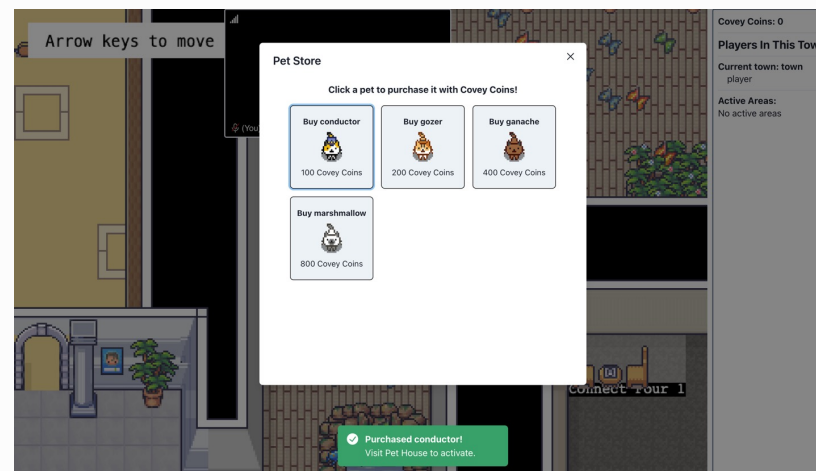# CS4530 Final Project: "Covey Pets"

Group 406: Jack Czenszak, Daniel Kim, Aidan Matrician

Pet store and pet house areas are shown as overlays on the map; pets follow around players on the town map and are visible to all other users.

Players can interact with the Pet Store to purchase new pets.

Our Feature: Covey Pets

The mission of Covey.Town is to emulate public and private meeting places where users can chat, play games, and watch videos together. The most important aspect of an online social environment is an active user base. Unfortunately, the current version of Covey.Town does little to maintain its active users or to encourage users to explore different towns. As a user, if your current conversation dies down or your Tic-Tac-Toe partner disconnects, what else is keeping you online? What encourages you to find a new town and seek more social interactions?

We developed a new feature concept for our final project: **Pets**. Pets are unlockable, cosmetic companions that accompany players on the town map. Players may purchase pets from the new interactive PetStore area with Covey Coins, which are earned by winning games of Tic-Tac-Toe or Connect Four. Additionally, players may view, select (to follow the player on the map), and rename pets at the new PetHouse area.

While Covey.Town would normally reset the state of a user after they disconnect, we additionally created a persistent database for maintaining the Covey Coin balance and pet inventory of players who authenticate with GitHub. This allows players to maintain their progress as they visit different towns and seek interactions with other players.

Demo and Source

Our demo web application is available at https://spring24-project-s24-406.onrender.com and our code at https://github.com/neu-cs4530/spring24-project-s24-group-406.

Our Technology Stack & Design

We implemented the pets feature in the existing Covey.Town codebase. Like players, the location of each client's pet is managed locally by the TownGameScene, which then emits movement events over the socket to sync with other users. Conversely, modifications to a player's Covey Coin balance and pet inventory are handled entirely in the backend when a player interacts with game areas, the PetStore, or the PetHouse—this ideally prevents malicious users from tampering with their own state and/or other users' states.

Pet store and pet house areas are represented as "objects" in the town tilemap, meaning they can be easily moved or replicated using the map editor "Tiled." Modal graphics elements for both pet areas are implemented in React and utilize Chakra elements.

Users have the option to authenticate with GitHub SSO. Once a user logins with GitHub, their authentication token is generated by the town service and stored in the React app's context. If a token is present when a user joins a new town, the town service loads their pet inventory, Covey Coins, and canonical ID from a persistent database and updates their data when they disconnect.

Our database is implemented using PostgreSQL and hosted on Heroku. Our CI/CD pipeline runs an automated test suite on the frontend and backend components, and then deploys the site using Heroku (backend) and Render (frontend).

Future Work

The design space for pets is extremely large. In addition to following players around the town map, pets could be fed/exercised, be traded to other players, be given accessories, etc. Future work will almost certainly consider extending player's interactions with their pets or adding new kinds of pets.
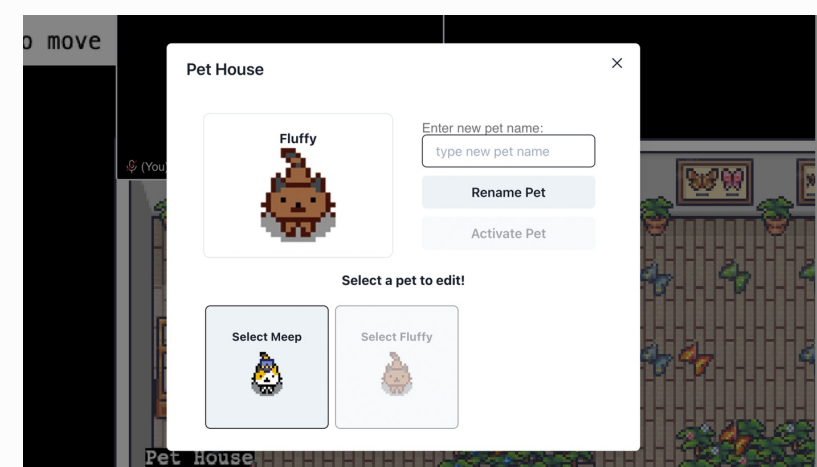
Players can interact with the Pet House to manage their inventory of pets.
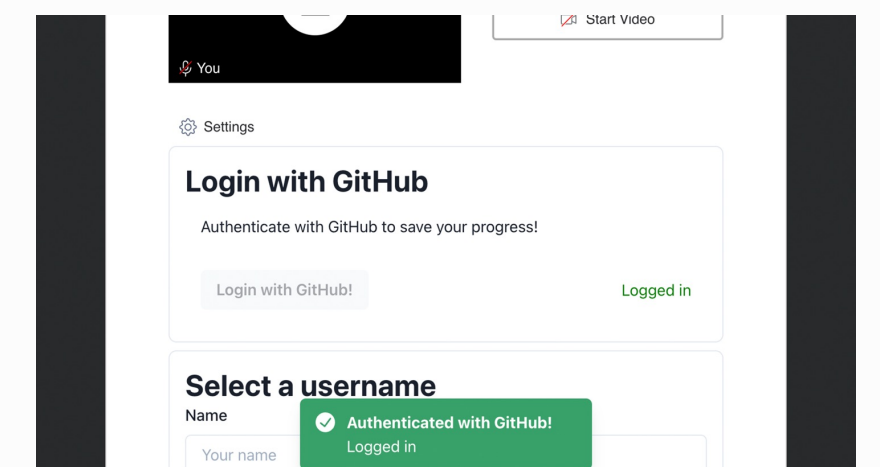
Players can alternatively authenticate with GitHub prior to joining a new town; this causes data related to their Covey Coins and pets to be loaded from/stored in a persistent database.