BK TP.HCM

BK - OISP

FACULTY OF COMPUTER SCIENCE AND ENGINEERING

PRACTICE ON SOFTWARE ENGINEERING CARO WEB GAME

Lecturer: Mr. Quan Thanh Tho

Name	ID
Trần Quốc Anh	1852247
Lê Trung Hiếu	1852365
Nguyễn Gia Huy	1852405
Thái Thành Tài	1852080
Nguyễn Tiến Thành	1852740

April 28, 2020
DOCUMENT HISTORY

Date	Version	Changes	Person in charge
Apr 28	1.1.1	- Add introduction and reformat	Le Trung Hieu
Apr 28	1.1	- Add Use case diagrams, Use case detail, User story, Flow chart	Whole team
Apr 27	1.0.1	- "Chatting system" added User story, Use-case diagram, Flow chart	Thai Thanh Tai
Apr 16	1.0	- Write report - Describe 5 functions	

CONTENT

FUNCTIONAL REQUIREMENTS:

1. Introduction
2. Use-case diagram
3. Methods of implementation
4. Functions description:
   a. Login and Register
   b. Player's Interaction
   c. Navigate page
   d. Communicate
   e. Process game logic

FUNCTIONAL REQUIREMENTS

User management system: Login screen and user's data storage.

Chatting system: allow users to communicate for personal purposes and interact.

Navigation page: direct people to the content users are looking for.

Game logic: handle the internal mechanism of the game in order to perform all the tasks needed for it to work.

Player's Interaction: Two players after being navigated to a room, they will see a caro board and choose who plays first. Players move their cursor and click on board to play when in turn.

1/ INTRODUCTION

Our demand for entertainment has been improving recently, which results in the introduction of numerous video games and entertainment applications. In addition, people these days are prone to use the web rather than offline applications to save memory. Therefore, we decided to develop a web-app named Caro Web Game. In this way, we can bring people advantages in suffering from stress and probably create a community for people having the same hobbies.

Caro Web Game is a well-known game for almost all generations. Our web-app provides a playground for various players. Users can communicate via the chatting system or compete with others. In this way, they not only play games but also make new friends or establish new relationships.

2/ USE-CASE DIAGRAM
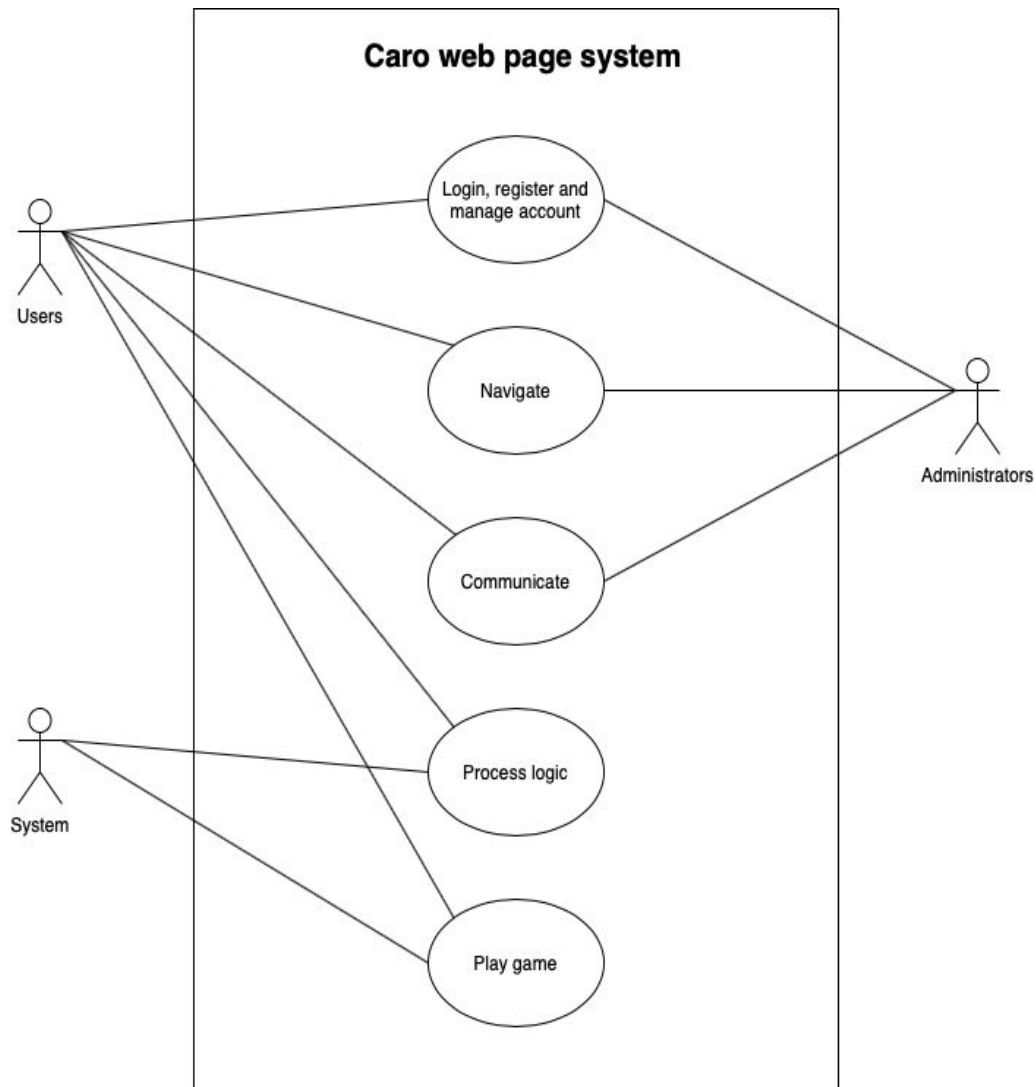
3/ METHODS OF IMPLEMENTATION

