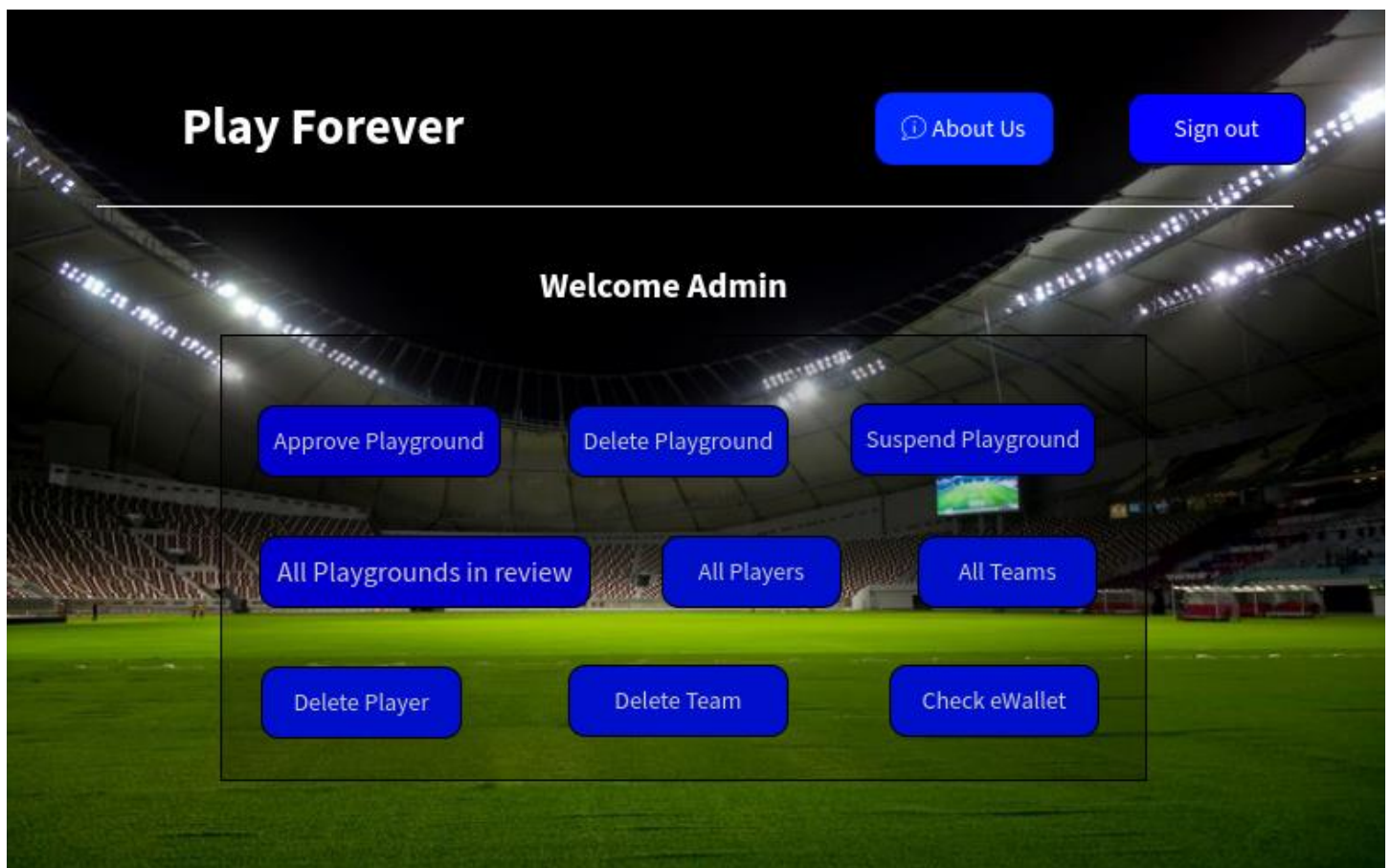
6- Add Playground:



7- After register or sign in as player:



8- After login as admin:



Task 2 : Development

Project Phase 3: Implementation

1- GOFO.java:

```
package gofo;
import java.util.ArrayList;
import java.util.Scanner;
import gofo.register;
import gofo.login;
import gofo.period;
import gofo.playground;
import gofo.DB;
import gofo.eWallet;

public class GOFO {
    static Scanner input = new Scanner(System.in);
    public static void main_menu(){
        System.out.println("\t*****");
        System.out.println("\tWelcome To Playgrounds' Application");
        System.out.println("\t*****");
        System.out.println("Please choose from the following:");
        System.out.println("*****");
        System.out.println("\t1-Register");
        System.out.println("\t2-Login");
    }
    public static void about_program(){
        System.out.println("This program used to help players to book playground that is near to them online");
    }
    public static void main(String[] args) {
        DB db = new DB();
        book_playground b = new book_playground();
        eWallet e = new eWallet("saeed" ,"0000" , 5000);
        register x = new register("saeed" , "saeed@gmail.com" , 0000 , "Cairo" , "1234" , "player" , e);
        db.store_register(x);
        eWallet e2 = new eWallet("ahmed" , "0000" , 10000 );
        register y = new register("ahmed" , "ahmed@gmail.com" , 1111 , "Giza" , "4321" ,
"playgroundOwner" , e2);
        db.store_register(y);
    }
}
```

```
playground play1 = new playground();
play1.set_name("aaaa");
play1.set_location("Cairo");
play1.set_size(1000);
play1.set_price_per_hour(100);
play1.set_available_hours(1,5);
play1.set_cancellation_period(2, 3);
db.active_playground(play1);
playground play2 = new playground();
play2.set_name("bbbb");
play2.set_location("Giza");
play2.set_size(1000);
play2.set_price_per_hour(120);
play2.set_available_hours(1,8);
play2.set_cancellation_period(2, 3);
db.active_playground(play2);
playground play3 = new playground();
play3.set_name("cccc");
play3.set_location("Cairo");
play3.set_size(1000);
play3.set_price_per_hour(140);
play3.set_available_hours(1,5);
play3.set_cancellation_period(2, 3);
db.active_playground(play3);
playground play4 = new playground();
play4.set_name("dddd");
play4.set_location("Giza");
play4.set_size(1000);
play4.set_price_per_hour(120);
play4.set_available_hours(1,5);
play4.set_cancellation_period(2, 3);
db.store_playground(play4);
main_menu();
int regiser_login = input.nextInt();
if(regiser_login == 1){
    register r = new register();
    System.out.print("Enter Name: ");
    String register_name = input.next();
    r.set_name(register_name);
    System.out.print("Enter Email: ");
```

```

String register_email = input.next();
r.set_email(register_email);
System.out.print("Enter Phone: ");
int register_phone = input.nextInt();
r.set_phone(register_phone);
System.out.print("Enter City: ");
String register_city = input.next();
r.set_city(register_city);
System.out.print("Enter Password: ");
String register_Password = input.next();
r.set_password(register_Password);
while (true){
    System.out.println("Register_Type: 1- player  2- playgroundOwner");
    int choose = input.nextInt();
    if(choose == 1){
        r.set_player_playgroundOwner("player");
        db.store_register(r);
        break;
    }
    else if(choose == 2){
        r.set_player_playgroundOwner("playgroundOwner");
        db.store_register(r);
        break;
    }
    else{
        System.out.println("Please choose 1 or 2");
    }
}
System.out.println("eWallet Information: ");
System.out.print("Enter eWallet name: ");
String eWallet_name = input.next();
System.out.print("Enter eWallet password: ");
String eWallet_password = input.next();
System.out.print("Enter eWallet balance: ");
double eWallet_balance = input.nextDouble();
eWallet ee = new eWallet(eWallet_name , eWallet_password , eWallet_balance);
r.add_eWallet(ee);
System.out.println("\t\tYou are registered at GOFO");
System.out.print("\t\tHello ");
System.out.println(r.get_name());

