

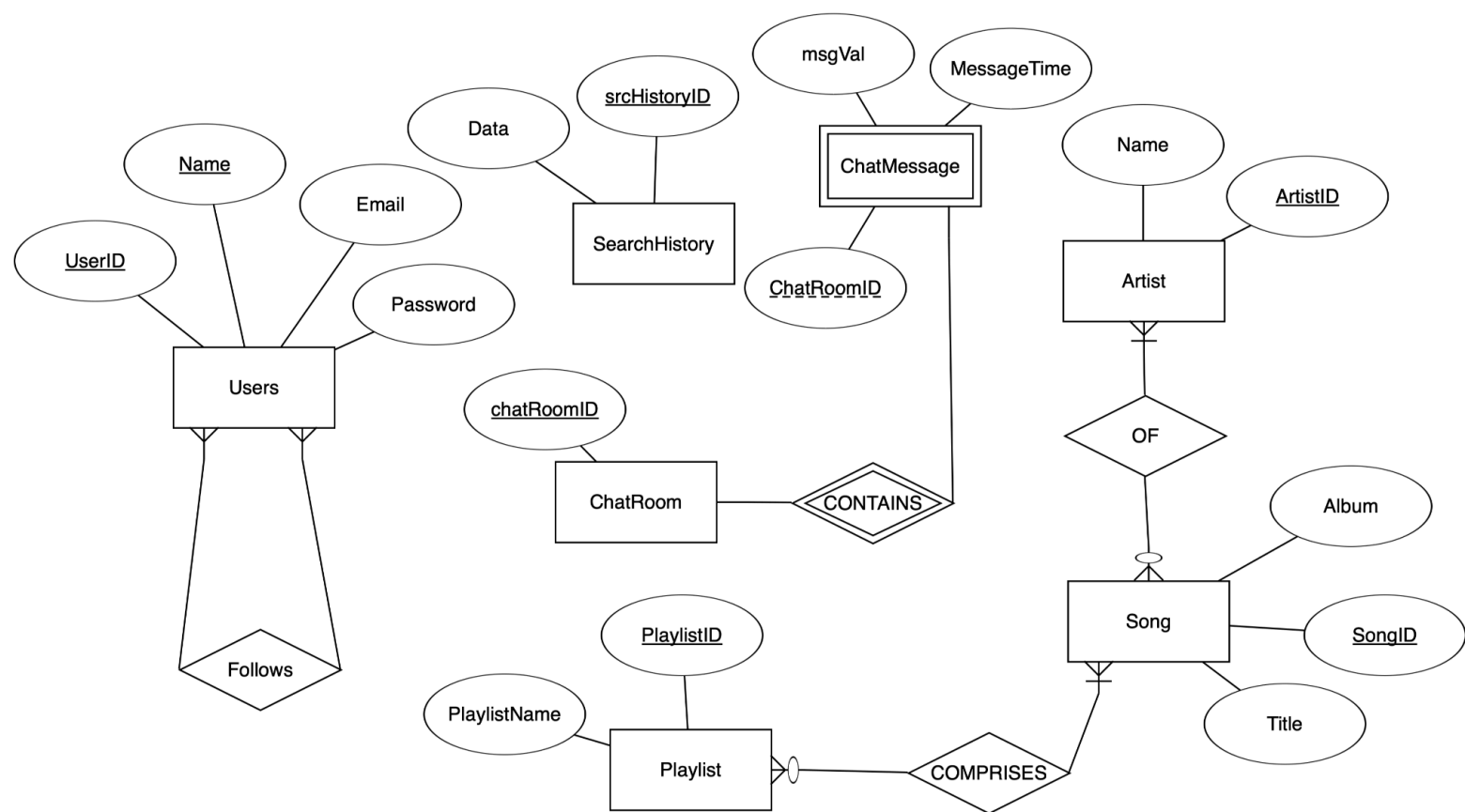
Introduction

A music streaming platform with integration of a chatting system. It is a mix between a streaming platform like Spotify and a chatting software like Discord.