- Platform: Web
- Front-end: HTML, CSS, Bootstrap, Javascript
- Back-end: Python, Django Framework
- Database: Postgresql

4/ USE CASE SCENARIO

a. Login, register and manage account
- *Scenario*

Use Case ID:	1		
Use Case Name:	Login, register and manage account		
Create By:	Nguyễn Gia Huy	**Last Updated By**	Nguyễn Gia Huy
Date Created:	27/04/2020	**Date Last Updated**	27/04/2020
Actors:	Users		
Description:	Users can register and login to the web. Therefore, they can use all other features of the app (play games, chatting, etc...) They can also see their information, like their rank level, match history, chat history or maybe change their password.		
Trigger:	Users		
Preconditions:	Users must have a unique username and email, or they will have to create one.		
Postconditions:	Users can now access to all other features of the app		
Normal Flow:	1. User accesses the home page, he will encounter a login screen. If he already has an account, he can login normally, otherwise he will have to create one. 2. To register, users need to plug in email, username, and password. Username and email have to be unique, meaning that they cannot be duplicated in the database. The request will be sent to the server and will create a new user if validated. 3. To login, users need to input the existing username/email and the corresponding password. The request will be sent to the server and will be queried in the database for validation. 4. Once the user has login, they can see their match history, chat history, rank level and change their password if necessary. 5. System will show the correct data for users, as well as updating new data to the server.		

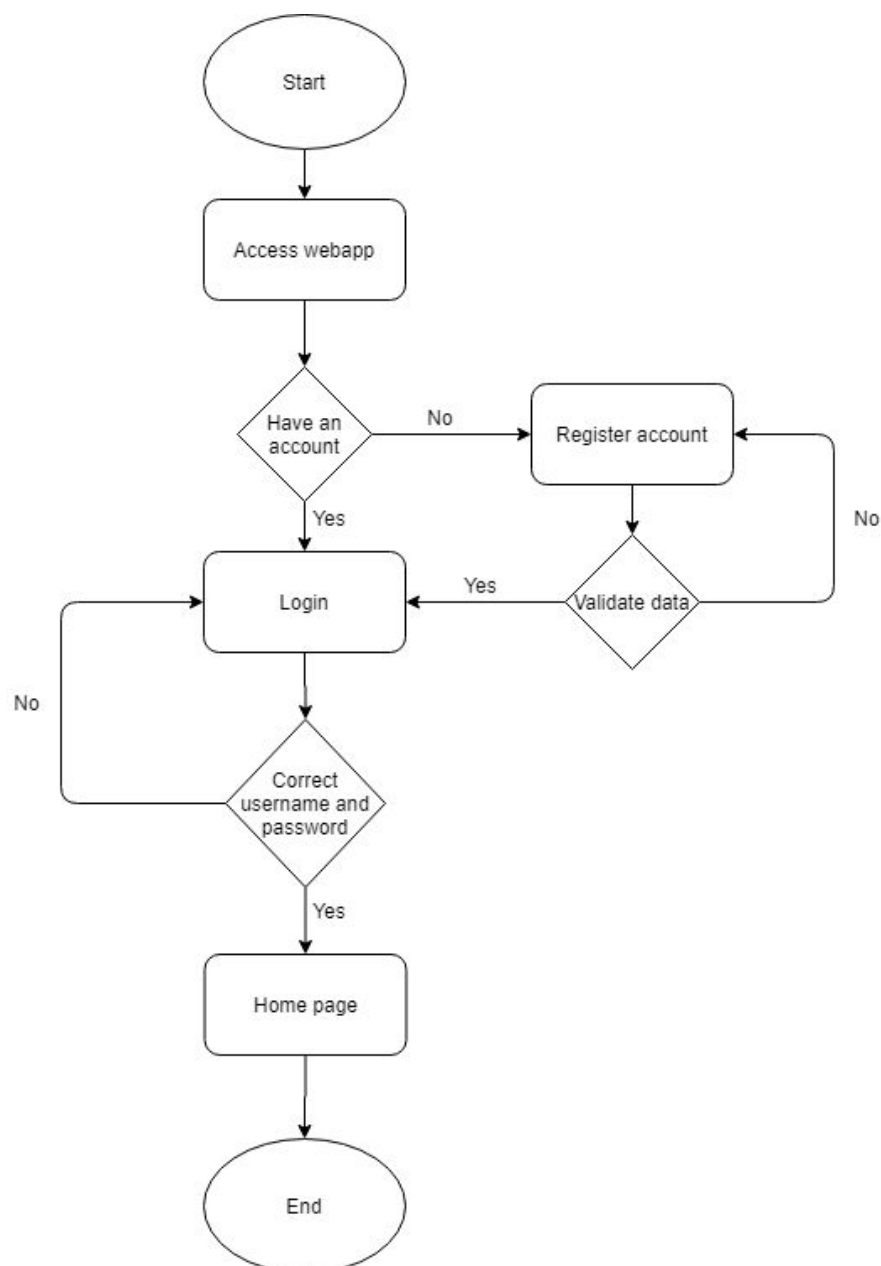|	
Alternative Flow:	
Exceptions:	Exception at step 1: 1a. If a user already has an account, jump to step 3.
Notes and Issues:	