```



```

if(r.get_player_playgroundOwner() == "playgroundOwner"){
    while(true){
        System.out.println("Please choose: ");
        System.out.println("\t 1- add playground");
        System.out.println("\t 2- view eWallet information");
        System.out.println("\t 3- About this program");
        System.out.println("\t 4- Exit");
        int choose = input.nextInt();
        if(choose == 1){
            playground p = new playground();
            System.out.print("Enter name: ");
            String playground_name = input.next();
            p.set_name(playground_name);
            System.out.print("Enter location: ");
            String playground_location = input.next();
            p.set_location(playground_location);
            System.out.print("Enter size: ");
            double playground_size = input.nextDouble();
            p.set_size(playground_size);
            System.out.print("Enter price per hour: ");
            double playground_price = input.nextDouble();
            p.set_size(playground_price);
            System.out.println("Enter available hours: ");
            System.out.print("\tFrom: ");
            int from = input.nextInt();
            System.out.print("\tTo: ");
            int to = input.nextInt();
            p.set_available_hours(from , to);
            System.out.println("Enter cancellation period: ");
            System.out.print("\tFrom: ");
            int from1 = input.nextInt();
            System.out.print("\tTo: ");
            int to1 = input.nextInt();
            p.set_cancellation_period(from1, to1);
            db.store_playground(p);
            System.out.println("\tPlayground added and in review");
        }
        else if(choose == 2){
            System.out.println(ee.toString());
        }
    }
}

```

```

        else if(choose == 3){
            about_program();
        }
        else if(choose == 4){
            break;
        }
        else{
            System.out.println("Please choose 1 , 2 , 3 or 4");
        }
    }
}
else if(r.get_player_playgroundOwner() == "player"){
    while(true){
        System.out.println("Please choose: ");
        System.out.println("\t 1- Book playground");
        System.out.println("\t 2- view eWallet information");
        System.out.println("\t 3- About this program");
        System.out.println("\t 4- Exit");
        int choose = input.nextInt();
        int j = 0;
        if(choose == 1){
            System.out.println("Playgrounds available to you");
            db.show_playground_at_specific_location(r.get_city());
            System.out.print("Please write the playground that you want: ");
            String name = input.next();
            for(int i = 0 ; i < db.active_playgrounds.size() ; i++){
                if(db.active_playgrounds.get(i).get_name().equals(name)){
                    j = i;
                }
            }
            System.out.println("Period: ");
            System.out.print("\tFrom: ");
            int f = input.nextInt();
            System.out.print("\tTo: ");
            int t = input.nextInt();
            int c = t - f;
            System.out.print("price is : ");
            System.out.println(c * db.active_playgrounds.get(j).get_price_per_hour());
            System.out.println("Do you want to book this playground?");
            String yy = input.next();

```

```

        if(yy.equals("yes") || yy.equals("Yes")){
            b.book(db.active_playgrounds.get(j) , r);
            r.e.set_balance(r.e.get_balance() - (c *
db.active_playgrounds.get(j).get_price_per_hour()));
            if(r.e.get_balance() < 0){
                System.out.println("You haven't enough money");
            }
            else{
                db.store_book_playground(b);
                System.out.println("You are booked the playground");
            }
        }
    }
}
else if(choose == 2){
    System.out.println(ee.toString());
}
else if(choose == 3){
    about_program();
}
else if(choose == 4){
    break;
}
else{
    System.out.println("Please choose 1 , 2 , 3 or 4");
}
}
}
}
else if(regiser_login == 2){
    login l = new login();
    System.out.print("Enter Name: ");
    String login_name = input.next();
    l.set_Lname(login_name);
    System.out.print("Enter Password: ");
    String login_Password = input.next();
    l.set_Lpassword(login_Password);
    if(db.check(l) == -1){
        System.out.println("\tIncorrect username or password");
    }
    else if(db.check(l) == -2){

```

```

System.out.println("\t Hello Admin");
while(true){
    System.out.println("Please choose: ");
    System.out.println("\t 1- Approve playground");
    System.out.println("\t 2- Showing all active playground");
    System.out.println("\t 3- exit");
    int choose = input.nextInt();
    if(choose == 1){
        System.out.println("\t All playgrounds in review");
        db.print_all_playgrounds_in_review();
        System.out.println("Please enter the playground that you want to active it:");
        int rr = 0;
        String name = input.next();
        if(db.check_playground_name(name) == -2){
            System.out.println("the playground name not found");
        }
        else{
            db.active_playground(db.all_playgrounds_in_review.get(db.check_playground_name(name)));
            db.all_playgrounds_in_review.remove(db.check_playground_name(name));
            System.out.println("playground as approved");
        }
    }
    else if(choose == 2){
        db.print_active_playgrounds();
    }
    else if(choose == 3){
        break;
    }
    else {
        System.out.println("Please enter 1 or 2");
    }
}
else{
    System.out.println("\t You are logged in");
    if(db.all_registers.get(db.check(l)).get_player_playgroundOwner() == "player"){
        System.out.print("\t Hello ");
        System.out.println(db.all_registers.get(db.check(l)).get_name());
        while(true){

```

```

        System.out.println("Please choose: ");
        System.out.println("\t 1- Book playground");
        System.out.println("\t 2- view eWallet information");
        System.out.println("\t 3- About this program");
        System.out.println("\t 4- Exit");
        int choose = input.nextInt();
        int j = 0;
        if(choose == 1){
            System.out.println("Playgrounds available to you");

db.show_playground_at_specific_location(db.all_registers.get(db.check(l)).get_city());
            System.out.print("Please write the playground that you want: ");
            String name = input.next();
            for(int i = 0 ; i < db.active_playgrounds.size() ; i++){
                if(db.active_playgrounds.get(i).get_name().equals(name)){
                    j = i;
                }
            }
            System.out.println("Period: ");
            System.out.print("\tFrom: ");
            int f = input.nextInt();
            System.out.print("\tTo: ");
            int t = input.nextInt();
            int c = t - f;
            System.out.print("price is : ");
            System.out.println(c * db.active_playgrounds.get(j).get_price_per_hour());
            System.out.println("Do you want to book this playground?");
            String yy = input.next();
            if(yy.equals("yes") || yy.equals("Yes")){
                b.book(db.active_playgrounds.get(j) , db.all_registers.get(db.check(l)));

db.all_registers.get(db.check(l)).e.set_balance(db.all_registers.get(db.check(l)).e.get_balance() - (c
* db.active_playgrounds.get(j).get_price_per_hour()));
                if(db.all_registers.get(db.check(l)).e.get_balance() < 0){
                    System.out.println("You haven't enough money");
                }
                else{
                    db.store_book_playground(b);
                    System.out.println("You are booked the playground");
                }
            }
        }
    }
}

