App

STATE:
- **playlist:** [ ], **setPlaylist:** function
- **songs:** [ ], **setSongs:** function
- **game:** { }, **setGame:** function

PROPS:
- **playlists**: a list of playlist objects {} (each object includes an id, name, playlist_photo, songs array, rating)
- **songs**: a list of song objects {} for a specific playlist (each object includes an id, name, artist, album_photo, preview_url)

one of these views based on ROUTES

Navbar

STATE:
- ----------

PROPS:
- **user**: logged in user

Home

ROUTE: /

STATE:
- ----------

PROPS:
- ----------

PlaylistList

ROUTE: /playlists

STATE:
- ?

PROPS:
- **playlists**: a list of playlist objects {} (each object includes an id, name, playlist_photo, songs array, rating)

PROPS (ACTION):
- **setPlaylist:** function to set current playlist (accepts an id)

Playlist

STATE:
- ----------

PROPS:
- **playlist:** object { }
  - **id:** number
  - **name:** string
  - **playlist_photo:** url string
  - **rating:** number

PROPS (ACTION):
- **setPlaylist:** function to set current playlist (accepts an id)

PlaylistPage

ROUTE: /playlists/:id

STATE:
- ----------

PROPS:
- **playlist:** object { }
  - **id:** number
  - **name:** string
  - **playlist_photo:** url string
  - **rating:** number
  - **songs:** a list of song objects {} (each object includes an id, name, artist, album_photo, preview_url)

PROPS (ACTION):
- **setStart:** function that stores playlist id in state

SongList

STATE:
- ----------

PROPS:
- **songs:** a list of song objects {} for a specific playlist (each object includes an id, name, artist, album_photo, preview_url)

Song

STATE:
- ----------

PROPS:
- **song:** object
  - **id:** number
  - **name:** string
  - **artist:** string
  - **album_photo:** url string
  - **preview_url:** url string

Might not need separate component for this

Ruby Zhuang

JoinGame

ROUTE: /join

STATE:
- **name:** string, **setName:** function
- **emoji:** string, **setEmoji:** function

PROPS (ACTION):
- **onJoin:** function that joins player to game lobby

Game

ROUTE: /game/:id

STATE:
- **gameStatus:** string, **setGameStatus:** function
- **round:** string, **setRound:** function that increases number of each round
- **gameWinner:** { }, **setGameWinner:** function

PROPS
- **host:** object { // Just id?
- **users:** object { }
- **numberOfRounds:** number
- **playlist:** object { }
  - **id:** number
  - **name:** string
  - **playlist_photo:** url string
  - **rating:** number
  - **songs:** a list of song objects {} (each object includes an id, name, artist, album_photo, preview_url)

ACTIONS
- **setStart:** function that startsGame through setGameStatus
- **nextRound:** function that checks current round number against total number of rounds and calls setRound or setGameStatus accordingly
- **getGameWinner:** function that calculates game winner based on scores

game has one of these VIEWS (conditional rendering)

Lobby

ROUTE: /game/:id
if gameStatus = not started

STATE:
- ----------

PROPS:
- **users/players:** array or object of users including host
- **songs:** a list of song objects {} for a specific playlist (each object includes an id, name, artist, album_photo, preview_url)

PROPS (ACTION):
- **setStart:** function = start game

Waiting Room
- game has not started

Ruby Zhuang

GameInProgress

ROUTE: /game/:id
if gameStatus = started/active

STATE:
- **roundWinner:** { }, **setRoundWinner:** function

PROPS:
- **round:** object { } round details
  - **id:** number of round
  - **status:** started, in progress, completed
  - **winner:** { } player object
  - **song:** { }
- **roundNumber:** number of current round
- **users/players:** array or object of users including host
- **songs:** a list of song objects {} for a specific playlist (each object includes an id, name, artist, album_photo, preview_url)

PROPS(ACTION):
- **nextRound:** function to move to next round

Result

ROUTE: /game/:id
if gameStatus = completed

STATE:
- ----------

PROPS:
- **gameWinner:** { } player object of overall game winner

MusicPlayer

ROUTE: /game/:id
if gameStatus = started

STATE:
- ----------

PROPS:
- **songDone:** boolean, false initially → true when song finishes playing
- **winner:** null initially → becomes player object if song guessed
- **song:** current song object
  - **id:** number
  - **name:** string
  - **artist:** string
  - **album_photo:** url string
  - **preview_url:** url string

PROPS (ACTION):
- **nextRound:** function to go to next song/round

ACTIONS/FUNCTIONS:
- **playSongButton:** play song

MusicPlayer has one of these VIEWS (conditional rendering)

LoadingRound

STATE:
- ----------

PROPS:
- ----------

RoundUnknown

STATE:
- ----------

PROPS:
- **songDone:** boolean, false initially???
- **winner:** null initially
- **song:** current song object
  - **id:** number
  - **name:** string
  - **artist:** string
  - **album_photo:** url string
  - **preview_url:** url string

Song details not revealed when
- no one guesses song
  &&
- a song is still playing

Ruby Zhuang

RoundReveal

STATE:
- ----------

PROPS:
- **songDone:** boolean, true when song is done
- **winner:** null initially, becomes player object { } if someone guesses it
- **song:** current song object
  - **id:** number
  - **name:** string
  - **artist:** string
  - **album_photo:** url string
  - **preview_url:** url string

Display song details when
- a player guesses it or
- song finishes playing ends

Ruby Zhuang

TrackList

ROUTE: /game/:id
if gameStatus = started/active

STATE:
- ----------

PROPS:
- **round:** number that keeps track of which song is playing
- **rounds:** array/object
- **songs:** a list of song objects {} for a specific playlist (each object includes an id, name, artist, album_photo, preview_url)

songs-object-might-be-part-of-rounds-object/array

Ruby Zhuang

Track

STATE:
- ----------

PROPS:
- **currentRound:** boolean = true or false depending on if this is current round
- **round:** object
  - **id:** number of round
  - **status:** started, in progress, completed
  - **winner:** { } player object
  - **song:** { } below
- **song:** object
  - **id:** number
  - **name:** string
  - **artist:** string
  - **album_photo:** url string
  - **preview_url:** url string

ScoreBoard

ROUTE: /game/:id
if gameStatus = started

STATE:
- ----------

PROPS:
- **users/players:** object of players {}
  - **id:** number
  - **name:** string
  - **score:** 0 initially
  - **emoji:** string (or avatar)

ChatRoom

ROUTE: /game/:id
if gameStatus = started

STATE:
- **messages:** list (array) of message objects, **setMessages:** function

PROPS (ACTION):
- **setScore:** function to increase score of user with right guess, accepts id of user?
- **setRoundWinner:** function to set winner of round

MessageList

ROUTE: /game/:id
if gameStatus = started

STATE:
- ----------

PROPS:
- **messages:** list (array) of message objects

PROPS (ACTION):
- **setScore:** function to increase score of user with right guess, accepts id of user?
- **setRoundWinner:** function to set winner of round

Message

ROUTE: /game/:id
if gameStatus = started

STATE:
- ----------

PROPS:
- **message:** object
  - **id:** number
  - **user:** { id, name, emoji }
  - **message:** string

PROPS (ACTION):
- **setScore:** function to increase score of user with right guess, accepts id of user?
- **setRoundWinner:** function to set winner of round

setScore and setRoundWinner can be combined?

Ruby Zhuang

MessageInput

ROUTE: /game/:id
if gameStatus = started

STATE:
- **input:** string, **setInput:** function

PROPS (ACTION):
- **onSubmit:** function