

CS4530 Final Project: Pictionary Game

Group 109: Ryan Cheung, Ashna Jain, Oceane Kenfack Tonfack, Kayla Wang

The Feature: Pictionary

In the current CoveyTown, there are only two game implemented. These two games are TicTacToe and ConnectFour. These are two board based, logic games, but they do not have any collaborative or creative elements.

This is where our game of Pictionary comes into place. Pictionary is an engaging game, where player on teams take turns drawing and guessing a given word. This games incorporates a competitive spirit by having teams compete against each other and fosters players creativity as they use their imagination to draw to the best of their ability.



Future Work

There are many possible extensions that we originally discussed implementing but had to hold off due to the current scope of our project. Our main extension that we wanted to implement was a screenshot capability. This would allow users to take a screenshot of the current Pictionary Board and save it to their local machine allowing users to easily distribute their drawings and show others their masterpieces

Demo and Source

Our demo is available here:

<https://spring24-project-group-109.onrender.com/>

Our source code is available here:

<https://github.com/neu-cs4530/spring24-project-group-109>

Tech Stack and Design

We implemented the Pictionary game area into Covey.Town's existing codebase, represented as a PictionaryGameArea object in the tilemap. We did not require outside components or resources to implement our game, building our drawing and whiteboard functionalities internally with our backend, and rendering it in the frontend with React.js (in tandem with everything else). In terms of design, we used the same classes to integrate both the whiteboard and game functionality rather than splitting them apart for code readability and for less complicated interactions with the game state. We built the whiteboard based on an array of Colors and Pixels, types which we defined. We took advantage of React's Color library and used CirclePicker to implement the different color options.

