**HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY**

**Final network programing report**

**“Duoi hinh bat chu” game**

**Lecturer:** Tong Van Van

**Team member:** Le Dinh Huy – 20194776

Pham Minh Dang – 20194737

Dang Minh Khoi – 20194781

**School:** Information and Communications Technology

**I. INTRODUCTION**

**1. Motivation**

- Duoi hinh bat chu is a popular word puzzle game that challenges players to form words by connecting a series of given letters and picture

- Although it has become a popular game, the game “Duoi hinh bat chu” can still be improved to be online game that players can play in their own devices (laptop, pc ...)

- With online game, players can play it easily in their free time with difficulty variety, familiar user interface, play with friends or pratice with computer.....

**2. Objective and scope**

- The goal of the word catcher game development project is to create an engaging and interesting puzzle game in which the player has to find out and come up with the answer by combining pictures and letters.

- Scope:

+ User Interface: We tempt to build a client which is easy to use and friendly with players.

+ Game mechanics: A match will be taked place between two people, and in a match, each player will be received same pictures and they are trying to answer the correct answer to chat board and the faster will get the point.

Difficulty level: We divide into 4 categories for difficulties, this difficulty will be based on the point of owner of that room. Ex: 0 -> 10 point is easy, 11 -> 20 point is medium, 21 -> 30 point is hard, > 30 point is advanced.

+ Game mode: We have two modes: play with real opponent and play with computer. One for challenging themselves and one for practicing.

**3. Tentative solution**

- Build a database: using Sql Lite in order to avoid waiting time to set up other databases such as Mongo Db, Sql Server...

- Using Apis resquest for authentication such as login, register.

- Using Socket protocol for interaction in game

- Because of using C++ language then we choose Qt framework for building UI easily and handful docs

**II. REQUIREMENT SURVEY AND ANALYSIS**

**1. Functional overview**

- Authentication: user can login to the app by existed account, create new account or logout from the app if user has already signed in to the app

- Playing game: after user signed in to the app, he/she can do some actions as follow:

+ Creating new room and wait for other player to join the game. Only if the room has 2 players then owner can start the game.

+ In the game, each player try to answer the question, who has the fastest answer will get the point. Players leave the game will be counted as loser and punished by point.

+ Player can join the game by the room id listed in the welcome room. Player only join the game if the room is not full or level of the room approximate with player

+ Player can train herself/himself with play with computer room

+ Player can see rank of all user

**2. Functional description**

- The app has two main usecase: authentication and playing game:

A diagram of a game

Description automatically generated

**2.1. Description of usecase Authentication**

- Name of the usecase is authentication. Below is usecase image:

A diagram of a login and register

Description automatically generated

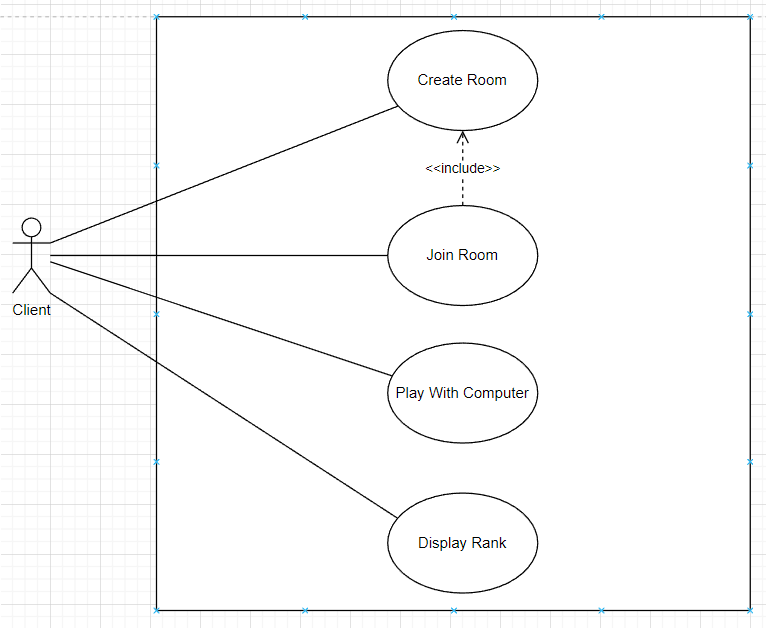
- If player has already had account then he/she can login into the app. If the username or password is wrong then an error will show up and player can not login into the app. Otherwise he/she will login into the app successfully

- If player want to sign up new account, the do it in register page. Requirements are username of new account must be different from existed accounts. There is also throw error if player register wrong username

- If player signed in, he/she can logout from the app

**2.2. Description of usecase Playing game**

- Name of the usecase is authentication. Below is usecase image:



- After player signed in, he/she can create new room and wait for other players to join the room. Only after the room has 2 players then the owner can start the game. In the game, each player tries to answer questions as fast as possible, who is fastest and has the correct answer will get a point. If one player leaves the game, he/she will be counted as loser and punished by decreasing her/his point. If the opponent leaves the game, the owner has to wait for other players of if the owner leaves the game, the room is gone and both player will be push to welcome room

- Player can join the room by the room id listed in the welcome room. He/she only joins the room if the room is not full or his/her level is fit with this room (in the room list also has the status of the room and the level of the room)

- Player can play game with computer: after he/she join play with computer room, he/she will answer questions and player can get answer if the question is so hard to get through this question

- Player can see rank of all users when they visit display rank page

**3. Functional requirement**

- With each player, there will be a new thread for handling incoming requests of each player

- Protect player information by hashing each player’s password by SHA256 algorithms, ensure authentication flow of players

- Players play game with small latency of displaying image and send answer in the game, ensure the correctness of the answer and arrange players with the same level and the point after the game will be store in database to classify level of player. Player can play with comuter to train herself/himself or see rank of all players

**III. TECHNOLOGY**

- Database structure: using Sql Lite to create database in order to store data. It takes a little time for set up and easy to use

- User interface: using Qt framework to build UI, it is quite convenient to use and familiar with user

- Restful Apis (GraphQl): using Apis request for login and register

- Tcp sockets: handling send and receive requests after player loged in

**IV. EXPERIMENT AND EVALUATION**

**1. Design**