Network Programming - HW3 Two-Player Online Game Part 3.

📖 **Congratulations, everyone! Welcome to Two-Player Online Game Part 3.**

In this assignment, you will build upon the two-player online battle game developed in Part 2, enhancing its functionality by adding features such as an invitation list, lobby broadcasting, and basic database recording.

The game lobby in this assignment should support multiple games for players (clients) to choose from. Players can upload and manage their games independently, and when creating or joining a room, they will be prompted to download the corresponding game files.

The goals of this assignment are:

1. **Enhance and optimize the lobby functionality to significantly improve the user experience.**
2. Use **sockets to Implement basic file upload/download capabilities and read/write management.**

Beyond meeting the basic requirements, we encourage students to make their projects resemble a real-world game server. Additionally, consider implementing design patterns, data structures, and algorithms to enhance your code's readability, maintainability, flexibility, and scalability.

Homework Requirements

- A. New lobby feature.
  - A-1 Basic Function
  - A-2 Save account info with an external data source, i.e .csv/t...
  - A-3 Lobby broadcasting
  - A-4 invitation Management Interface
  - A-5 Room Role
  - Bonus Challenge 1 - game switching feature
  - Bonus Challenge 2 - Lobby Chat Room
- B. Game file upload and download
  - B-1 Game Management Interface
  - B-2 Game downloading and Execution when Creating/Joini...
  - Bonus Challenge 3 - Game Rating and Feedback System
  - Bonus Challenge 4 - Expanding to Multiplayer Games
- Demo Requirement and Flow
- Grading Criteria (120pts)

Homework Requirements

A. New lobby feature.

In this HW, students should inherit the lobby concept from HW2, enhancing its functionality and optimizing its performance. You must complete two main programs, a lobby server and a client (user), and ensure their interactions meet the following requirements.

A-1 Basic Function

Similar to the requirements in HW2, except that the "game-playing" function needs to be updated to include an upload/download feature (see Section B. Game file upload and download). Students must ensure the following basic functionalities work correctly:

- User Registration
- User login
- User logout
- Display online status (user list, room list)
- Users can create public/private rooms and wait/invite other idle users to join.
- Other users can invite idle users in the lobby.
- Idle users in the lobby can join public rooms.
- Users in a room can leave the room.

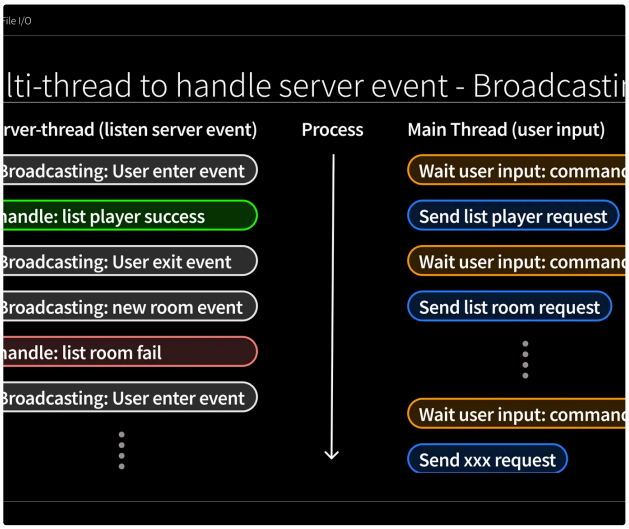
A-2 Save account info with an external data source, i.e . `csv` / `txt` or `database`

Even if the lobby server program is shut down and restarted, registered users should still be able to log in using their previously registered account credentials. Please implement a read/write feature in the lobby server to maintain users' account information using an external data source (such as CSV/txt or database) in the lobby server's directory so that it can be read when needed in the future.

