

Talk to me backend was developed with Flask version 1.1.2 and SocketIO version 2.3.0.

Here deals with the matchmaking and socket events for users to connect. For matchmaking, it uses MongoDB Atlas.

Functions in app.py

`change_vent_listen_to_talk(query):`

- Helper function to `check_queue()`.
- Takes in a query and changes all matching documents' queueType that have their wait time greater than 'x' seconds to 'talk'

`check_queue()`

- Deals with the matchmaking for Vent/Listen and Talk
- Continuously runs every 3 seconds using the APScheduler with `id=check_is_queue_ready`
- Users with 'talk' queue type gets matched with themselves
- Vent/Listen queue, tries to match with each other
 - If there is no match after a certain amount of time, will change their type to talk and get matched with talk

`delete_user_from_db(userObj)`

- Deletes document with matching userObj from database

`find_time_difference(userTime)`

- Helper function to `change_vent_listen_to_talk(query)`
- Determines whether a user has passed 'x' seconds
- Returns true/false

`handleMessage(jsonObj)`

- Makes sure the user sending a message exists
- Stores message in message collection and emits the 'send_message_to_client' using socket.io

`handle_message_like(body)`

- Makes sure the user exists
- Changes message into a liked state in message collection
- Emits 'message_liked' using socket.io

match_making(userIDs)

- Changes two users (userIDs) queueType to outQueue
- Creates a document in room DB for these two users with their corresponding room

notify_queue_complete(user_id)

- Emits socket event that their queue is complete for both users in user_id

request_queue(body)

- When user clicks on their preferred queue, they will be placed in that queue

user_auth()

- Generates an auth for each user that joins goes to talk-to-me main page

user_disconnect()

- If user disconnects/closes the tab, they get removed from db and queues

user_join_room(secret)

- Have the user with the provided secret join their room

user_leave_room(secret)

- Have the user with the provided secret leave their room

user_sid_assoc(secret)

- Updates user's sid by looking for 'hello'