```

```

        }
    }
    else if(choose == 2){
        System.out.println(db.all_registers.get(db.check(l)).e.toString());
    }
    else if(choose == 3){
        about_program();
    }
    else if(choose == 4){
        break;
    }
    else{
        System.out.println("Please choose 1 , 2 , 3 or 4");
    }
}
}
else if(db.all_registers.get(db.check(l)).get_player_playgroundOwner() ==
"playgroundOwner"){
    while(true){
        System.out.println("Please choose: ");
        System.out.println("\t 1- add playground");
        System.out.println("\t 2- view eWallet information");
        System.out.println("\t 3- About this program");
        System.out.println("\t 4- Exit");
        int choose = input.nextInt();
        if(choose == 1){
            playground p = new playground();
            System.out.print("Enter name: ");
            String playground_name = input.next();
            p.set_name(playground_name);
            System.out.print("Enter location: ");
            String playground_location = input.next();
            p.set_location(playground_location);
            System.out.print("Enter size: ");
            double playground_size = input.nextDouble();
            p.set_size(playground_size);
            System.out.print("Enter price per hour: ");
            double playground_price = input.nextDouble();
            p.set_size(playground_price);
            System.out.println("Enter available hours: ");

```

```

        System.out.print("\tFrom: ");
        int from = input.nextInt();
        System.out.print("\tTo: ");
        int to = input.nextInt();
        p.set_available_hours(from , to);
        System.out.println("Enter cancellation period: ");
        System.out.print("\tFrom: ");
        int from1 = input.nextInt();
        System.out.print("\tTo: ");
        int to1 = input.nextInt();
        p.set_cancellation_period(from1, to1);
        db.store_playground(p);
        System.out.println("\tPlayground added and in review");
    }
    else if(choose == 2){
        System.out.println(db.all_registers.get(db.check(l)).e.toString());
    }
    else if(choose == 3){
        about_program();
    }
    else if(choose == 4){
        break;
    }
    else{
        System.out.println("Please choose 1 , 2 , 3 or 4");
    }
}
}
}
}
}
}
}
}
}
}

```

2- Register.java:

```
package gofo;
import gofo.eWallet;

public class register {
    private String name;
    private int id;
    private static int n = 1;
    private String email;
    private int phone;
    private String city;
    private String password;
    private String player_playgroundOwner;
    public eWallet e;
    register(){
        this.name = "";
        set_id();
        this.email = "";
        this.phone = 0;
        this.city = "";
        this.password = "";
        this.player_playgroundOwner = "";
    }
    register(String name , String email , int phone , String city , String password , String
player_playgroundOwner , eWallet e){
        this.name = name;
        set_id();
        this.email = email;
        this.phone = phone;
        this.city = city;
        this.password = password;
        this.player_playgroundOwner = player_playgroundOwner;
        this.e = e;
    }
    public void set_name(String name){
        this.name = name;
    }
    public String get_name(){
        return this.name;
    }
}
```



```
}  
public void set_id(){  
    this.id = this.n;  
    this.n +=1;  
}  
public int get_id(){  
    return this.id;  
}  
public void set_email(String email){  
    this.email = email;  
}  
public String get_email(){  
    return this.email;  
}  
public void set_phone(int phone){  
    this.phone = phone;  
}  
public int get_phone(){  
    return this.phone;  
}  
public void set_city(String city){  
    this.city = city;  
}  
public String get_city(){  
    return this.city;  
}  
public void set_password(String password){  
    this.password = password;  
}  
public String get_password(){  
    return this.password;  
}  
  
public void set_player_playgroundOwner(String player_playgroundOwner){  
    this.player_playgroundOwner = player_playgroundOwner;  
}  
public String get_player_playgroundOwner(){  
    return this.player_playgroundOwner;  
}  
public void add_eWallet(eWallet e){
```

```

        this.e = e;
    }
    public eWallet get_eWallet(){
        return this.e;
    }
    @Override
    public String toString(){
        return ("Name: " + this.name + "\n" + "id: " + this.id + "\n" + "Email: " + this.email + "\n"
            + "Phone: " + this.phone + "\n" + "City: " + this.city + "\n" + "password: " + this.password
            + "\n" + "type: " + this.player_playgroundOwner + "\n" + "eWallet Inforamtion: \n" +
this.e + "\n");
    }
}

```

3- login.java:

```

package gofo;

public class login{
    private String Lname;
    private String Lpassword;
    login(){}
    login(String Lname , String Lpassword){
        this.Lname = Lname;
        this.Lpassword = Lpassword;
    }
    public void set_Lname(String Lname){
        this.Lname = Lname;
    }
    public String get_Lname(){
        return this.Lname;
    }
    public void set_Lpassword(String Lpassword){
        this.Lpassword = Lpassword;
    }
    public String get_Lpassword(){
        return this.Lpassword;
    }
    @Override

```

```
public String toString(){
    return "Name: " + this.Lname + "\n" + "Password: " + this.Lpassword + "\n";
}
}
```

4- period.java:

```
package gofo;

public class period{
    private int from;
    private int to;
    period(){
        from = 0;
        to = 0;
    }
    period(int from , int to){
        this.from = from;
        this.to = to;
    }
    public void set_from(int from){
        this.from = from;
    }
    public int get_from(){
        return this.from;
    }
    public void set_to(int to){
        this.to = to;
    }
    public int get_to(){
        return this.to;
    }
    @Override
    public String toString(){
        return "From: " + this.from + "    " + "To: " + this.to;
    }
}
```

5- playground.java:

```
package gofo;

public class playground{
    String name;
    String location;
    double size;
    double price_per_hour;
    private period available_hours;
    private period cancellation_period;
    playground(){
        name = "";
        location = "";
        size = 0.0;
        price_per_hour = 0;
    }
    public void set_name(String name){
        this.name = name;
    }
    public String get_name(){
        return this.name;
    }
    public void set_location(String location){
        this.location = location;
    }
    public String get_location(){
        return this.location;
    }
    public void set_size(double size ){
        this.size = size;
    }
    public double get_size(){
        return this.size;
    }
    public void set_price_per_hour(double price_per_hour){
        this.price_per_hour = price_per_hour;
    }
    public double get_price_per_hour(){
        return this.price_per_hour;
    }
}
```

```

    }
    public void set_available_hours(int from , int to){
        this.available_hours = new period(from , to);
    }
    public period get_available_hours(){
        return this.available_hours;
    }
    public void set_cancellation_period(int from , int to){
        this.cancellation_period = new period(from , to);
    }
    public period get_cancellation_period(){
        return this.cancellation_period;
    }
    @Override
    public String toString(){
        return "name: " + this.name + "\n" + "Location: " + this.location + "\n" + "size: " +
            this.size + "\n" + "Price per hour: " + this.price_per_hour + "\n" + "Available hours: " +
            this.available_hours + "\n" + "Cancellation period: " + this.cancellation_period + "\n\n";
    }
}

```

6- book_playground.java:

```

package gofo;

public class book_playground {
    private playground p = new playground();
    private register r = new register();
    public void book(playground p , register r){
        this.p = p;
        this.r = r;
    }
}