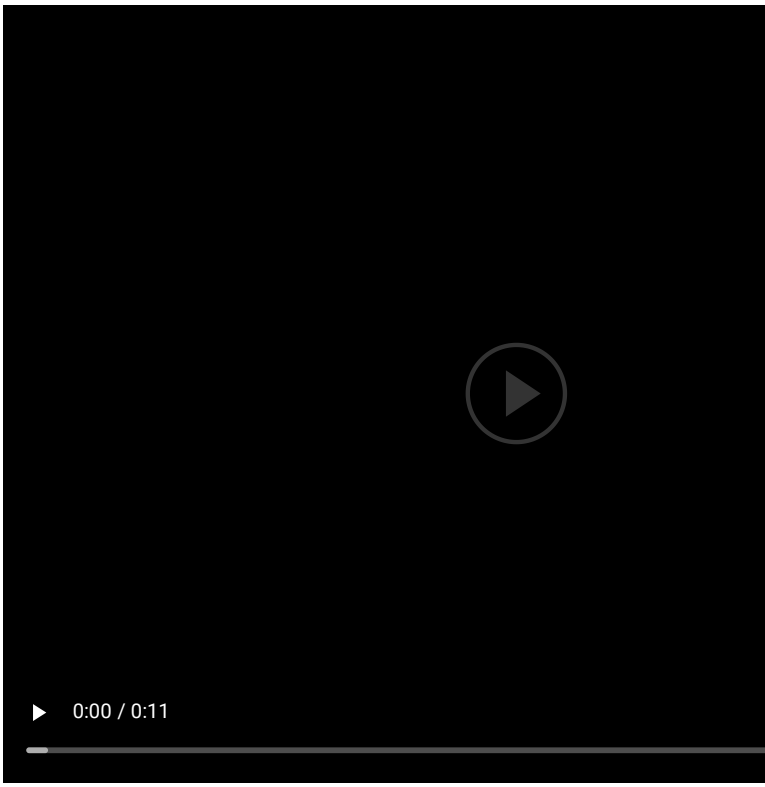
Basic login and registration

A-3 Lobby broadcasting

When a user logs in, logs out, or creates a public room,the lobby should broadcast events to all users,other users in the lobby should receive a notification message for these events.



Lobby Broadcasting - player enter

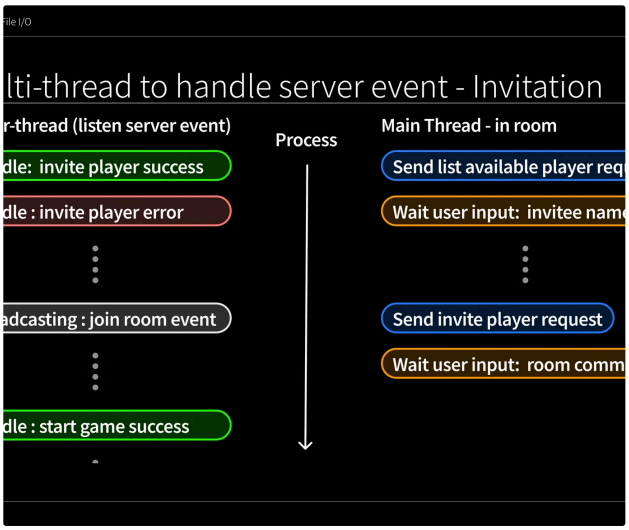


Lobby Broadcasting - player exit

A-4 invitation Management Interface

The user should be able to receive multiple invitations, and when an invitation is received, the invited user should be notified. The invited user can open the invitation management interface from the lobby at any time to respond.

(4a). If the invited user accepts the invitation but the inviter has closed the room or the room is full, the invited user should be informed and returned to the lobby.



A-5 Room Role

The user who creates a room will have the "Host" role, allowing them to perform the following functions in room

(5-1). Invite Players：

Send invitations to users in the lobby whose status is "idle."

(5-2). Start Game：

When the room is full,the host can decide to start the game.

(5-2a). If the room is not full and the host tries to start the game, an error should be shown, and the user in the room should return to the room.

(5-3). **Leave Room**:

If the host leaves the room, all other users in the room should be notified, and the host role should be transferred to a remaining user. If no other users remain, the lobby server should delete the empty room.

Bonus Challenge 1 - game switching feature

This feature awards an additional 5 pts

If the host selects the "Switch Game" option to change the game, all members in the room should download(see Section B. Game file upload and download) the corresponding game(the game should be playable).

Bonus Challenge 2 - Lobby Chat Room

This feature awards an additional 5 pts

In a real-world lobby server, a **lobby chatroom** plays an essential role in enabling players to communicate with each other. For this bonus task, your objective is to implement a chatroom that allows all players in the lobby to interact.

The basic requirement is to build a **multi-user chatroom** that is open to all players. Additionally, you can move the lobby broadcasting messages into the chatroom to prevent cluttering the lobby UI with irrelevant messages.

B. Game file upload and download

This assignment requires students to implement a game system where users can upload games to the lobby server, allowing other users to download and play them. The lobby should have a "game" folder to store games uploaded by users, and each user should also have their own game folder. This user's game folder will be used to upload games to the lobby or download games from the lobby.

B-1 Game Management Interface

Users can access the game management interface from the lobby, which should include the following features:

(1-1). List all the game which maintain by yourself:

Users should be able to view their own published games and related details (such as publisher, game description, etc.).

(1-2). User can chose their game file script to upload to lobby server:

you should provide a feature letting Users select a game file or enter a file name from their game folder to upload. After the user uploads a game, the lobby server should save the game in the lobby server's game folder, making it available for other users to download.

During the demo, you should ensure that when a user updates a game file to a new version, other users who download (see B-2) the file receive the newest game file.