● *Use case diagram*
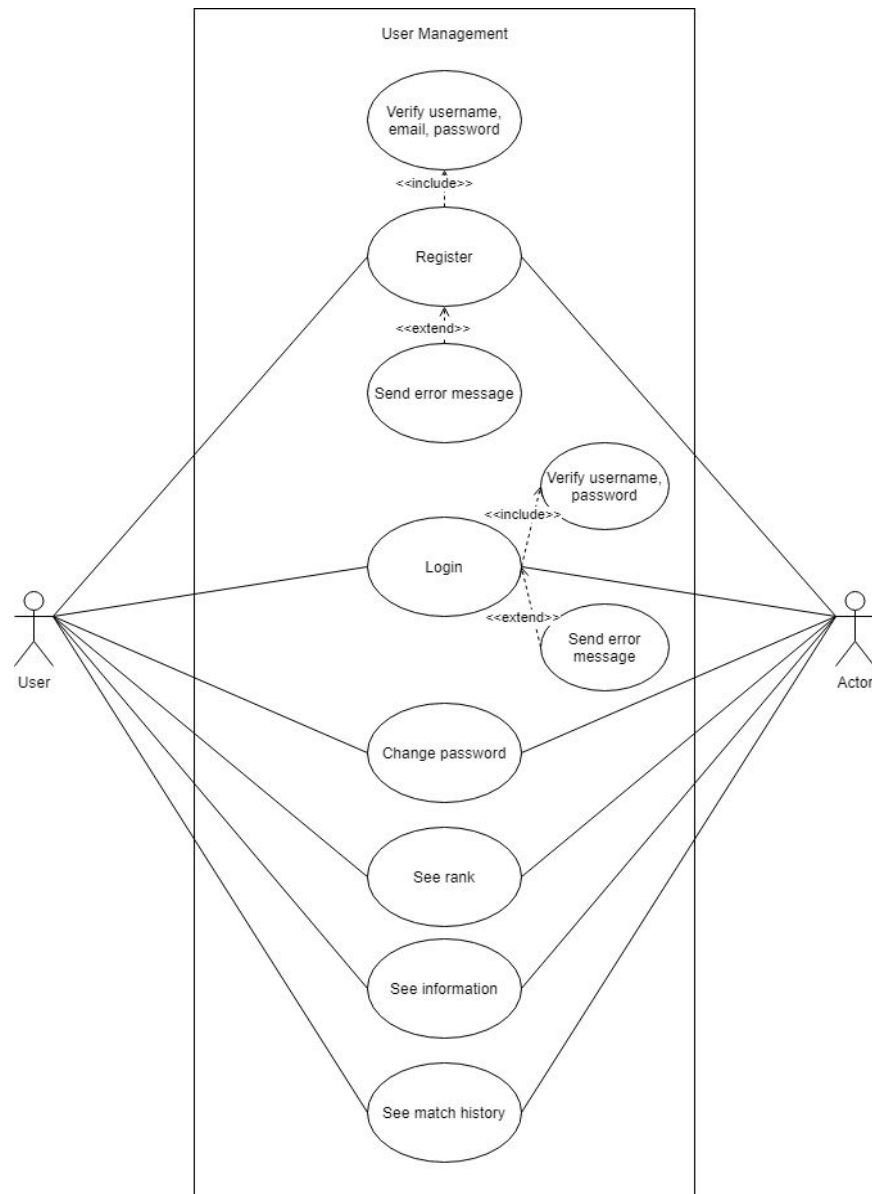
- *User story*

  _ As a guest user, I want to register an account to connect to the webapp.
  _ As a member user, I want to login to my account and start using the app.
  + If I want to see my account information, I would click on my name on the top-right corner of the page.
  + If I want to change password, I would like to click on "Change Password", where I can plug in my current password and a new password.
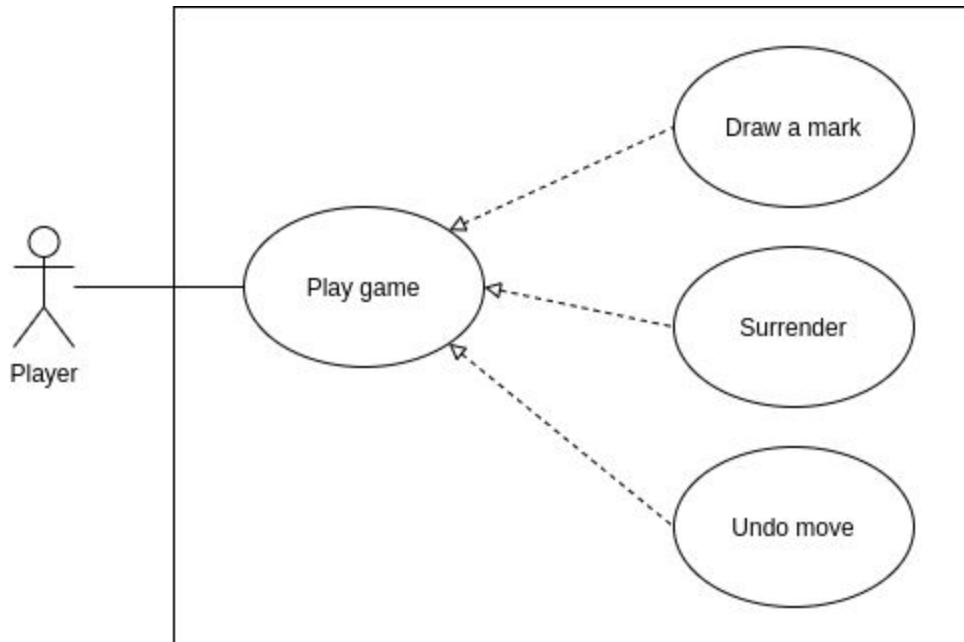
- *Flow chart*

```
                    ┌──────────┐
                   (   Start    )
                    └────┬─────┘
                         │
                ┌────────▼────────┐
                │  Access webapp  │
                └────────┬────────┘
                         │
                   ◇ Have an ◇     No    ┌──────────────────┐
                   ◇ account ◇──────────▶│ Register account │
                         │ Yes           └────────┬─────────┘  │ No
                ┌────────▼────────┐   Yes     ◇ Validate data ◇
        No      │     Login       │◀──────────◇              ◇
                └────────┬────────┘
                   ◇ Correct ◇
                   ◇ username and ◇
                   ◇ password ◇
                         │ Yes
                ┌────────▼────────┐
                │   Home page     │
                └────────┬────────┘
                         │
                    (   End    )
```

Start

Access webapp

Have an account

No — Register account

Yes

Login

Validate data — Yes

No

Correct username and password

No

Yes

Home page

End

b. Player's Interaction
- *Use case diagram*

- *Scenario*

Use Case ID:	2		