```

7- DB.java:

```
package gofo;
import java.util.Scanner;
import java.util.ArrayList;

class DB{
    DB(){
        public ArrayList <register> all_registers = new ArrayList <>();
        public ArrayList <playground> all_playgrounds_in_review = new ArrayList <>();
        public ArrayList <playground> active_playgrounds = new ArrayList <>();
        public ArrayList <book_playground> playgrounds_booking = new ArrayList <>();
        public void store_register(register r){
            all_registers.add(r);
        }
        public void print_all_registers(){
            for(int i = 0 ; i < all_registers.size() ; i++){
                System.out.print(all_registers.get(i));
            }
        }
        public int check(login l){
            if(l.get_Lname().equals("admin") && l.get_Lpassword().equals("admin")){
                return -2;
            }
            else{
                for(int i = 0 ; i < all_registers.size() ; i++){
                    if(l.get_Lname().equals(all_registers.get(i).get_name()) &&
l.get_Lpassword().equals(all_registers.get(i).get_password())){
                        return i;
                    }
                }
            }
            return -1;
        }
        public void store_playground(playground p){
            all_playgrounds_in_review.add(p);
        }
        public void print_all_playgrounds_in_review(){
            for(int i = 0 ; i < all_playgrounds_in_review.size() ; i++){
                System.out.print(all_playgrounds_in_review.get(i));
            }
        }
    }
}
```

```

    }
}
public void active_playground(playground p){
    active_playgrounds.add(p);
}
public void print_active_playgrounds(){
    for(int i = 0 ; i < active_playgrounds.size() ; i++){
        System.out.print(active_playgrounds.get(i));
    }
}
public void show_playground_at_specific_location(String location){
    for(int i = 0 ; i < active_playgrounds.size() ; i++){
        if(active_playgrounds.get(i).get_location().equals(location)){
            System.out.println(active_playgrounds.get(i));
        }
    }
}
public void store_book_playground(book_playground b){
    playgrounds_booking.add(b);
}
public int check_playground_name(String name){
    for(int i = 0 ; i < all_playgrounds_in_review.size() ; i++){
        if(all_playgrounds_in_review.get(i).get_name().equals(name)){
            return i;
        }
    }
    return -2;
}
}

```

Task 3 – Hosting and Documentation

Project Phase 5: Hosting and Documentation

I upload GOFO project at github and this is the url:

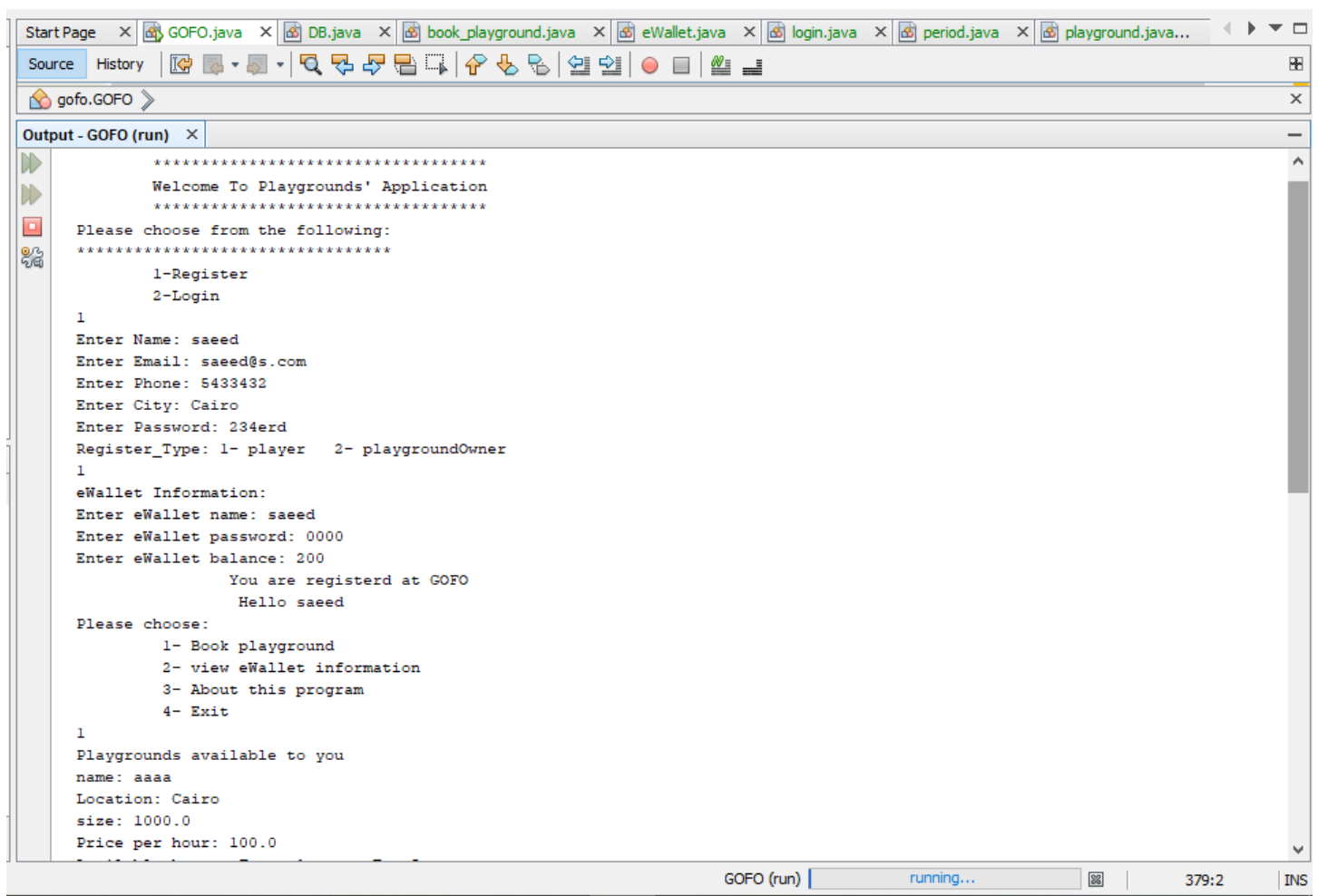
<https://github.com/SaeedGooda/GOFO>

I upload GOFO project at Google drive and this is the url:

<https://drive.google.com/drive/folders/1rxsp3QjpgQKd18SyuToZizp8crGmjt5W>

Task 4 – Screenshots and Video

1- This is a screenshots of register as player and view all playgrounds that exist in the city then add playground and see it after booking:



```
Start Page x GOFO.java x DB.java x book_playground.java x eWallet.java x login.java x period.java x playground.java...
Source History
gofo.GOFO
Output - GOFO (run) x

*****
Welcome To Playgrounds' Application
*****
Please choose from the following:
*****
1-Register
2-Login
1
Enter Name: saeed
Enter Email: saeed@s.com
Enter Phone: 5433432
Enter City: Cairo
Enter Password: 234erd
Register_Type: 1- player 2- playgroundOwner
1
eWallet Information:
Enter eWallet name: saeed
Enter eWallet password: 0000
Enter eWallet balance: 200
You are registerd at GOFO
Hello saeed
Please choose:
1- Book playground
2- view eWallet information
3- About this program
4- Exit
1
Playgrounds available to you
name: aaaa
Location: Cairo
size: 1000.0
Price per hour: 100.0

GOFO (run) running... 379:2 INS
```


Start Page x GOFO.java x DB.java x book_playground.java x eWallet.java x login.java x period.java x playground.java...

Source History

gofo.GOFO

Output - GOFO (run)

```
Playgrounds available to you
name: aaaa
Location: Cairo
size: 1000.0
Price per hour: 100.0
Available hours: From: 1      To: 5
Cancellation period: From: 2    To: 3

name: cccc
Location: Cairo
size: 1000.0
Price per hour: 140.0
Available hours: From: 1      To: 5
Cancellation period: From: 2    To: 3

Please write the playground that you want: aaaa
Period:
    From: 2
    To: 3
price is : 100.0
Do you want to book this playground?
yes
You are booked the playground
Please choose:
    1- Book playground
    2- view eWallet information
    3- About this program
    4- Exit
2
name: saeed
```

GOFO (run) | running... | 379:2 | INS

Start Page x GOFO.java x DB.java x book_playground.java x eWallet.java x login.java x period.java x playground.java...