(1-3). Save game file with an external data source:

The game file uploaded to the lobby server should also be stored in an external data source in the lobby server's directory to ensure that even if the server is shut down and restarted, the game file remains available for download.

Display game table and game file upload

Update the game file and make sure the file is the newest game file

B-2 Game downloading and Execution when Creating/Joining a Room:

(2-1). Download the corresponding game file when creating new room :

When creating a room, the host must select the game, and the corresponding game file should be downloaded if it is not already in the host's game folder.

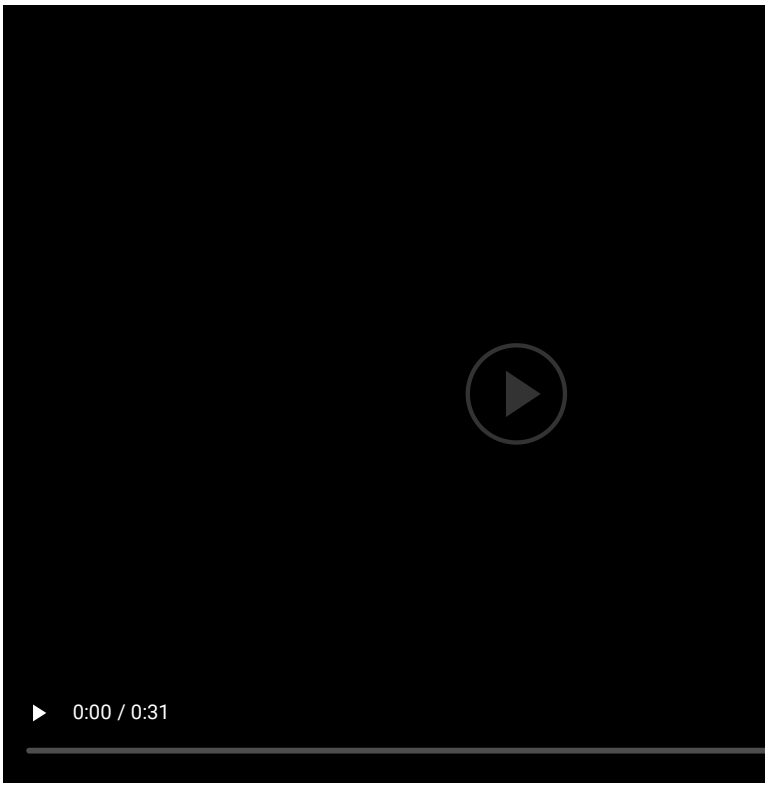
(2-2). Download the corresponding game file when joining room:

When other users enter the room, the system should check if the corresponding game script is already in their game folder. If it is not, the game script should be downloaded.

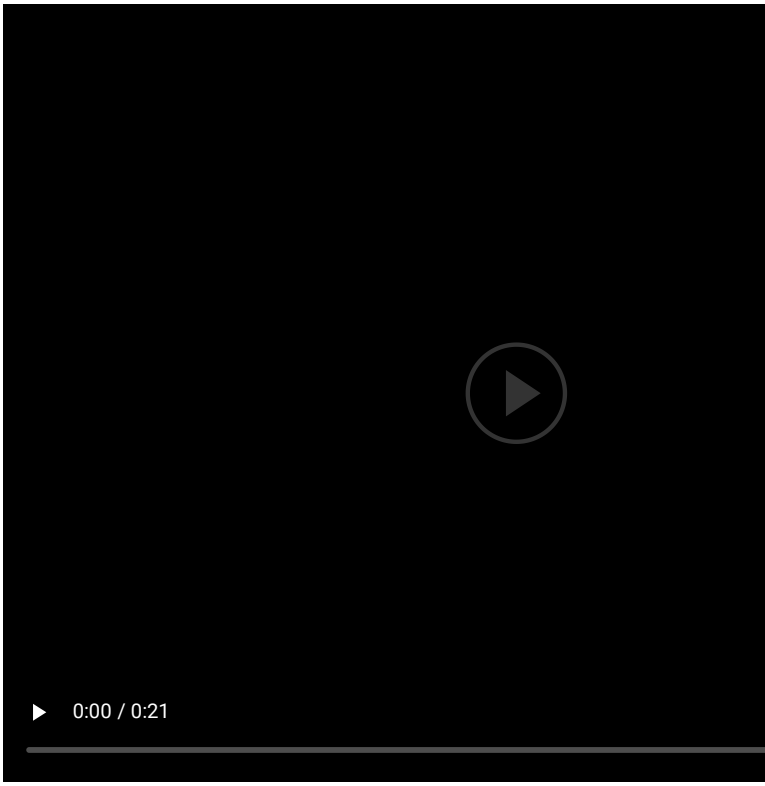
(2-3). Game Execution:

In the room, when the host selects "Start Game", all users in the room should run the downloaded game script from their game folders to begin playing.

