ErlangTheMovie

A Peer-to-peer Scattergories-based game

Our concurrency final project is built based on the world-famous scattergories board game!

One player creates a new lobby, and then other players can connect to them, or to other players that have already connected. The game starts when all players are ready, at which point no new player is allowed to join. After that, each round consists of submitting an answer and voting for who has the best answer, which ultimately determines the winner of the game.

We made it with all the pizzazz of everybody's favorite programming language: **Erlang**! Instead of joining a boring scattergories server, our project is built on a peer-to-peer network, where each player can connect to other players directly. This fast paced board game can now be enjoyed online, supporting multiple players and multiple rooms thanks to the strength of Erlang.

ERLANG

```
###########################################################'
###############
Game: room1   State: Waiting on User input

Current Leaderboard:
adnan: 0
jackson: 1
haijun: 2

Prompt: Languages, Letter: S

Submitted:

adnan
jackson: SUBMITTED
haijun: SUBMITTED
Answered: sql
```

```
###########################################################
###############
Game: room1   State: Waiting on User input

Current Leaderboard:
adnan: 0
jackson: 1
haijun: 2

Prompt: Languages, Letter: S

Submitted:

adnan
jackson: SUBMITTED
haijun: SUBMITTED
Answered: scheme
SCATTERGORIES #
```

```
###########################################################
##########
Game: room1   State: Waiting on User input

Current Leaderboard:
adnan: 0
jackson: 1
haijun: 2

Prompt: Languages, Letter: S

Submitted:

adnan
jackson
haijun
Enter your answer:
SCATTERGORIES #
```

Peer-to-peer networking poses unique challenges. Since each player stores the entire state of the game, there needs to be numerous checks to ensure that each player's stored state is the same as the others'. Some examples of situations that could cause desync include: a player joining right as the game starts, two players sending ready states at the same time, and re-submitting messages