

ICS 167: Multiplayer Online Systems Project

Pong Game

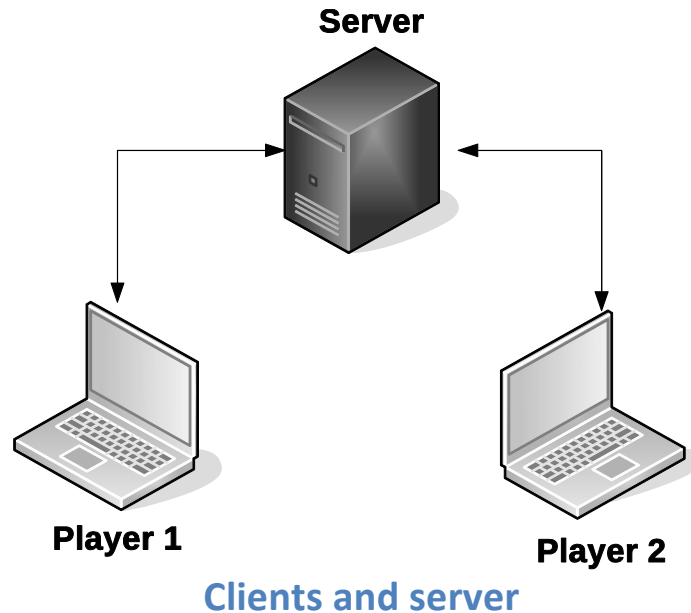
In this project, you are going to build a client-server multiplayer pong game. The goal of this project is to get you familiar with the common issues network game developers face, including network latency & jitter, connection management, data consistency, etc. Therefore, the focus is on the engineering aspects of game development. Other aspects such as art and environment will not be part of the criteria for evaluation of the project.



Pong game

There are two parts you need to implement:

1. **Frontend (web browser client):** This will be the interface for the players to interact with the game. You are going to use html and javascript to implement this part.
2. **Backend (game server):** The server runs a game loop that decides what events occur and when do they occur. Besides, since the web browser clients cannot communicate directly with each other, the server should do the coordination for them. You are going to implement this part using C++.



As in the Chat Room example, the clients should connect to the server via WebSocket. You will be able to implement a working pong game by simply writing the event handlers for both sides. However, it is recommended that you know how the connection is created and managed. You are also free to modify the libraries for optimization for your projects.

The project is divided into several milestones. You need to make certain progress every week to complete the project. Each deliverable (upload to class dropbox) is associated with a grade that counts toward the overall project grade (50% of class grade).

Students are required to work in teams of at most 4. Before you start to work on the project, email the names of your team to elzarki@uci.edu. Switching teams later on will not be possible. Splitting team maybe approed depending on circumstances.

Milestone 1: Single Player (Due in dropbox 11:59pm: 2/11/2015)

In this phase you need to develop a single player pong game. Instead of playing against another player, there will only be one player who will play the game in a three-walled space. The system consists of a game client and a game server. The features you need to implement are listed as follows:

- Client
 - Text fields to specify the IP address/port of the game server to connect.
 - Text fields to enter the user's game ID or Nickname.
- Server
 - Record the connected player's ID.
 - Incorporate a physics engine that tracks the ball movement.

- Keep track of the number of consecutive hits the player makes. I.e., keep score and display on the screen the no. of hits versus the total no of times the ball was presented to the player (i.e., success rate).
- Deny service to incoming requests to start a game, when one game is already in progress..

Please put your server and client code in different folders with one team member's student ID as the name of the files and the folder. Compress them to a zip file, and make sure you upload it to the course dropbox by the deadline. Please use this Student for consecutive submissions. Always list all team members IDs and names in the files. Track participation and what each member contributed. This will be part of the overall evaluation of the team grade and individual grades.