Amos Byon

Anne Kao

Jess Lai

Jon Park

Kevin Kim

Detailed Documents

Our basic GUI layout is presented and labeled in the ProjectWireFrame.pdf showing the areas for player information and actual playing field.

We will be using a MySQL database using the jdbc:mysql connector. Within our schema, we will have a single table named PlayerInfo storing the player statistics usernames, wins, losses, shots fired, and time played so that players can track their progress over time.

Character Classes: Character is inherited from Sprite and contains the map and a character data class that contains positional, health, item, and projectile information. There are functions to move the character, detect when an item has been picked up, implement item effect, detect collision with obstacles, and check for when the character is hit by a projectile.

Character also has a Move State class that has functions to implement character animation while the character is moving.

Chat Classes: The Chat Server that receives data from one client and sends it to the other. The Chat Window that sets up the chat screen GUI. The Chat Thread implements Runnable and while it runs, it receives messages from the server that the other client sends.

Item Class: Item inherits from Sprite and has functions to randomize the item, set up the item when the game starts, and activate/deactivate the item.

Main Map Class: Main Map contains a tiled map that has functions to render the foreground, render the background, and set up the obstacles that the characters cannot traverse.

Energy Bar Classes: There is an energy bar class that implements Runnable with a function that decrements the bar every time a projectile is shot. While the thread is running, it increments the bar by 1 every 100 milliseconds.

Projectile Class: Projectile inherits from Sprite. There are functions to detect whether it has collided with any obstacles.

Screen Classes: The Gameplay Screen sets up the game once both players connect and renders the game. It contains functions to draw the characters, items, and projectiles.

It also includes a Network Manager class that communicates with the Server and sends the player’s character and receives the opponent’s character.

The Screen Main Menu sets up the GUI for the first screen that appears when the program is started and takes in the username of the player and the IP address where the Server will be hosted.

The Screen result sets up the GUI for the Win/Lose screen that appears when the game is over.

Server: The Server class passes characters between the clients using ServerThread Runnables. It is also responsible for writing the game statistics to the database once the game is over.