(2-3a). Special handling is required for cases when the game content downloaded differs between users or if there is a bug in the uploaded game during execution. Any errors should not cause the program to crash; instead, the system should return to the state before the game started in the room.



Download the corresponding game file when creating new room



Download the corresponding game file when joining room

Bonus Challenge 3 - Game Rating and Feedback System

This feature awards an additional **5 points**.

In this challenge, you will implement a **rating and feedback system** for the games in your lobby. Players should be able to:

1. **Rate the game** on a scale (e.g., 1 to 5 stars).

2. **Provide written feedback** on their experience.

You should also create a user-friendly **UI display** where:

- All submitted feedback and ratings are shown for each game.
- The **average rating** for each game is calculated and displayed to help new players gauge its quality.

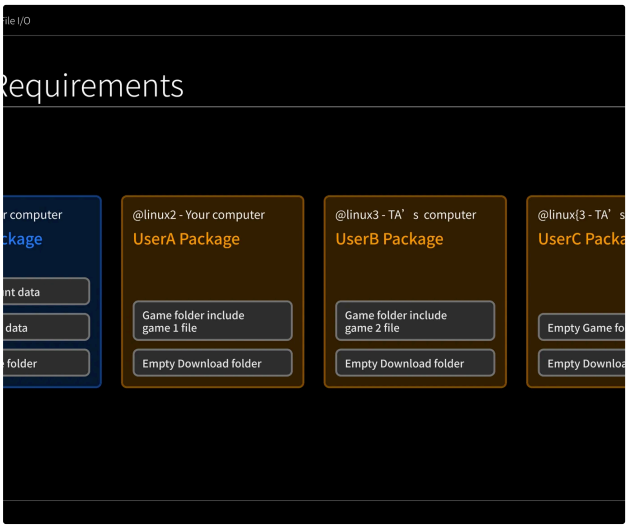
Optional enhancement: Allow users to edit or delete their feedback, and consider adding sorting options (e.g., by date or rating).

Bonus Challenge 4 - Expanding to Multiplayer Games

This feature awards an additional **5 points**.

In this challenge, extend the current system to support **multiplayer games** beyond just two players.

Demo Requirement and Flow



- Please place the program and all relevant folders on the NYCU CSIT's linux server
- When an event occurs (EX: user login/logout, room creation/deletion, or game upload), the user list, room list, and game list in the lobby and for all users must be consistent during queries in order to receive points.
- For the demo, you will need the lobby, userA, userB, and userC. Please use your own computer to connect to the NYCU CSIT's linux server and run the lobby and userA. And then assist the TA in connecting to the server on the TA's computer using your account to run userB and userC. The TA will use the UI seen on userB and userC as the basis for evaluation and scoring. Please ensure the program has clear prompt messages; if the prompts are unclear, some scoring criteria will not be counted.
- The lobby, userA, userB, and userC must each have their own game folder. The game folder for userB should contain at least 1 game files,the game folder for userA should contain 1 game files (should be different from userB's) , while the game folders for the lobby and userC should initially be empty.
- TA will first use userB to upload a game file you provided in userB's folder and test the related upload/download interacting functionalities. Then, TA will modify the file in userB's folder (e.g., by adding a line `print("modified")`) and upload it again. You need to ensure that when other users re-enter the room with this specified game, they download the modified version of this file.

Grading Criteria (120pts)

A. New lobby feature (35 pts):

- Lobby event broadcasting (5pts)
- Automatically or manually switch Host (5 pts)
- Save account info with external datasource i.e . `csv` / `txt` or `database` (10pts)
- User Invitation management (15 pts)

B. Game file upload and download(65 pts)

- List All Game (10pts)
  - Displaying the available game table with Name / Author / Description attribute (5pts)
  - Save game info with external datasource i.e . `csv` / `txt` or `database` (5pts)
- Upload Game (30pts)
  - Game Management Interface (5pts)
  - List all the game which maintain by yourself (5pts)
  - Upload game file to create / update new game (20 pts)
- Download Game (25 pts)
  - Select the game and download corresponding script when user create new gaming room. (15 pts)
  - Download corresponding game when user enter existing gaming room. (10 pts)

C. Bonus Challenge (20 pts)

- **Challenge 1.** Host is allowed to change game he/she want to play **(5pts)**
  - **Challenge Level:** 🌕🌗⚫⚫⚫
- **Challenge 2.** Lobby Chat Room **(5pts)**
  - **Challenge Level:** 🌕🌕🌗⚫⚫
- **Challenge 3.** Player can rating and command (feedback) on game **(5pts)**
  - **Challenge Level:** 🌕🌕🌕🌗⚫
- **Challenge 4.** Develop the game played by more than two players **(5pts)**
  - **Challenge Level:** 🌕🌕🌕🌕🌗