CS 4530 Final Project: Voting in Conversation Areas

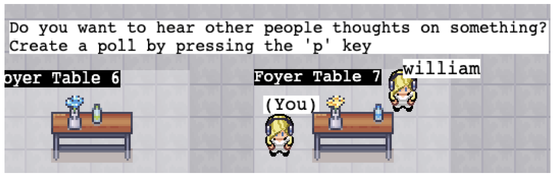
Group 2J: William Cunningham, Thomas Nguyen, Ashwin Sambasivam, Vishal Ramesh

Our Feature - Voting

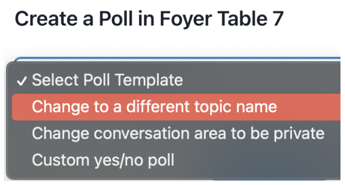
We created a voting abstraction that allows players to initiate votes which can change properties of conversation areas. We felt that players in a conversation area should have more autonomy over their conversations. So, we implemented features allowing users to democratically propose and vote on changes to a conversation area's topic and privacy settings.

We also created a custom voting feature that allows a player to ask any question to all other players in the conversation, without having a side effect on the state of the server itself.

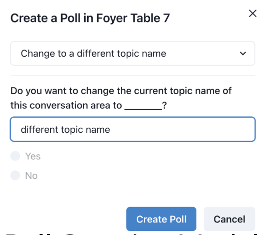
We hope our feature will promote democratic consensus building through sparking additional conversation and debate within Covey Town.


Do you want to hear other people thoughts on something?
Create a poll by pressing the 'p' key
oyer Table 6
Foyer Table 7
william
(You)

Message tooltip that appears in an active conversation area


Create a Poll in Foyer Table 7
✓ Select Poll Template
Change to a different topic name
Change conversation area to be private
Custom yes/no poll

Dropdown to select the type of poll


Create a Poll in Foyer Table 7

Change to a different topic name

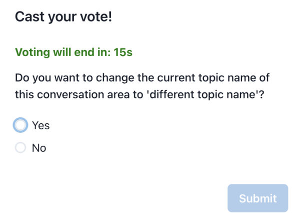
Do you want to change the current topic name of this conversation area to _______?

different topic name

Yes

No

Poll Creation Modal


Cast your vote!

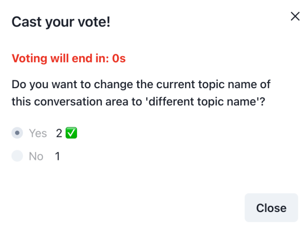
Voting will end in: 15s

Do you want to change the current topic name of this conversation area to 'different topic name'?

Yes

No

Voting Window Modal


Cast your vote!

Voting will end in: 0s

Do you want to change the current topic name of this conversation area to 'different topic name'?

[✓ Yes] 2 ✅

No 1

Post-Voting Window

Demo and Source

Our demo is available at https://merry-chebakia-c57312.netlify.app/ , and our source code is available at https://github.com/neu-cs4530-s22/team-project-group-2j.

Our Technology Stack and Design

We implemented the vote manager object which held a reference to a conversation area. Therefore, there is a one-to-one relation where each conversation area may only have one active vote manager at a time. When a player enters an active conversation area, current occupants see a tooltip message inviting them to start a vote. If the player decides to click the keyboard shortcut 'p', a React/Chakra input modal appears where the player can select which type of vote they want to start. After submitting the form, a vote manager is initialized via an api call and tracked by the backend CoveyTownController. This is then synced to each client using socket-io. All current occupants of the conversation area then see a modal pop up where they can cast their vote. This calls an api on the backend where the votes are tracked via its vote manager. After the vote timeout ends, sockets notify the frontend that the vote is over and all the votes are aggregated and reported to each player. Players then see any side effects of the vote if possible, such as the conversation topic changing.

Future Work

One of the elements of this project we are most proud of is our reusable voting abstraction. While we originally set out to create polls that just change the conversation area topic, we now see that there are many other pieces of the game we could change through voting. For example, we could easily add the ability to vote on other aspects of a conversation area, such as its list of occupants. We could also add the ability to vote on polls which would have a more dramatic visual effect on the server, such as changing the color of a conversation area.

Another aspect we would like to build upon is the existing voting options other than just simply yes/no options. We foresee expanding this to include more dynamic voting options where the creator of the vote/poll can choose the number of voting options as well as customize the option itself. In conjunction with this, it could be interesting to allow the configuration of side-effects on the frontend.