

## Project about

### PlayStation App

N	Student Name
1	سعد عبد الله إبراهيم السهلي
2	محمد معجب محمد الدوسري
3	ريان محمد عبيد الشريف
4	ناصر حمد ناصر آل ذبيان

Supervised by : د. محمد سعد محمد العسيري

Year: 2023

---

## *INDEX*

---

<b>Content</b>	<b>Page</b>
Feasibility Study &Project Proposal	<b>3</b>
Project requirements	<b>5</b>
Activity diagram	<b>6</b>
Project Use Case Modelling	<b>8</b>

Sequence Diagrams	<b>12</b>
Class Diagram	<b>14</b>

## Introduction

It is a software application for iOS and Android devices that allows us to identify connected friends or view progress or compare awards with friends or create a personal profile to display to friends. The first version of the application was released on November 15 / 2013.

## Problems

- 1-You can't download games without Turn on a PS console
- 2-You can't message your friends without Turn on a PS console
- 3-Miss the latest PlayStation news without Turn on a PS console
- 4-You cannot play/delete your PS games without Turn on a PS console

## Background

Sony Corporation is a Japanese electronics company, founded by Masaru Buha in 1946, under the name "Tokyo Communications Engineering Corporation". It is also one of the world's largest manufacturers of consumer electronic products

## Proposed solution

- 1-Download games to your PS console remotely

2- Message your friends and coordinate group gaming sessions

3- Keep up with the latest PlayStation news

4- Launch/ Delete Your PS Games Remotely

## **Work Plan**

In the PS App application, we will use the Incremental model of application development where the application is developed in successive stages

### **1 – Determine requirements**

Application requirements are determined by collecting information from players and developers

### **2 – Design**

The application is designed according to the requirements specified in the first stage

### **3 – Implementation**

Design is performed like writing code or testing an application

### **4 – validation**

It is verified that the application meets the requirements

### **5 – Evolution**

The necessary modification are being made to the application based on feedback from players and developers

The application targets two categories of society :

**1 – Players:** the application allows players to purchase games or communicate with players

**2 - Developers:** the application provides a set of tools and resources that help them display games or announce upcoming updates

## **Project requirements**

### **Functional requirements:**

- 1- The system must allow the player to log in using their (email and password).
- 2- The system must send a verification message to verify the player's identity.
- 3- The system must allow the player to use a unique name that has not been used before.
- 4- The system can allow the player to view his downloaded games.
- 5- The system can allow the player to delete his games.
- 6- The system can allow the player to add new games.
- 7- The system can allow a player to create a message to others.
- 8- The system must allow the player to add another player.
- 9- The system's ability to send announcements about events occurring at a specific time.

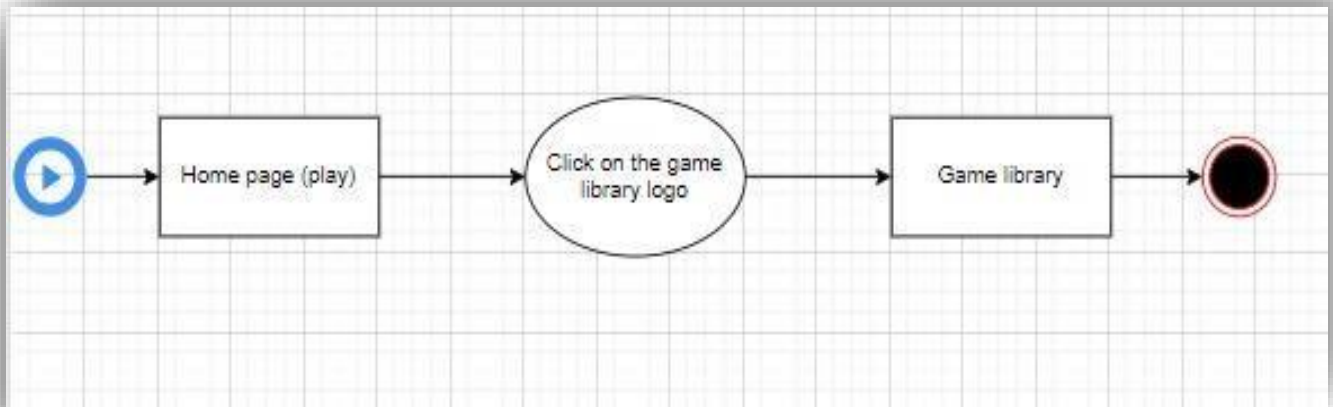
- 10- The system must allow the game developer to Download game in store.
- 11- The system must allow the game developer to delete game in store.
- 12- The system allows the game developer to modify (the shape, cover, color, and settings) of the game.
- 13-The system allows the player's close friends to view his real name.
- 14-The system allows the player to rate games
- The system allows the developer to see how players rate his product.
- 15-The system allows the developer to download a trial version of the game.
- 16-The system allows players to download trial games.
- 17-The system allows players to charge the digital wallet balance from the application.

