
Design Document for Spinder

Group <SD-323>

Member1 Qusai: Backend portion of Design details (pg. 4)

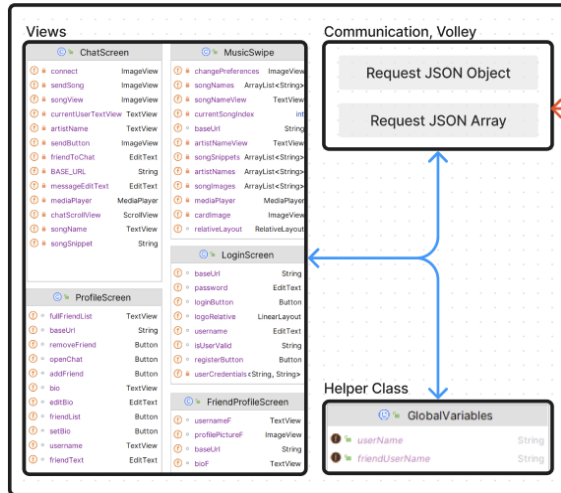
Member2 Nayma: Frontend Portion of Design details (pg. 3)

Member3 Justin: Backend and Database Block Diagrams

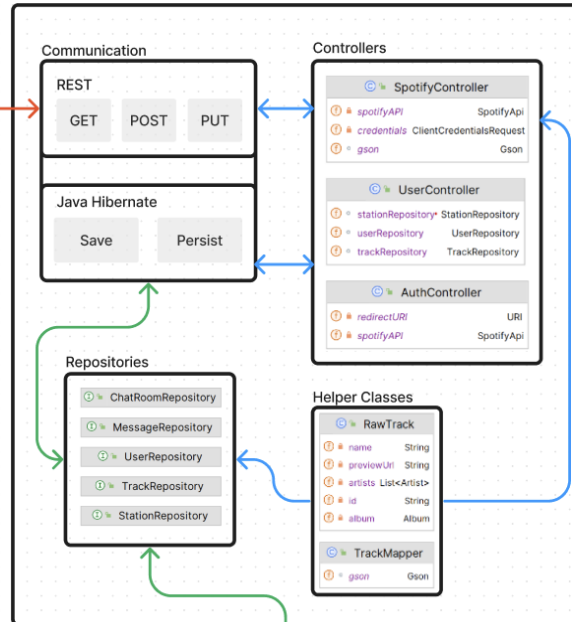
Member4 Brandon: Frontend Diagram

Block Diagram

Front End



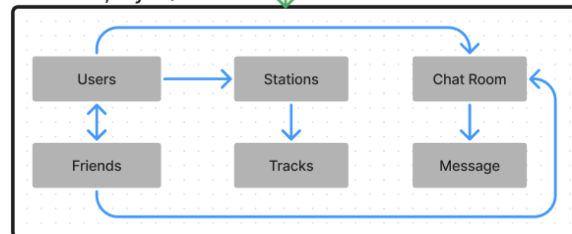
Back End



Legend



Database, MySQL



Frontend

- LoginScreen
 - EditText: Username
 - EditText: Password
 - Button: Register
 - When clicked, it sends the username & password to the backend to be stored in the database, therefore creating an account
 - Button: Login
 - When clicked, it sends a request to the backend asking if those credentials exist, if they do then they are let into the app, if they don't then the user must register first.
- ScreenSetter
 - EditText: Song Name
 - Slider: Tempo
 - Slider: Popularity
 - Slider: Volume
 - Button: SetSeed
 - When pressed it sends the song name that was typed in to the backend to seed the song recommendations
- MusicSwipe
 - ImageView: Album Cover
 - TextView: SongName
 - TextView: ArtistName
 - This class just allows the user to swipe through all of the recommendations that spotify returns based on the seed they set
- ProfileScreen
 - Button: Remove Friend
 - Button: Add Friend
 - Button: Open Chat
 - Button: Set Bio
 - TextView: Username
 - ImageView: Profile Picture
 - EditText: Set Bio
 - Sends the text as a json string request to the backend to persist their bio to what they types
 - TextView: Bio
 - Displays the Bio they set
 - This class allows the user to interact with friends (adding, removing, chatting)
- FriendProfileScreen
 - TextView: Username
 - TextView: Bio
 - ImageView: Profile Picture
 - This class just displays the profile of the user (not current user)
- ChatScreen
 - EditText: friendToChat
 - Button: Connect

- When this button is pressed it sends the username that was typed in the friendToChat editText and creates a session with that user to be able to chat with them
- EditText: Message
- Button: SendText
 - Sends what was typed in the message to the other user to chat

Backend

- Http requests received from frontend through UserController class.
 - Get, Post, Put, & delete requests used to allow for front end to edit, and read from, database.
 - Post requests used for user creation and station creation as well as ChatRoom creation.
 - Put requests used to edit song seeds and seed settings for User Stations as well as user info (friends, bio etc.).
 - Delete requests used User deletion
- Controllers
 - AuthController
 - Used to authenticate users with spotify accounts.
 - UserController
 - Primary Controller used for the editing of all things associated with Users: Stations, Tracks, ChatRooms.
 - SpotifyController
 - Used to send Get requests to spotify to send and receive song data.
- Repositories for Users, Stations, Tracks, Messages, ChatRooms used to persist data from and about users on mySql server.
 - User Class stores information about User (name, password, friends, stations, etc.)
 - Contains an instance of Station which is edited through User reference.
 - Station Class stores seed information (Track, Popularity, Tempo, Volume)
 - Contains an instance of Track which is edited through Station reference.
 - Track Class contains information about a parsed track received from Spotify (id, image, artist etc.).
 - Message Class contains information about the sender of a message and its contents.
 - ChatRoom Class contains information about two users chatting history (list of messages and participants).
 - Contains a List of message objects.

