CS4530 Final Project: Dynamic Map

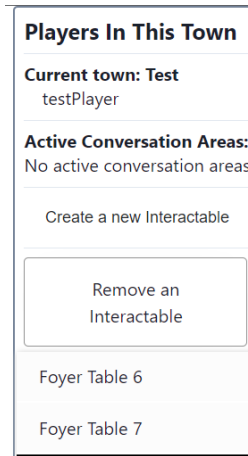Group 405: Jack Furci, Kate Lauro, Wenyu Pan, Marc Vanni

Description

The original release of Covey.Town featured one map layout with baked-in Conversation and Viewing Areas. When a user would join, they would have a choice to join whichever areas were set up for them and no more. If a user wanted to foster large-scale interactions by having less Interactable Areas on the map or have the ability to have multiple Viewing Areas wherever they pleased, there was no way to achieve this.

Giving a user the power to modify the map by adding and removing Interactable Areas allows for flexibility in interactions and makes the community of players in Covey.Town able to create a space they feel comfortable interacting in.

Our feature adds two new buttons to CoveyTown: "Create a new Interactable" and "Remove an Interactable." The button for adding an InteractableArea allows users to place a ConversationArea or ViewingArea right where they are standing (if the Interactable fits there). The button for removing an interactable allows users to choose from a dropdown list of Interactable IDs representing the current InteractableAreas on the map. The project can be found and used here:
https://main--fall-22-team-project-group-405.netlify.app/

https://github.com/neu-cs4530/fall-22-team-project-group-405

Technology Stack & Design

Our design follows a structure similar to that of the original CoveyTown codebase. We treat the addition and removal of interactables as events that get emitted to the backend TownsController, invoking API requests to either POST new areas or DELETE existing areas, respectively. Event listeners in the backend Town class are in place to alter the state of the Town in response to these requests. We needed to set up new ServerToClient events to alert the frontend TownController of these changes. The listeners in the frontend controller alter the state of the frontend representation of the Town, as long as the backend operation was successful. In the case that an operation was unsuccessful in the backend (e.g. a player tries to add an area in a place where it won't fit), a React hook will be triggered to display an error message. Otherwise, we use Phaser3 library to render the added and removed interactables on the map in response to these events.

Future Work

Our feature has much room for improvement and expansion. Currently, our implementation successfully allows users to remove existing Interactables from and add new Interactables to the Covey.town map. When a user creates a new Interactable, it is automatically placed where the user is currently standing. Possible future developments include allowing a user to choose the name of the Interactable Area they are adding, customize the size of the Interactable they are adding, and/or select where on the map they would like to place the new Interactable. Additionally, as of now, players who join a town after a player has added or removed an InteractableArea will not see these updates shown on the map. In the future we would investigate this and strive to keep the state of the town consistent even after new players join.