### **Non-functional requirements:**

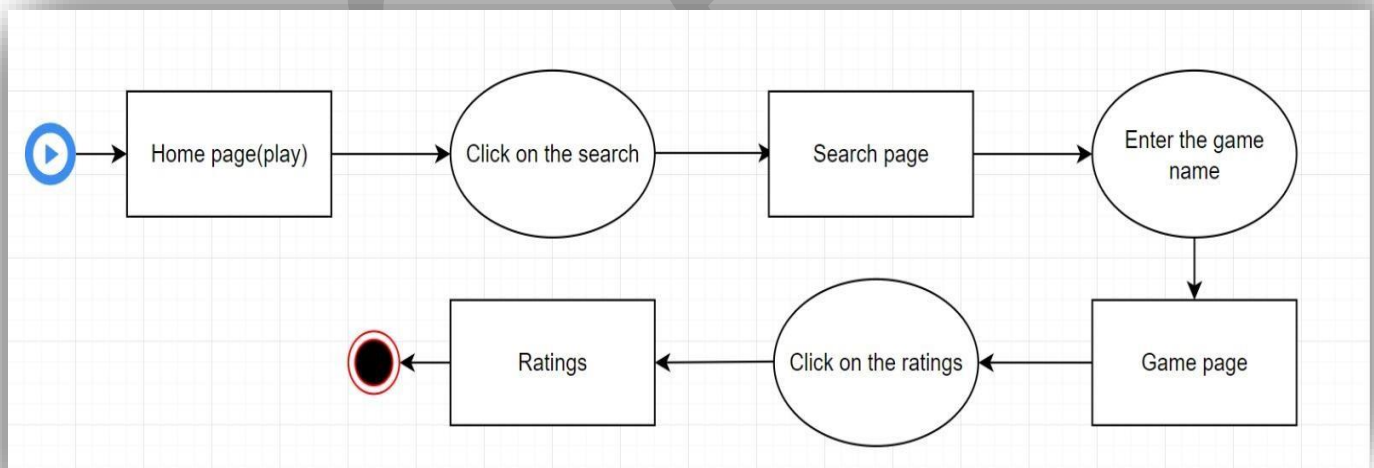
- 1- The running time of the application should not exceed four minutes.
- 2- The system must enable the player to perform several tasks without reducing performance and efficiency.
- 3- The possibility of easy navigation and use for users.
- 4- Every user account must be secure and not accessible to unauthorized persons.

### **Activity diagram:**

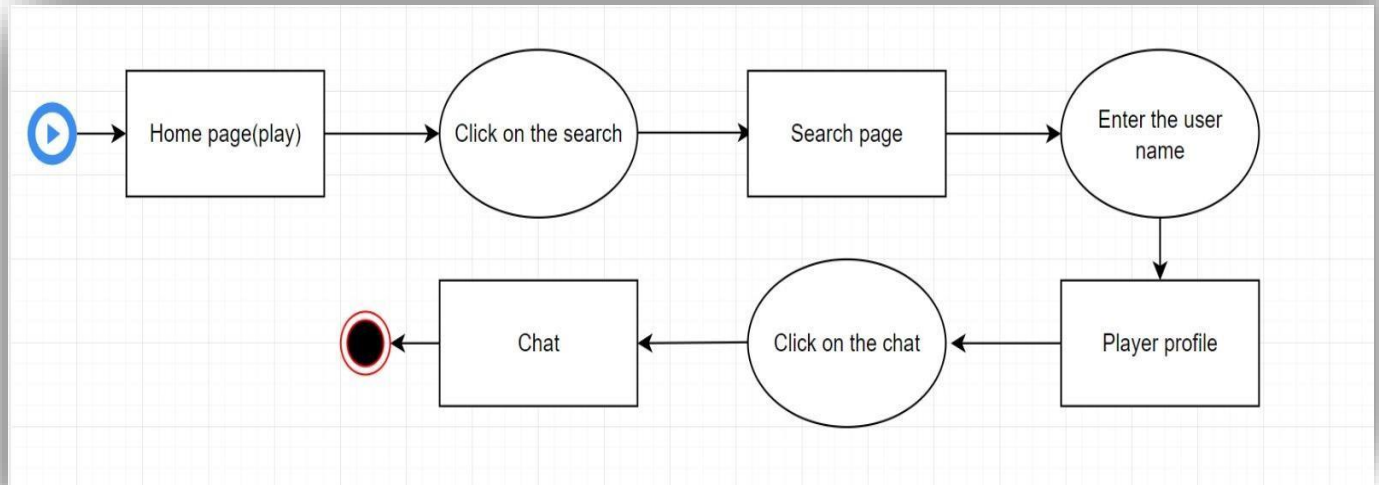
(View games)