Use Case Name:	Play game		
Create By:	Nguyễn Tiến Thành	Last Updated By	Nguyễn Tiến Thành
Date Created:	27/04/2020	Date Last Updated	27/04/2020
Actors:	Player		
Description:	Player compete with an opponent in a match room to find winner and collect score		
Preconditions:	2 players are in a match room page		
Postconditions:	A player won and his/her score is updated in record		

Normal Flow:	1. Player clicks on "Ready" button and waits for opponent's ready 2. 2 players are ready then match begins 3. Player clicks on a cell of the board 4. System checks whether it is a valid move 5. If valid, detect the player's symbol 6. Draw that symbol on the cell 7. Check for winning condition: if no winner, switch turn to opponent and repeat step 3
Alternative Flow:	*Alternative 1:* at step 7 7a. If there is winner, end the game *Alternative 2:* at step 3 3a. If player clicks "Surrender" button 3b. Stop the game 3c. Opponent is the winner, adjust 2 players' score and rank *Alternative 3:* at step 3 3a. Opponent from previous move clicks "Undo" button 3b. System asks for player's permission 3c. If not allowed, continue step 3 for current player 3d. Otherwise, delete previous opponent's move 3e. Reset turn to opponent and continue with step 3
Exceptions:	*Exception1:* at step 1 1a. Player clicks "Leave" the room, then remove player from room page
Notes and Issues:	

- User story

  - As a player, I am in a room with an opponent. When I am ready, I click the ready button and wait for the opponent's ready.
  - If I go first, I will place a mark on the board and wait for the opponent's turn, otherwise I wait for my opponent to go first. We will play until there is a winner.