Source History

gofo.GOFO

Output - GOFO (run)

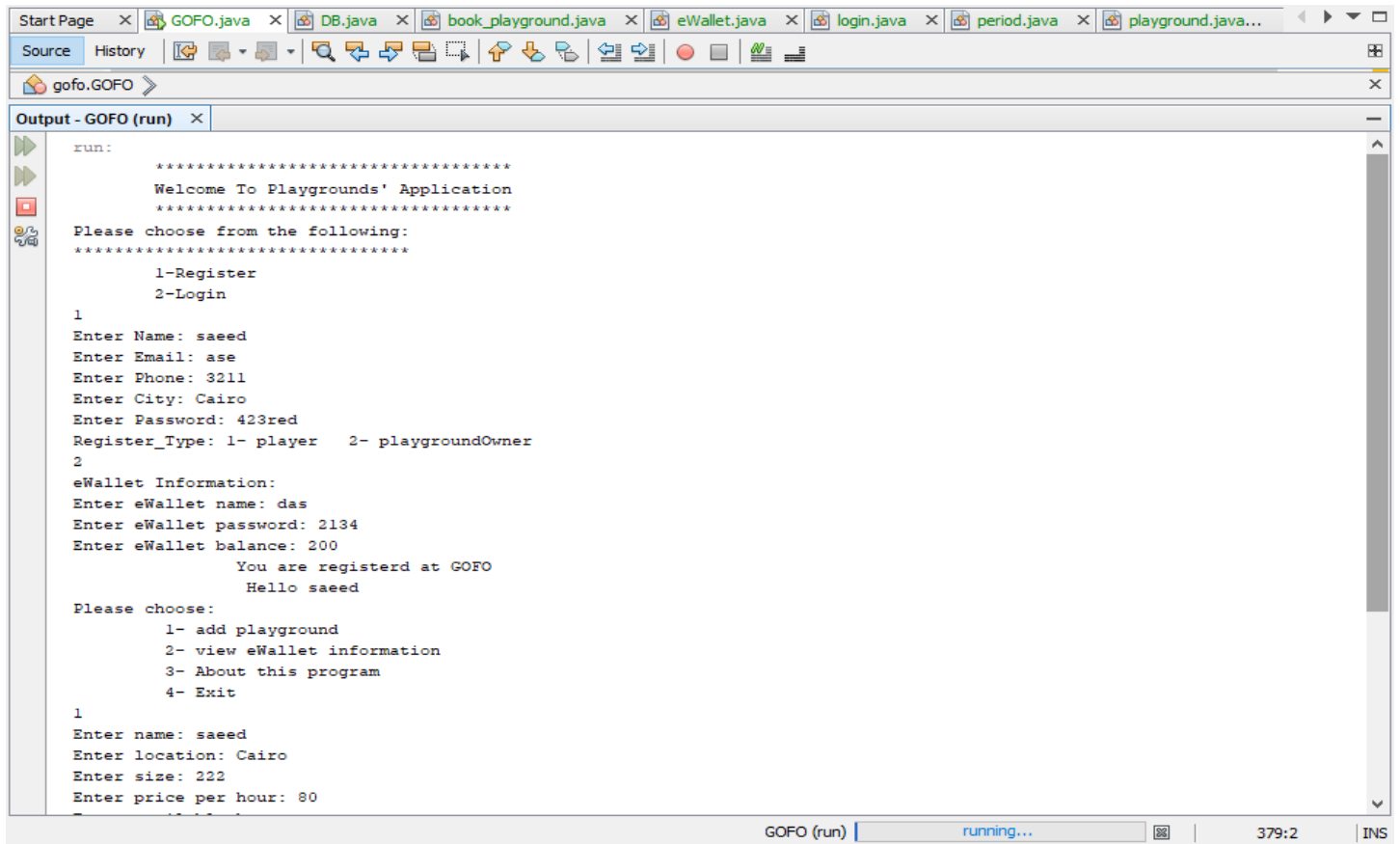
```
name: cccc
Location: Cairo
size: 1000.0
Price per hour: 140.0
Available hours: From: 1      To: 5
Cancellation period: From: 2    To: 3

Please write the playground that you want: aaaa
Period:
    From: 2
    To: 3
price is : 100.0
Do you want to book this playground?
yes
You are booked the playground
Please choose:
    1- Book playground
    2- view eWallet information
    3- About this program
    4- Exit
2
name: saeed
password: 0000
balance: 100.0

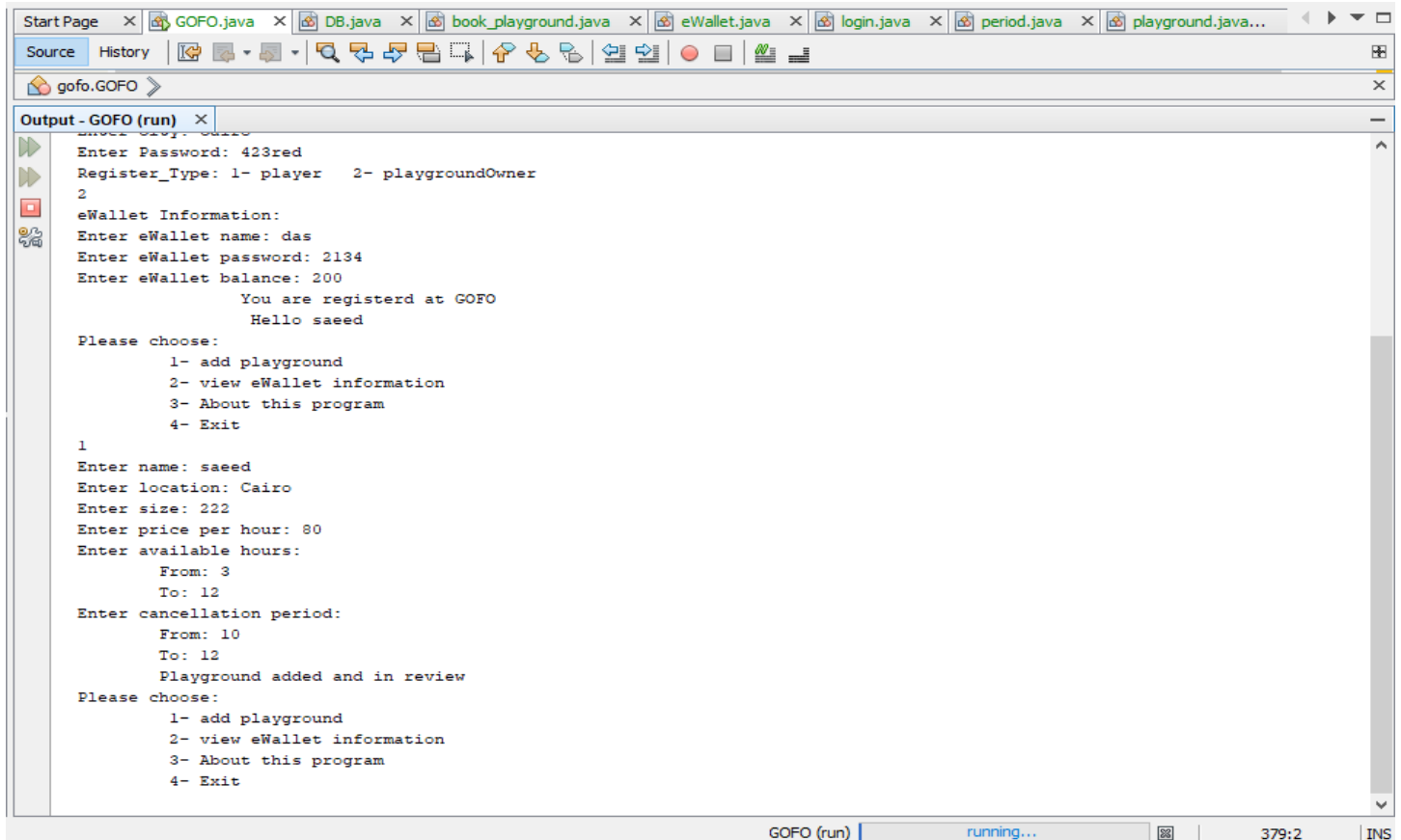
Please choose:
    1- Book playground
    2- view eWallet information
    3- About this program
    4- Exit
```

