

CS4350 Final Project : *Battleship Implementation*

Group 410 : Abby, Andrew, Kelsey and Alexis

Our Feature : Battleship Game

Now that CoveyTown has a Game Area, the next logical step would be to add more options of games to play. A great two player game that would work well in CoveyTown is Battleship. It can be played by two people and observed by any other players that want to watch..

Since CoveyTown now has a variety of games, the leaderboard should reflect that by adding options to see wins, losses and ties per specific game and in total. That way players can view their rankings for both specific games and their overall gaming record.

Since players are expected to join and leave various towns, their scores should be able to travel with them. Otherwise every time they join a town they will always start with zeros in every category, which could be discouraging.

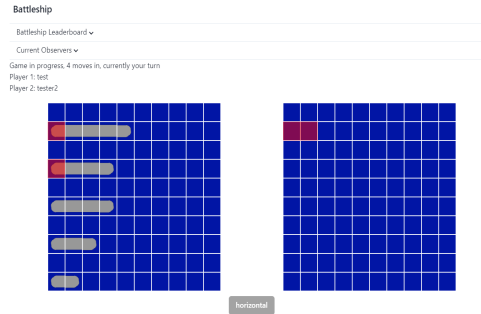
This necessitates the inclusion of an account system. To make the login process simple, users are prompted to sign up or login with a valid email address and a password. Then users are able to set a username which acts as their display name in the town and on leaderboards.

Additionally, players should be able to save their favorite towns to be able to return to meet specific players to engage in rematches to bolster their leaderboard standing.

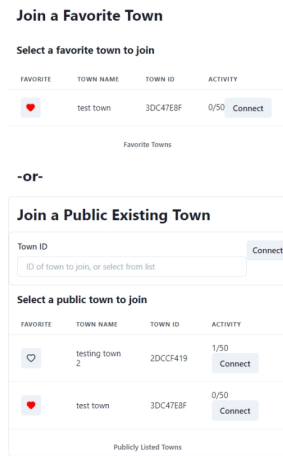
Demo and Source Links

Our demo site is available at : <https://team-410.onrender.com/>

Our code is available at : <https://github.com/neu-cs4530/fall23-team-project-group-410>

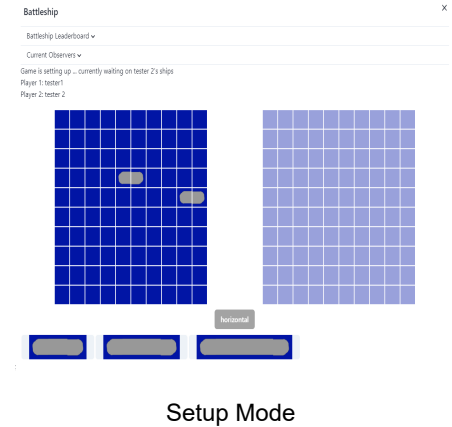


Player	Wins	Losses	Ties
tester1	1	1	0
tester 2	1	1	0



Our Technology Stack and Overall Design Decisions

We implemented a game instance of Battleship in the existing covey.town codebase - similar to how TicTacToeGame is implemented. We utilized pre-existing areas in covey.town by changing them to Battleship areas. When two players enter an "empty" Battleship Game Area, it enters a 'SETUP' phase. Once both players have completed setting up, the game is played. This is all tracked through BattleshipController, which delegates to the backend .



Our pipeline runs an automated test suite on the frontend and backend components, and then deploys the site to Render.com. It ensures that all test files are run and pass all tests, as well as making sure there are no linter errors present in any of the files.

Future Work

This is only the initial version of the login system, and many improvements could be made for future implementations. For example, it would be helpful if users had the ability to reset their passwords if they forget it. With that, it would also be helpful to have confirmation emails to verify a user's email address. All of this is doable with firebase's authentication system. It would also be worth considering if users should be able to change their usernames and if there should be any limitations on that.

Another feature we could add is the ability to edit town settings without needing to save a town password. Right now when you create a town, a toast pops up telling you to save the password for that town if you want to edit it in the future. Since there is now a login system, that created town could be stored with the user's information. Then a user would be able to edit the settings of any towns they created without worrying about saving town specific passwords elsewhere.

Future work could also include abstracting any instances of similar play between different game modes and adding other types of games.