● Flow chart
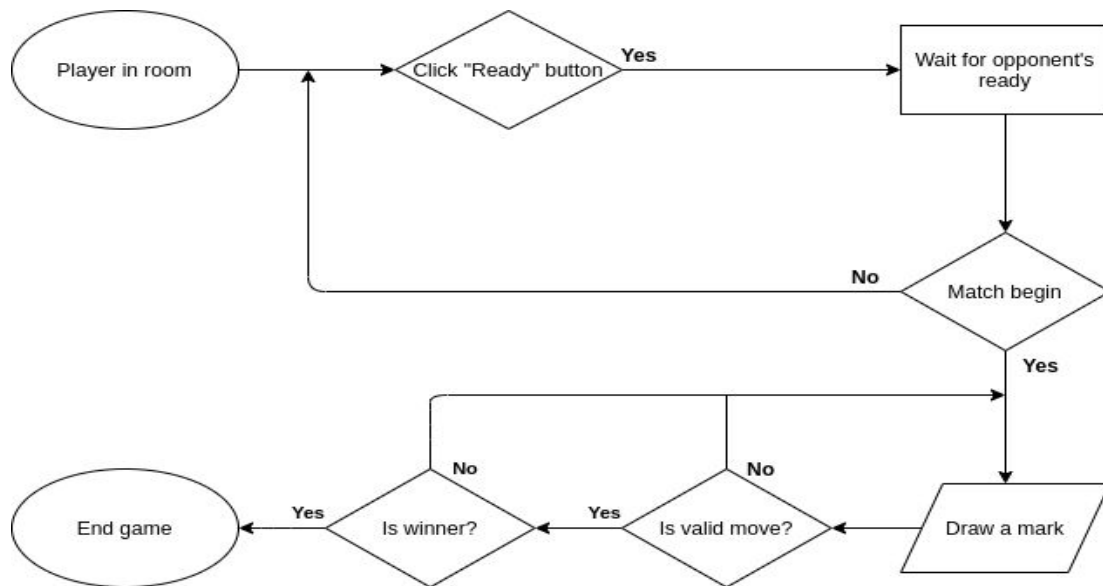
c. **Navigate page**
   ● *Scenario*

Use Case ID:	5		
Use Case Name:	Navigate		
Create By:	Trần Quốc Anh	Last Updated By	Trần Quốc Anh
Date Created:	27/04/2020	Date Last Updated	27/04/2020
Actors:	User		
Description:	A user interface element that contains links to other sections of the website.		
Trigger:	User want to access to page and play game		
Preconditions:	User has internet and browser to access to the page		
Postconditions:	None		

|	
Normal Flow:	1. Users access to the homepage. 2. Click on what they want to see in the navigation bar included: "Home", "News", "Download", "Contacts", "Play now". 3. Users log in with username and password to play game with other players. 4. Users are moved to choosing room page, and they can click "Join available room" or "Create new room" 5. Display Caro table and opponent, user can interact with table to play. 6. If player's score is in top 10, they will be display in ranking table
Alternative Flow:	Alternative at step 3: 3a. Users can click "log in as quest" to play without sign up, but the score will not be saved.
Exceptions:	
Notes and Issues:	None