GOFO (run) | running... | 379:2 | INS

2- This is a screenshots of register as playground owner and add playground and tell him that playground is added and in review:



```
StartPage x GOFO.java x DB.java x book_playground.java x eWallet.java x login.java x period.java x playground.java...
Source History
gofo.GOFO
Output - GOFO (run) x
run:
*****
Welcome To Playgrounds' Application
*****
Please choose from the following:
*****
1-Register
2-Login
1
Enter Name: saeed
Enter Email: ase
Enter Phone: 3211
Enter City: Cairo
Enter Password: 423red
Register_Type: 1- player 2- playgroundOwner
2
eWallet Information:
Enter eWallet name: das
Enter eWallet password: 2134
Enter eWallet balance: 200
You are registerd at GOFO
Hello saeed
Please choose:
1- add playground
2- view eWallet information
3- About this program
4- Exit
1
Enter name: saeed
Enter location: Cairo
Enter size: 222
Enter price per hour: 80
GOFO (run) | running... | 379:2 | INS
```



```
StartPage x GOFO.java x DB.java x book_playground.java x eWallet.java x login.java x period.java x playground.java...
Source History
gofo.GOFO
Output - GOFO (run) x
Enter Password: 423red
Register_Type: 1- player 2- playgroundOwner
2
eWallet Information:
Enter eWallet name: das
Enter eWallet password: 2134
Enter eWallet balance: 200
You are registerd at GOFO
Hello saeed
Please choose:
1- add playground
2- view eWallet information
3- About this program
4- Exit
1
Enter name: saeed
Enter location: Cairo
Enter size: 222
Enter price per hour: 80
Enter available hours:
From: 3
To: 12
Enter cancellation period:
From: 10
To: 12
Playground added and in review
Please choose:
1- add playground
2- view eWallet information
3- About this program
4- Exit
GOFO (run) | running... | 379:2 | INS
```

3- Sign in as admin , approve playground and show all active playgrounds note that(admin name and password is "admin"):

```
run:
*****
Welcome To Playgrounds' Application
*****
Please choose from the following:
*****
1-Register
2-Login

2
Enter Name: admin
Enter Password: admin
Hello Admin
Please choose:
1- Approve playground
2- Showing all active playground
3- exit

1
All playgrounds in review
name: dddd
Location: Giza
size: 1000.0
Price per hour: 120.0
Available hours: From: 1      To: 5
Cancellation period: From: 2   To: 3

Please enter the playground that you want to active it:
dddd
playground as approved
Please choose:
1- Approve playground
2- Showing all active playground
3- exit
```

```
2
name: aaaa
Location: Cairo
size: 1000.0
Price per hour: 100.0
Available hours: From: 1      To: 5
Cancellation period: From: 2   To: 3

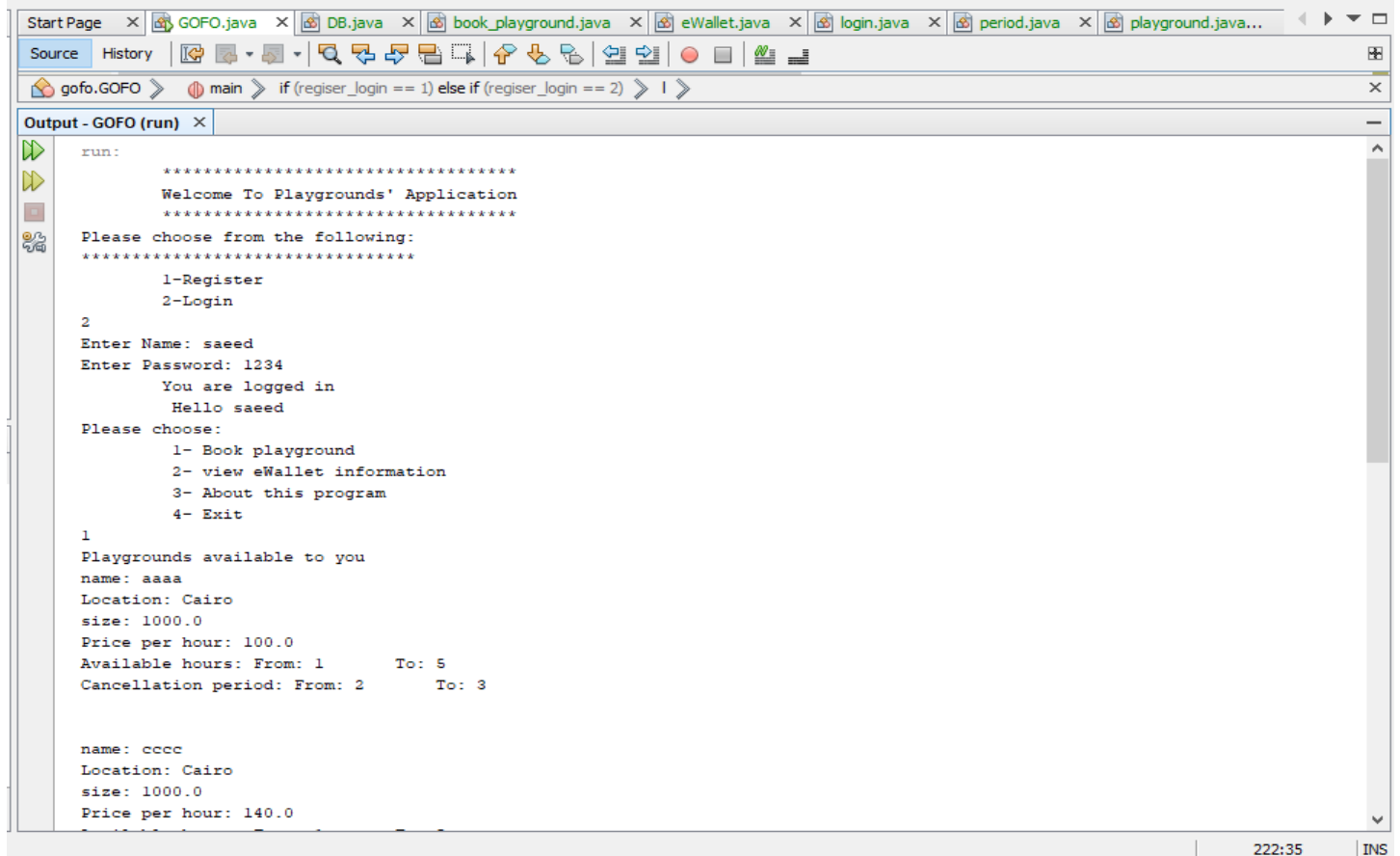
name: bbbb
Location: Giza
size: 1000.0
Price per hour: 120.0
Available hours: From: 1      To: 8
Cancellation period: From: 2   To: 3

name: cccc
Location: Cairo
size: 1000.0
Price per hour: 140.0
Available hours: From: 1      To: 5
Cancellation period: From: 2   To: 3

name: dddd
Location: Giza
size: 1000.0
Price per hour: 120.0
Available hours: From: 1      To: 5
Cancellation period: From: 2   To: 3

Please choose:
1- Approve playground
2- Showing all active playground
```

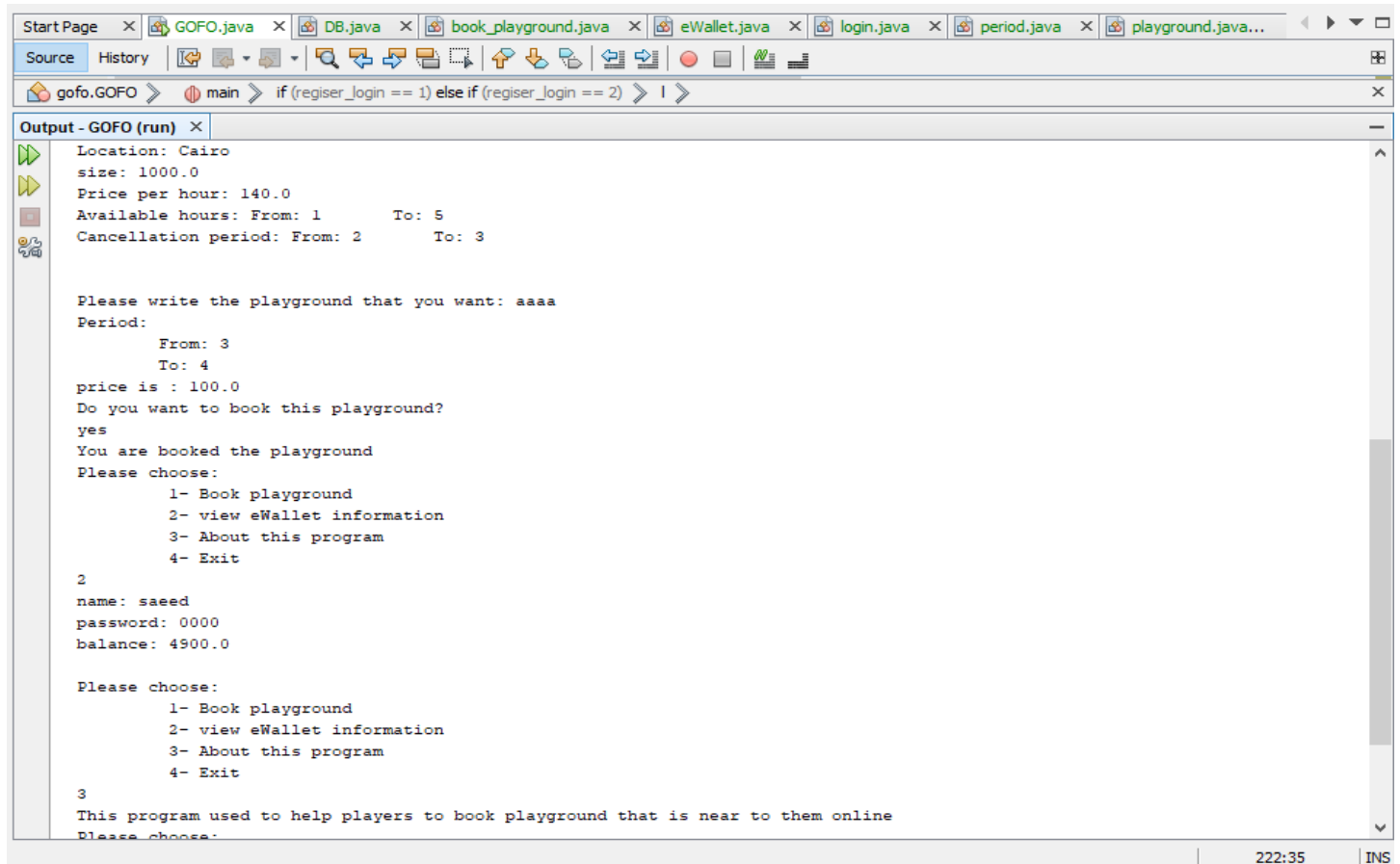
4- Player login and book playgrpund:



```
Start Page x GOFO.java x DB.java x book_playground.java x eWallet.java x login.java x period.java x playground.java...
Source History
gofo.GOFO > main > if (regiser_login == 1) else if (regiser_login == 2) > | >
Output - GOFO (run) x
run:
*****
Welcome To Playgrounds' Application
*****
Please choose from the following:
*****
1-Register
2-Login
2
Enter Name: saeed
Enter Password: 1234
You are logged in
Hello saeed
Please choose:
1- Book playground
2- view eWallet information
3- About this program
4- Exit
1
Playgrounds available to you
name: aaaa
Location: Cairo
size: 1000.0
Price per hour: 100.0
Available hours: From: 1 To: 5
Cancellation period: From: 2 To: 3

name: cccc
Location: Cairo
size: 1000.0
Price per hour: 140.0
```

222:35 | INS



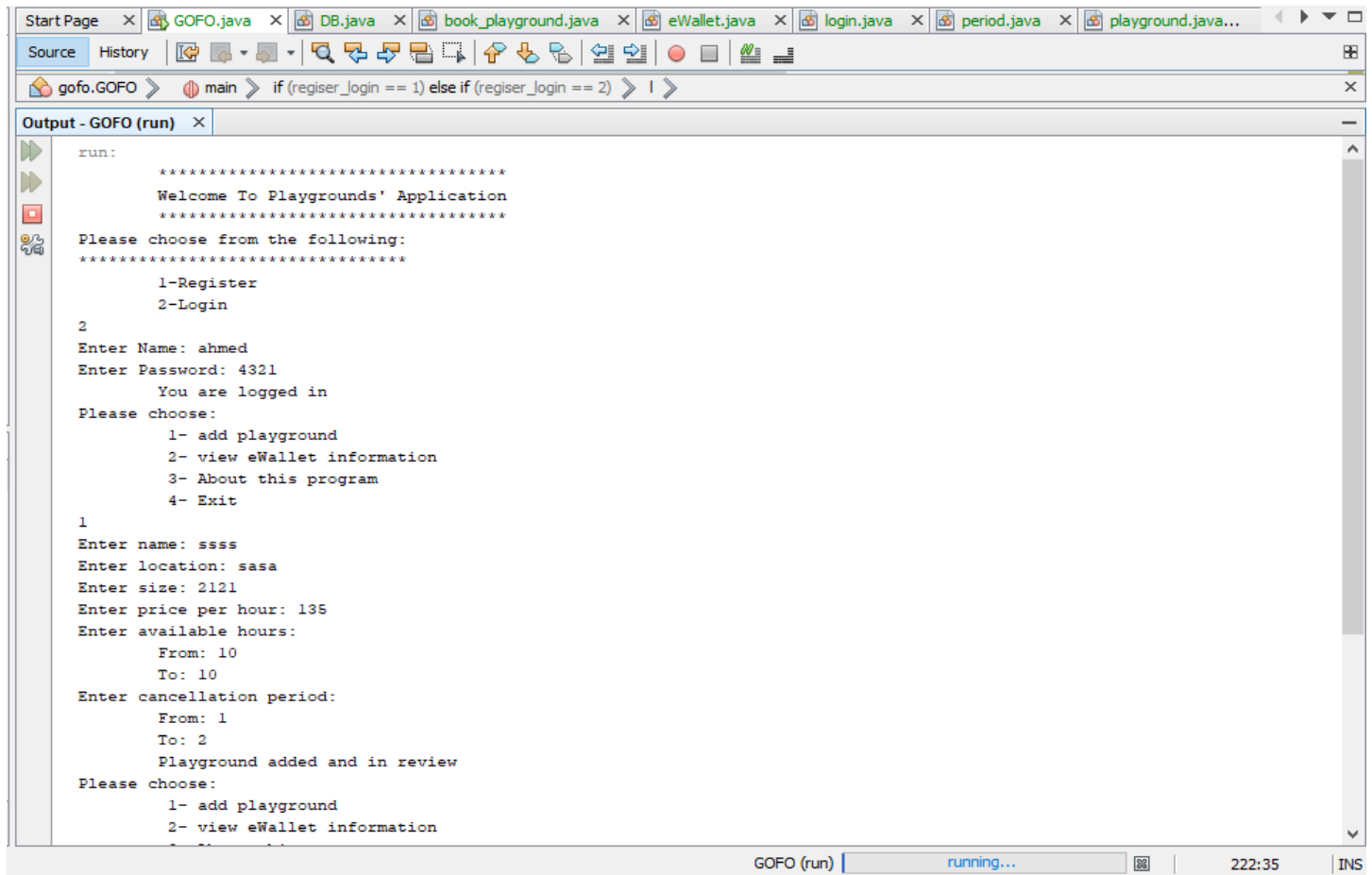
```
Start Page x GOFO.java x DB.java x book_playground.java x eWallet.java x login.java x period.java x playground.java...
Source History
gofo.GOFO > main > if (regiser_login == 1) else if (regiser_login == 2) > | >
Output - GOFO (run) x
Location: Cairo
size: 1000.0
Price per hour: 140.0
Available hours: From: 1 To: 5
Cancellation period: From: 2 To: 3

Please write the playground that you want: aaaa
Period:
From: 3
To: 4
price is : 100.0
Do you want to book this playground?
yes
You are booked the playground
Please choose:
1- Book playground
2- view eWallet information
3- About this program
4- Exit
2
name: saeed
password: 0000
balance: 4900.0

Please choose:
1- Book playground
2- view eWallet information
3- About this program
4- Exit
3
This program used to help players to book playground that is near to them online
Please choose:
```

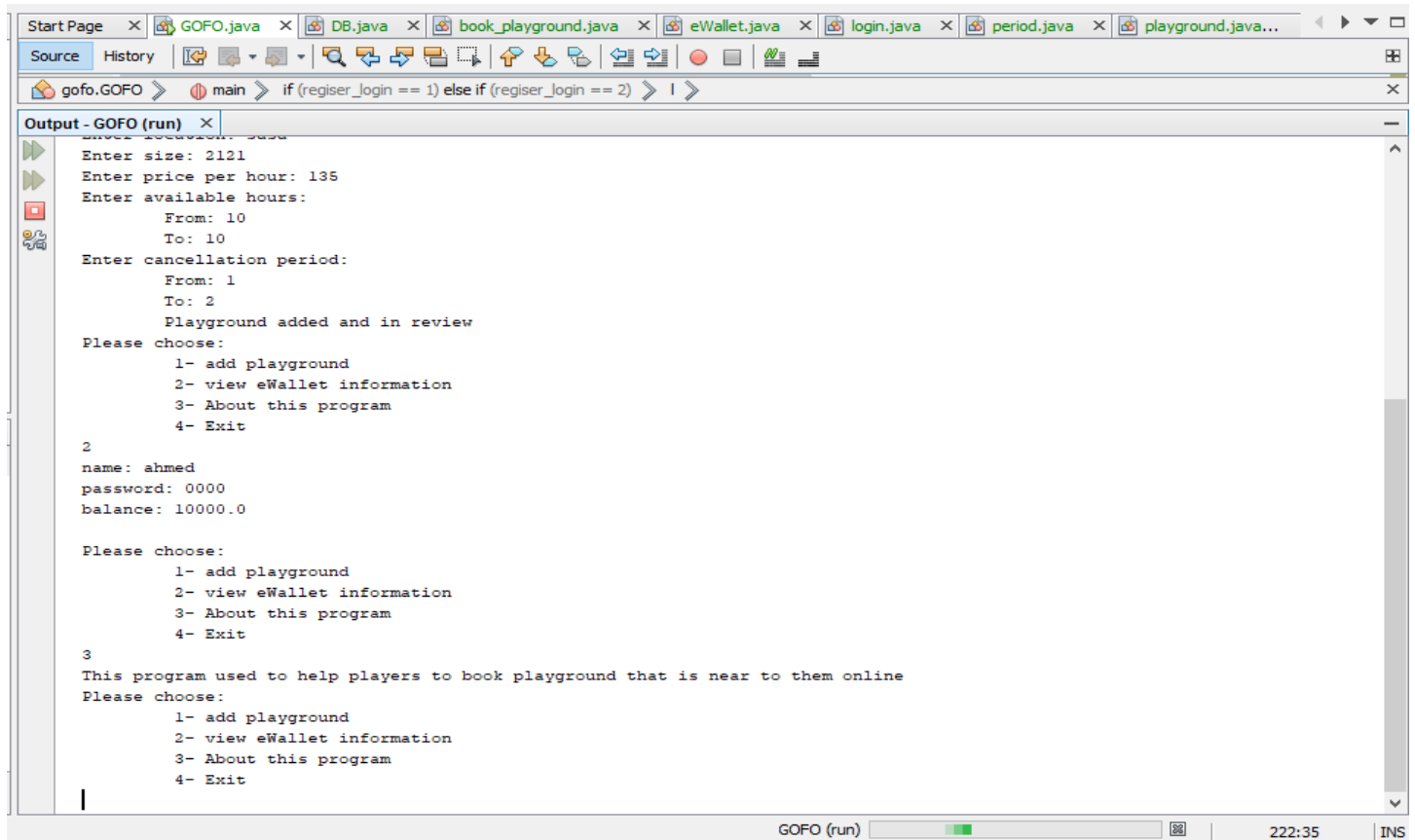
222:35 | INS

5- Playground owner login and add playground:



```
Start Page x GOFO.java x DB.java x book_playground.java x eWallet.java x login.java x period.java x playground.java...
Source History
gofo.GOFO > main > if (regiser_login == 1) else if (regiser_login == 2) > | >
Output - GOFO (run) x
run:
*****
Welcome To Playgrounds' Application
*****
Please choose from the following:
*****
1-Register
2-Login
2
Enter Name: ahmed
Enter Password: 4321
You are logged in
Please choose:
1- add playground
2- view eWallet information
3- About this program
4- Exit
1
Enter name: ssss
Enter location: sasa
Enter size: 2121
Enter price per hour: 135
Enter available hours:
From: 10
To: 10
Enter cancellation period:
From: 1
To: 2
Playground added and in review
Please choose:
1- add playground
2- view eWallet information
```

GOFO (run) running... 222:35 INS



```
Enter location: sasa
Enter size: 2121
Enter price per hour: 135
Enter available hours:
From: 10
To: 10
Enter cancellation period:
From: 1
To: 2
Playground added and in review
Please choose:
1- add playground
2- view eWallet information
3- About this program
4- Exit
2
name: ahmed
password: 0000
balance: 10000.0
Please choose:
1- add playground
2- view eWallet information
3- About this program
4- Exit
3
This program used to help players to book playground that is near to them online
Please choose:
1- add playground
2- view eWallet information
3- About this program
4- Exit
```

GOFO (run) 222:35 INS

Video link:

<https://www.mediafire.com/file/gflaiw1vo43pmbe/GOFO.webm/file>