

Albert Chang
Andrew Chen
Joanna Clark
Steven Tran
Valeria Vikhliantseva

ICS 168 Sprint Report 4 - Frogger

Current State of Game:

After working out problems with how the game connects and deals with latency through the use of Clumsy, we have upgraded our lobby system to check a table on our MySQL database to see if there are games running. If they are active, people who click “Join Lobby” can see them update and join through connecting via the listed IP address associated with the Game Name. The table also deletes games that have finished or if too many people (leaving the host by themselves) disconnect.

We have also implemented access for players to join games that have already started and handled when people disconnect from the game. Although all of these functions work we have some major bugs that we want to fix next week before showcasing the game to our peers. One major bug is that if someone has disconnected from a game previously, that game is unable to accept new players without the game crashing. Also, if the person hosting the game disconnects, the game on the database is still considered “active” and won’t disappear.

We have fixed the issues with the airplane zone where frogs would be able to move over the water. Furthermore, we have fixed the very difficult bug concerning the player’s frog being ultimately deleted if going off the screen with a log. The game now processes it as the frog dying and it respawns the player’s frog correctly at the beginning or at a checkpoint.

Plans for After This Sprint:

For the next sprint, we plan on ensuring that when players drop, others can join correctly. Since our code relies on connection IDs from each client, we need to figure out how we will index any new player correctly.

Since the next sprint is the final one, during the next week, we plan to add some extra features that we pushed aside like sound effects and music, as they were not important to the milestones at the time. If we have the time, we could also brush up the UI on the title and lobby screens to make them more user friendly.

Once everything is finished, our team was thinking about implementing a spectating functionality that allows players to join and just watch games, rather than play themselves. We thought about having the ability to view through the camera of other players or even pan the camera through the arrow keys, just as other online games such as League of Legends does.