● *Use-case diagram*

● *User story*

- As a user, I want to be able to easily navigate to the home page and play game.
- I want to be able to know how the site is organized simply by looking at the menu so that I can easily find what I'm looking for.
- I want to be able to orient myself within the site at all times.
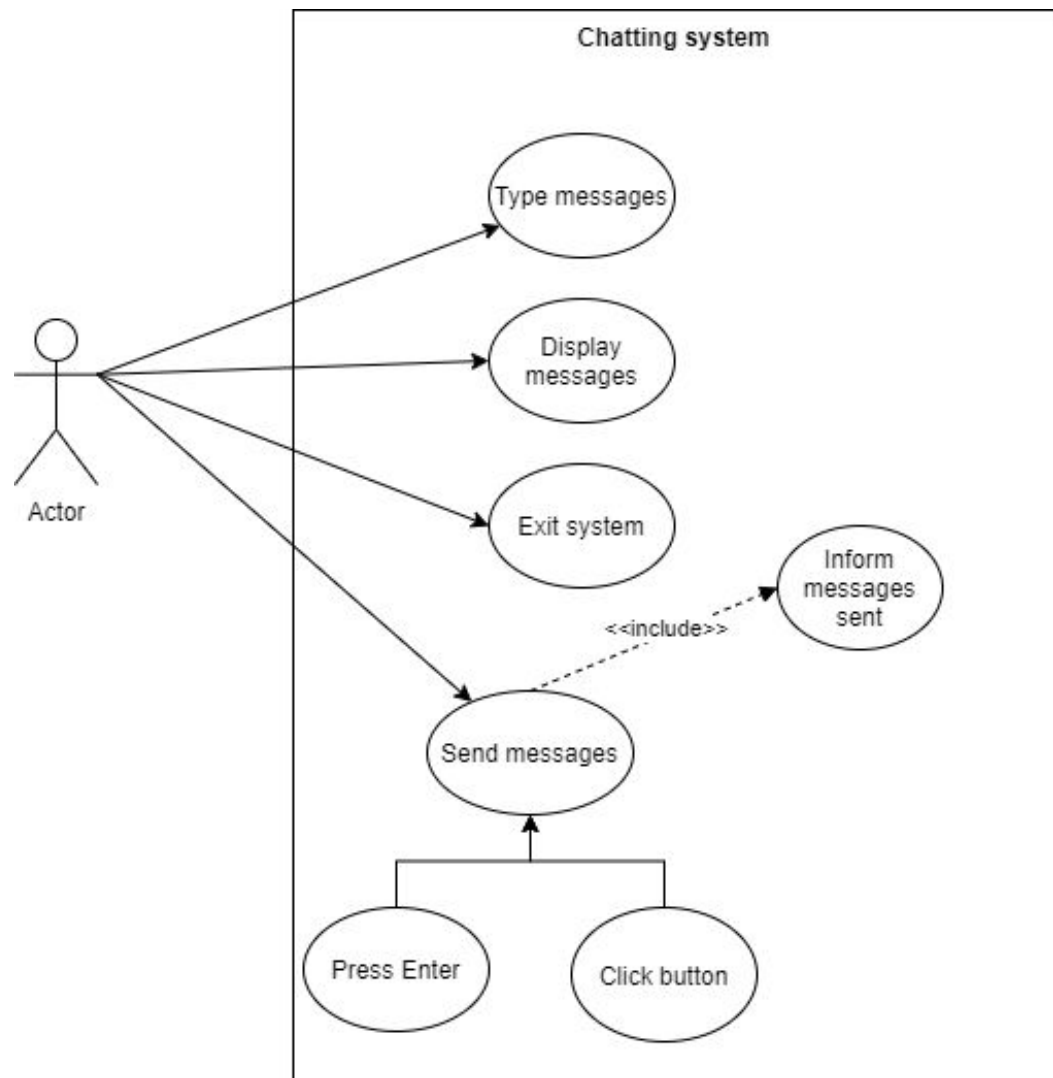
● *Flow chart*
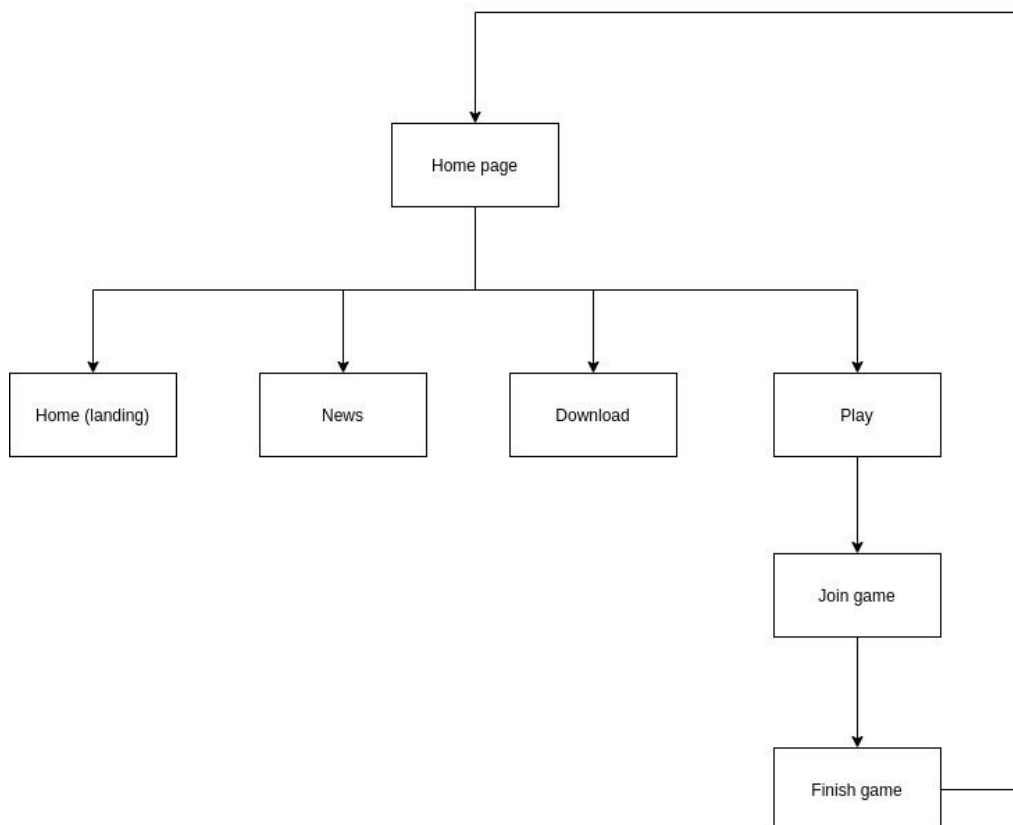
d. *Communicate:*
   ● *Scenario*

