COP 290 Assignment 3

Ping Pong Game

Aditi
2014CS10205

Ayush Bhardwaj
2014CS10091

Nikhil Gupta
2014CS50462

April 2016

Contents

1 Objectives

Design a desktop app which is:

- Simulation of ping pong game
- Allows multiple players to play at one time using p2p networking.

2 Overall Design

1. The server side will be programmed in Java using sockets.
2. Volley will be used to send requests and receive responses.
3. Doxygen will be used to create HTML documentation of the entire code base.
4. The entire code will be split up in multiple files to ensure modularity in code.

3 User Interface

3.1 Front End

4 Sub Components

4.1 Physics

Corner cases : corners

4.1.1 Databases

Table 1: Hostel Level Complaint

S.No.	Fields	Type	Description
1	Complaint Id	String	Unique Id for Complaint
2	User Id	String	Unique User Id
3	Complaint Type	Int	Complaint category
4	Complaint Content	String	Content of complaint
5	Extra Info	Image	Upload a photo
6	Admin Id	String	Id of person assigned
7	Time Stamp	Time	Time of filing the complaint
8	Resolved	Boolean	Resolved or Not
9	Mark for resolution	Boolean	Option for complaint addressee to seek approval
10	Comment	String	Any comments
11	Previous Id	Int	Previous complaint id if any
12	Hostel	Int	Hostel Id
13	Anonymous	Boolean	Anonymous or not
14	Number of Up-votes	Int	Number of Up-votes
15	Number of Down-votes	Int	Number of Down-votes
16	Number of Neutrals	Int	Number of Neutral people
17	Number of Satisfied	Int	Number of people satisfied
18	Number of Dissatisfied	Int	Number of people dissatisfied with solution

4.2 Computer Player Algorithm

The computer player can have variable speed. But the range of speed varies on the basis of level of difficulty.

- Initially computer player tries to align its paddle center with the center of the ball.
- When the ball is about to reach a wall, the computer player first calculates according to the assumption of a static paddle. And moves
- After the collision has occurred, the computer again moves accordingly.
- Another event will be choosing between ball catching versus catching power up objects. The decision will basically be based upon the distance of the power up object and the expected distance of the ball from the paddle after collision.

4.3 p2p Networking

We would use socket programming to implement p2p networking model. Each client has (n - 1) sockets, one for interaction with all the other n-1 players. Each client sends the following information to its peers:

1. Crontab will also be used to generate push notifications for devices.
2. In case a particular client is disconnected from the network, a pop up would be displayed to the other players, and according to their decision, a computer player will be added or the game would continue with one less player.

5 Interaction amongst Sub Components

6 Testing Of Components

6.1 Single peer testing

- Unit testing will be used to check if the server and end points are working correctly.
- For each endpoint, stress testing will be done via python or bash scripts to verify that the APIs perform as expected in various situations.

6.2 Computer Player Testing

6.3 Overall p2p Testing

We will use the app on our desktops once it is ready to identify and squash any remaining bugs.

7 Extra Features

- Every player gets an option of choosing the orientation of the board visible to him, according to his comfort.
- Special items would be floating around on the board, which on being captured, will provide the player with special powers.
- The special powers include : extra speed of paddle, more length of paddle and extra points.
- Some danger items would also be floating, which on being captured, would either slow down the paddle, or reduce its length.
- The game would have multiple levels that would differ by increasing ball speed and decreasing paddle length.
- While considering collision, frictional drag will also be considered.
- Corners of the board and the paddles will be circular.
- Players can play with more than one balls also.

8 Future Endeavors

9 Source Code

The source code of the project is maintained in the following repository:
https://github.com/aditi741997/COP290_PingPong.git

References