) Review player ratings)



(Communicate with users)



## Project Use Case Modelling:

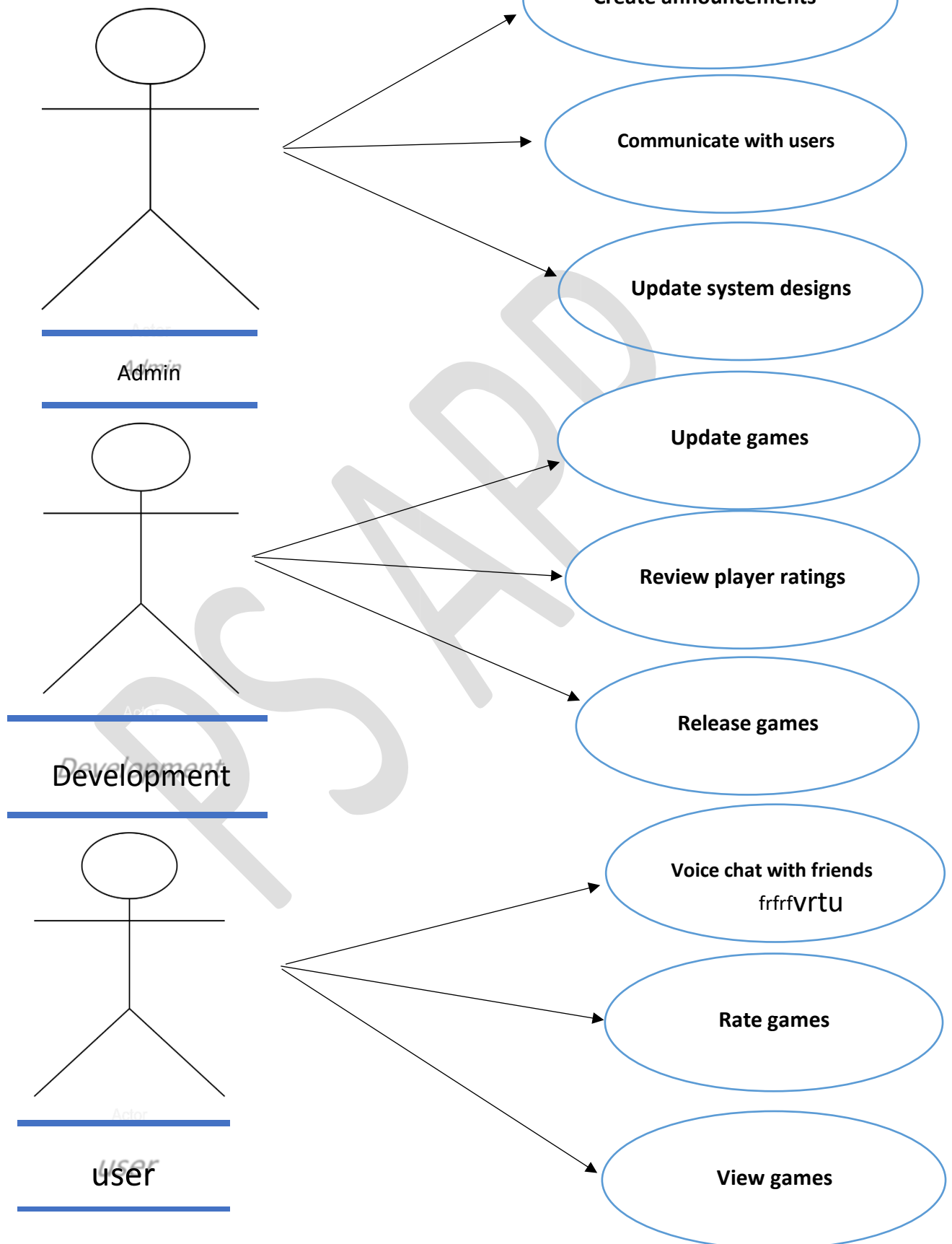
**Actor:**

Actors	Roles
<b>Admin</b>	<ul style="list-style-type: none"> <li>- Create announcements</li> <li>- Communicate with users</li> <li>- Update system designs</li> </ul>

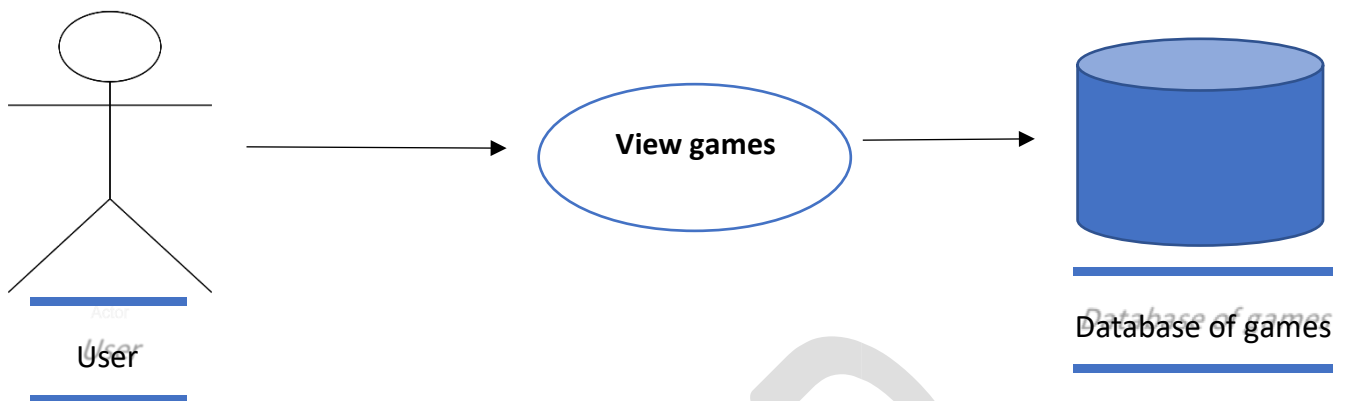


<b>User</b>	<ul style="list-style-type: none"><li>- View games</li><li>- Rate games</li><li>- Voice chat with friends</li></ul>
<b>Development</b>	<ul style="list-style-type: none"><li>- Review player ratings</li><li>- Release games</li><li>- Update games</li></ul>

## Use cases & its related use cases:

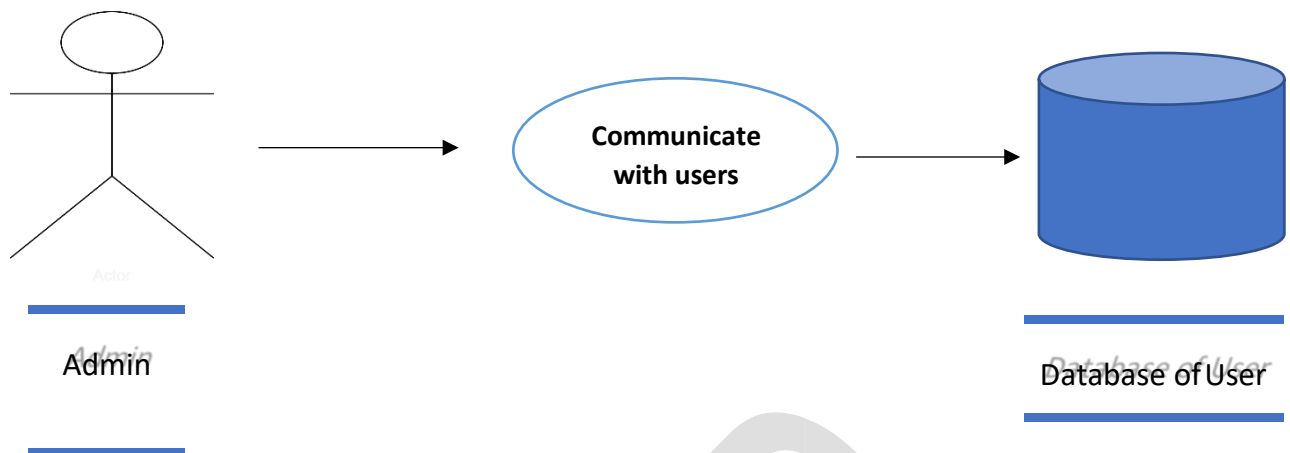


**Table 1:**



User : View Games	
<b>Actors</b>	User and DB of games
<b>Description</b>	It explains the interaction of the user (the player) and the way he views the games he has and those in the database designated for me to store games.
<b>Data</b>	User email, User password, Game code, And Verification code
<b>Stimulus</b>	User Want to view the games.
<b>Response</b>	Games information (Name, code)
<b>Comments</b>	No Comments.

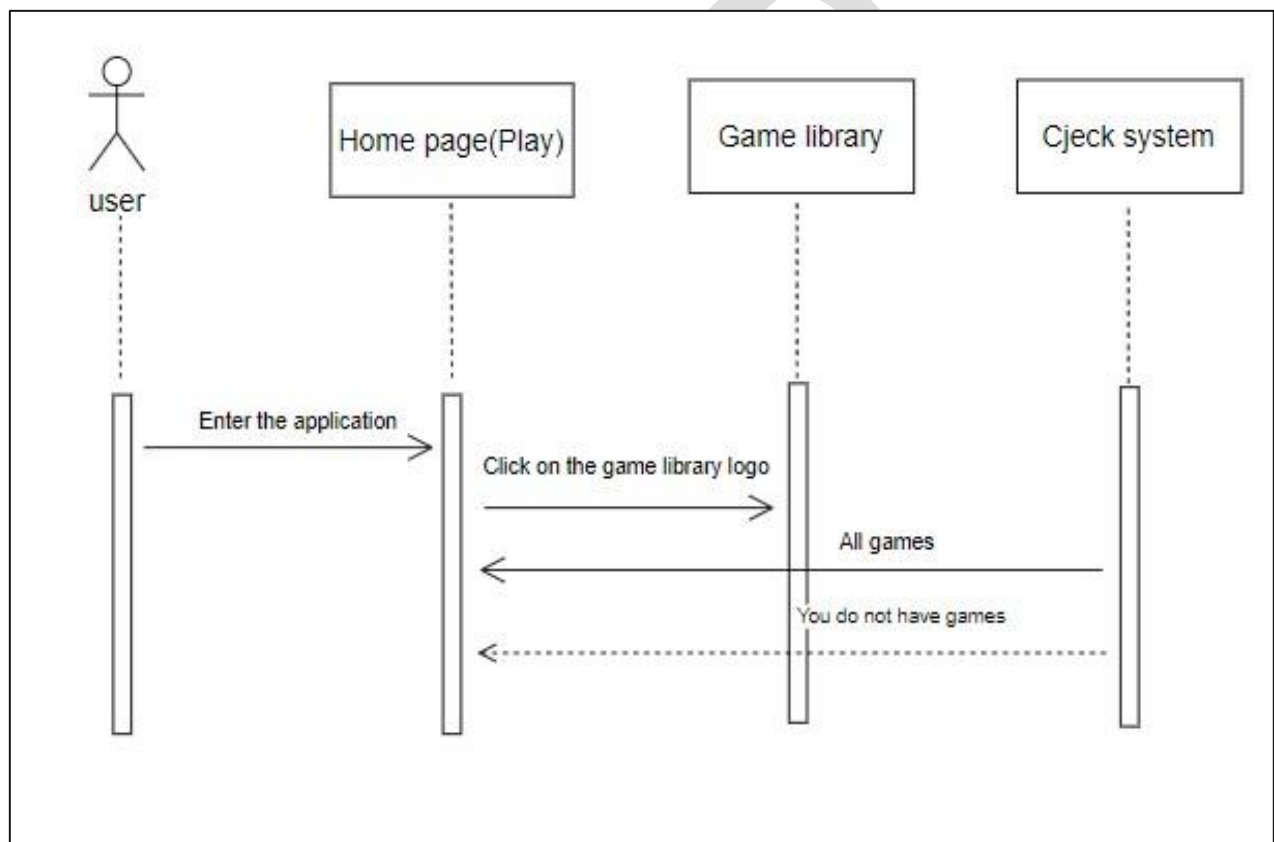
Table 2:



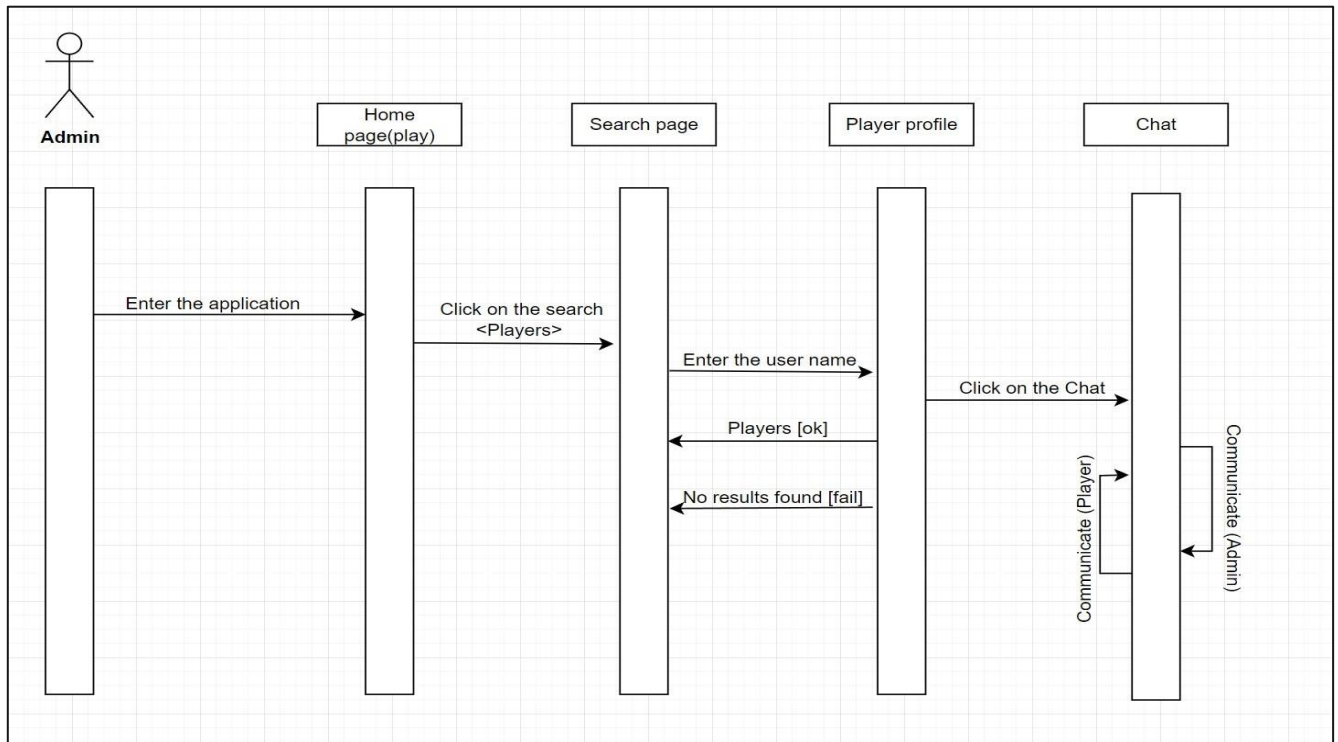
Admin: Communicate with users	
<b>Actors</b>	Admin and DB of User
<b>Description</b>	It explains the admin's method of communicating with users and helping them with any technical problem they face, as well as responding to their inquiries.
<b>Data</b>	User name, user ID, user email, admin ID.
<b>Stimulus</b>	Want to Communicate with users Admin.
<b>Response</b>	Helping them by (communicating with them, solving problems, and protecting users)
<b>Comments</b>	No Comments.