Use Case ID:	4		
Use Case Name:	Communicate		
Create By:	Thai Thanh Tai	Last Updated By	Thai Thanh Tai

Date Created:	Apr 27	Date Last Updated	Apr 27
Actors:	Users		
Description:	Users can access to chatting system (chat box) to send the messages to competitor in the same room. Another participant can see and reply on these messages by the same way. Chat box can show time when messages sent. Chatting system also can be Minimized or Exit through buttons.		
Trigger:	User		
Preconditions:	1. Users must have account to login if they want to communicate 2. Users have the Internet connection. 3. Players send messages in the same room. 4. Chat box should contain some characters to activate "Send" button.		
Postconditions:	None		
Normal Flow:	1. Chatting system is available when battles start 2. People can type text in a given blank box. 3. After texting, people can press Enter or use the Send button to send messages. 4. After sending, there will be an announcement that messages sent successfully, that text will be blurred soon. 5. Users can minimize or turn off the chatting system to save space for other graphics.		
Alternative Flow:	3.3 If text is blank, people can not send, Sent button will be deactivated		
Exceptions:	No Internet signal		
Notes and Issues:	No		

● Use-case diagram

Chatting system

Type messages

Display messages

Exit system

Inform messages sent

<<include>>

Send messages

Actor

Press Enter

Click button

● User story
   - As users have a desire to communicate with others (participants, players) to transfer what they want.
   - I want to keep in touch with others for the following battles or championships.

● Flow chart
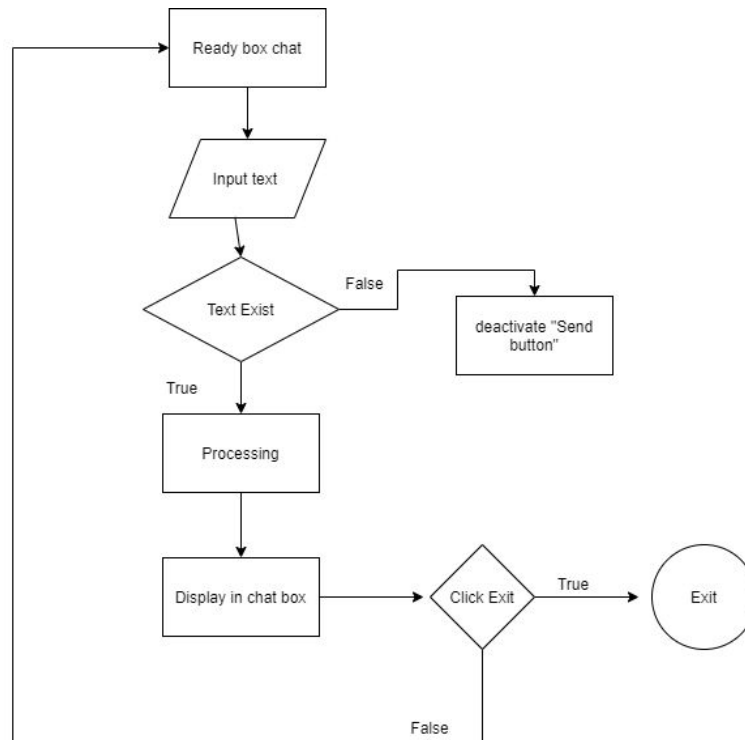
e. Process game logic
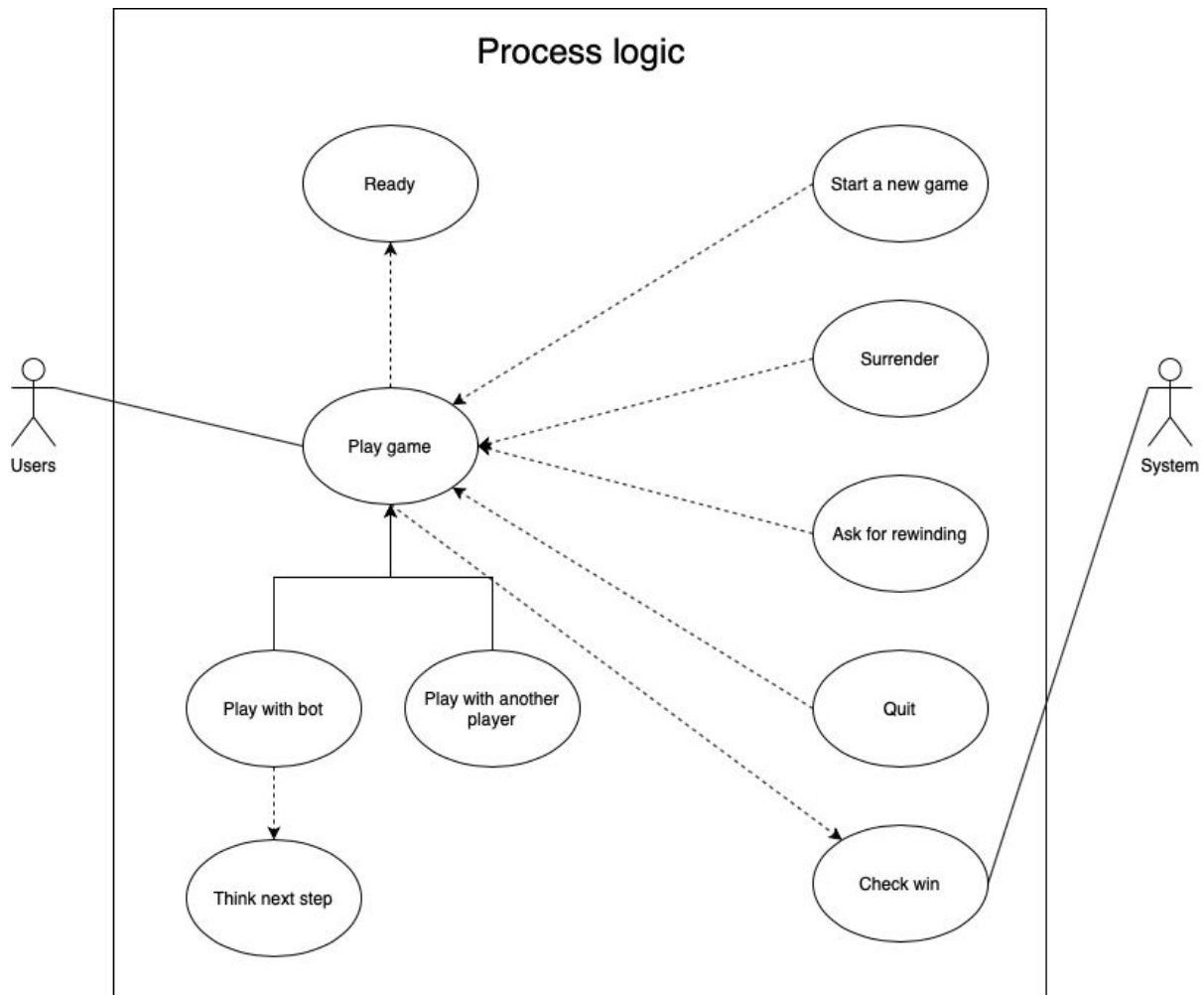● *Scenario*

