Ice and Fire

By: Kevin Jacob, Jonathan Powers, Michael Metz, Mitch Segura, Rahul Shenoy, Austin Skeffington

Description: Our project consists of two portions, a website, and the actual game. The website is just a simple website where users are able to create accounts so that the game can distinguish stats for each player. The website has a registration and a login feature so that each user may make a “secure” account. This gives the game a location (player ID/username) to store each score so they are not just meaningless numbers. The website has a leaderboard tab where the top scores in our game may be viewed as well as an account page where the user may see their scores in the game. The game is essentially a point and click dungeon crawler where the player must try to survive by killing enemies and improving themselves with powerups and health dropped by the enemies. The movement style is similar to games like Diablo 2 where you point and click to move and target enemies. The overall goal is to gain points for slaying enemies and last for as long as you can, thus improving your score to try and get on the leaderboard.
Project Tracker: We used Jira as our project tracker but we were not actively using it since discord was just a much more effective way of communicating.
Jira link: https://5-guys-game-studio.atlassian.net/jira/software/projects/TL3/boards/1
VCS: We primarily used Unity Teams to keep track of our code for the game itself.
Link to the repository for the code pertaining to the website:
https://github.com/rjshenoy/SQL-for-Website-Project-CSCI-3308

Contributions: We used discord for the majority of code sharing and we also worked together in person using VSCode’s LiveShare feature so that we may all work off of one computer, so there are only commits from one person.

Mitch: My contributions were more heavily focused on the website. The website was built using node JS, as well as integration with SQL. I built the website's main framework, the database table that hosts users and scores, as well as posting it online. I used docker to run the server end of the website, and used AWS as a hosting platform. However, due to its pricing and bottlenecking of speed, I switched the hosting to Heroku.
Jonathan: I worked on mainly enemy spawning, player movement, and general game rules. With a great deal of overlap with Mike's portion of the project, we utilized peer review programming and collaboration in learning the Unity program and scripting in C#.
Michael: I worked primarily on the creation of the game so almost all of my work was done on Unity in creating the front end for the game and also writing numerous scripts in C#.
Rahul: I worked with the website team to construct the website with a focus on the leaderboard page. My primary contribution was creating the SQL database and a functional link between the front and back ends. I also helped with general website features.
Kevin: My focus was split by doing moderate work on both aspects of the project rather than focusing all my time on one part. For the gameplay portion, I was involved in the creation of some UIs, such as Pause Menu, DeathScreen, MainMenu, as well as the creation of the scripts

pertaining to the UIs. Towards the end, I helped create the backend of the project with some help in NodeJS and SQL.
Austin: I worked nearly exclusively on the website portion of the project, primarily in group coding sessions with Rahul and Mitch. Major features I worked on included api requests between the front and back end, general html/ejs site features like the navbar and general aesthetics, and some SQL work for the leaderboard.

Deployment: https://webdevgroupcu.org/mime9599/DEV/FIREYICE/