## Sequence Diagrams:

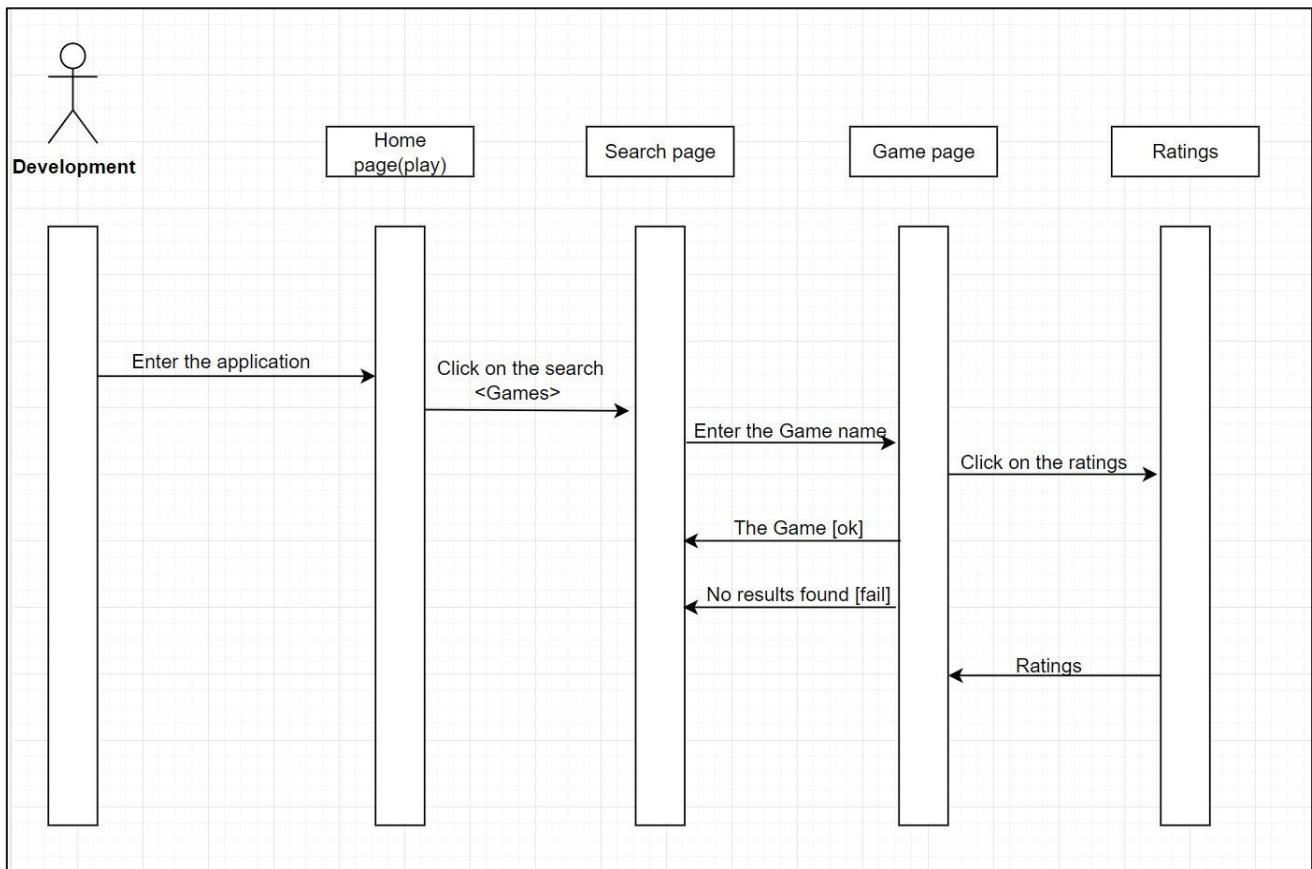
)View games)



### )Review player ratings)



Communicate with users)



Class Diagram:

#### Class user(player)

- Contact ID (Int)
- Name (String)
- Emile (String)
- Nickname (String)
- Address (String)
- phone # (Int)
- Bank account (Int)
- password (Int)
- language (String)
- country (String)

#### Class games

- ID(Int)
- Name (String)
- Version (Int)
- Age classification (Int)
- Developer name (String)
- Date created (Date)
- Operator type (Int)
- Type game (String)
- Abstract (String)

- New account ()
- Add ()
- Delete ()
- Update ()
- Search ()
- download ()
- Review ()

- add ()
- delete ()
- Update ()
- Review ()
- maintenance ()

#### Class Developer

- ID(Int)
- Name (String)
- Game id(int)
- Name game (String)
- license number (Int)
- Address (String)
- Email (String)

- Add ()
- delete ()
- Update ()
- Review ()
- maintenance ()
- Support ()
- Design ()