Use Case ID:	5		
Use Case Name:	Process game logic		
Create By:	Le Trung Hieu	**Last Updated By**	Le Trung Hieu
Date Created:	Apr 27	**Date Last Updated**	Apr 27
Actors:	Users, system		
Description:	Users interact with the board and make their movements. The system will automatically check for win condition based on rules of the game. There are two modes, the users can choose to play with another player or a bot. In 'Player vs bot' mode, the system will automatically generate a reasonable step for the bot based on algorithms that have been		

	programmed. In 'Player vs player' mode, the users can communicate with each other through a chat system. Each player has a timing bar to handle their thinking time of a whole match. In case of time, that player will be forced to lose. Moreover, there are some options such as 'New game', 'Surrender', 'Rewind', 'Quit'... (They work as their names) When a mark is placed in the board, the system will check whether someone has won the game. After each match, the ranking point will be updated.
Trigger:	User, system
Preconditions:	Users must login into the system and find a room to start a new game.
Postconditions:	The ranking of users will be updated after a game.
Normal Flow:	1. Users login into the game and are redirected to the home page. 2. Users must join a room or accept an invitation to be redirected to the gaming page. 3. The game will start only when two players are ready. 4. In the turn, users can make a movement by clicking on the board and their steps will be displayed on the interface. 6. The system will handle the next step for AI and check for win condition each time the mark is placed. 5. If someone wins the game, the backend will handle their ranking point and users can start a new game by repeating from step 3.
Alternative Flow:	4.1. Players can send messages, surrender signals or rewind requests during playing time.
Exceptions:	4.2. If a player quit the game, the other one will win.
Notes and Issues:	No

● Use-case diagram

Process logic

● User story:

- As a player, if I want to practice, I can choose 'PvB' mode, if I want to play with another player and get a ranking point, I can choose 'PvP' mode.
- While playing, if there is no way for me to win the match, I can surrender, if I make some mistaken steps, I can ask the opponent for rewinding.

● Flow chart: