Group members: Cedric Hansen

Douglas Myrdek

Brian Emigholz

Dylan Richmond

**PROJECT DESCRIPTION**

This project will be a desktop game, whose gameplay will be similar to “Cards Against Humanity” and “Quiplash”. This game will pit 4-20 players (subject to change) against each other to find who has the funniest responses to the questions. The gameplay will be as follows. Consider a game with 10 players. In any given round, players 1&2 will get a certain random question, players 3&4 will get another random question, 5 & 6 another, and so on. Once each player has typed a response to their question, every other player will get a chance to vote on the other players response. So for example, player 1 (and player 2) will vote on whether they thought player 3 or 4 had a funnier response, then on player 5 or 6’s response, and so on. At the end of each voting round, players will get points based on how many votes they had, and the game will continue for however many rounds was decided by the host. The specifics of the gameplay are subject to change, but for now we will aim for this.

In terms of the interface, we will be creating several GUI’s to display the data/game in a reasonable, attractive format. Upon opening the application, the user will have 2 options: create game, and join game.

1. Upon pressing the create game button, the user will be brought to a screen which gives them a few options, including setting the number of rounds, max number of players, etc.., along with a field to enter their own name, and a button to actually launch the game (make the lobby available for others to join). The host will then be brought to a waiting screen which will display which users are currently in the lobby, along with a button to start the game, bringing all users to the gameplay previously described.
2. Upon pressing the join game button, the user will be brought to a screen which has text fields for their name, and the lobby code (which for now, will likely just be the IP address, or a port number, or something along those lines), and a button for them to “join” whatever game was created by the host in bullet point 1. Once the user joins the game, they will be brought to the waiting screen along with the host, but will not be given the button to start the game. These users will also be brought to the gameplay as previously described

At the completion of the game, a leaderboard will be shown, and a winner will be declared.

**SYSTEM REQUIREMENTS**

|  |  |  |
| --- | --- | --- |
| Identifier | Priority | Requirements |
| REQ1 | 1 | The system shall connect all users to the host using host’s ip address or a port number |
| REQ2 | 1 | The system shall create a new instance of a game |
| REQ3 | 1 | The system shall create new users |
| REQ4 | 4 | The system shall run on any computer with java installed |
| REQ5 | 1 | The system shall create a “client” socket for each user |
| REQ6 | 1 | The system shall create a “serverSocket” for whoever is hosting the game |
| REQ7 | 5 | The system shall allow users to add their own questions/question packs to the game |
| REQ8 | 5 | The system shall allow users to save their username and past scores |
| REQ9 | 5 | The system shall allow users to add a profile pictures |
| REQ10 | 2 | The system shall allow the game to complete rounds of gameplay |
| REQ11 | 5 | The system shall allow the host to select game settings |
| REQ12 | 3 | The system shall display the username of all players connected to the lobby while in the “Pre game lobby” page |
| REQ13 | 4 | The system will only allow the host to begin the game while in the “Pre game lobby” page |
| REQ14 | 3 | The system will display the current question, and question number for each round of play |
| REQ15 | 3 | The system will not allow users to submit blank responses to questions |
| REQ16 | 2 | The system will allow users to vote on which question they found more entertaining |
| REQ17 | 2 | The system will keep track of how many votes a certain answer received |
| REQ18 | 3 | The system will display a leaderboard between rounds |
| REQ19 | 2 | Once the host has created a lobby, there will be a text label which will contain the “room code” which will be entered by players joining the game while on the join game page. |
| REQ20 | 2 | The system will give users the appropriate question depending on which round of gameplay |
| REQ21 | 3 | The system shall give random questions to each pair of users, which will be taken from a large set of questions |
| REQ22 | 2 | The system will have GUI’s to display all of the games content |
| REQ23 | 2 | The system will allow users to answer questions |
| REQ24 | 4 | The system shall allocate the appropriate number of points based on how many votes an answer received |

\*Note: Priority 1 indicates most important, while 5 is least important

**USER STORIES**

|  |  |  |
| --- | --- | --- |
| Identifier | User Story | Size |
| ST-1 | As a host, I can create a game lobby | 10 pts |
| ST-2 | As a user who wants to join existing lobby, I can join an existing lobby by typing in the room code | 15 pts |
| ST-3 | As a host, I can start the game from the pregame lobby screen | 10 pts |
| ST-4 | As any user, I can enter in my username | 1 pt |
| ST-5 | As a host, I can see who has joined my lobby | 5 pts |
| ST-6 | As a host, I can select the settings for the game | 3 pts |
| ST-7 | As a player, I can see which question I need to answer | 3 pts |
| ST-8 | As a player, I can vote on other peoples answers | 5 pts |
| ST-9 | As a player, I can see the leaderboard at the end of every round | 3 pts |
| ST-10 | As a player, I can type my response to my question | 1 pt |
| ST-11 | As a player, I can resize my window | 3 pts |

**GITHUB REPOSITORY**

<https://github.com/dmyrdek/csc380project>