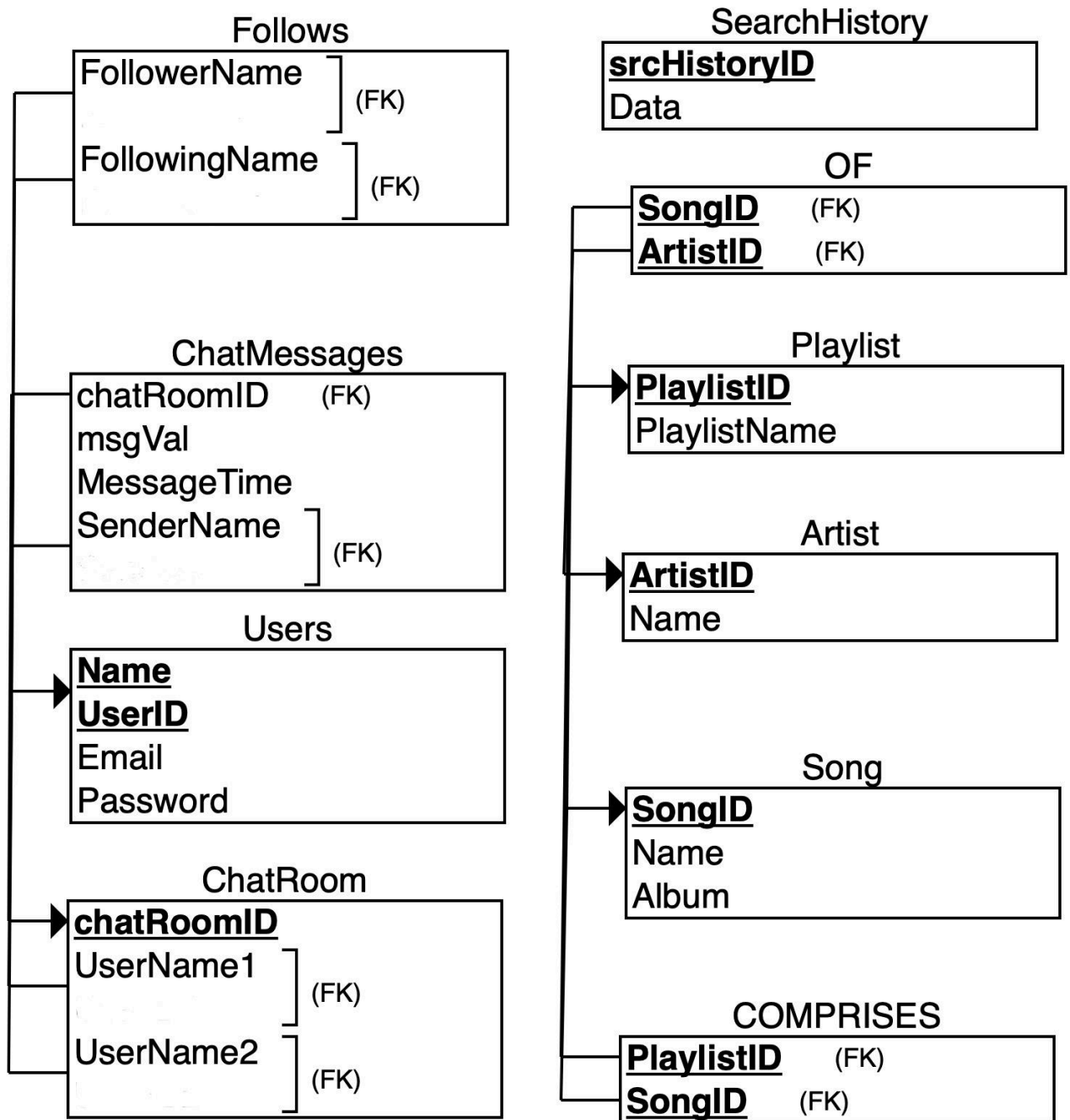
Project Features

- **User Management:**
 - **Sign up, login, and logout functionalities for users.**
- **Song Management:**
 - **Upload songs with details like title, artist, and album.**
 - **Search songs and artists by title or name.**
- **Playlist Management:**
 - **Create personalized playlists.**
 - **Add songs to playlists and play them.**
- **Social Features:**
 - **Follow other users.**
- **Chat Room:**
 - **Create chat rooms and chat with other users in real-time.**
- **Admin Panel:**
 - **View and delete data from any entity for administrative purposes.**

ER/EER Diagram

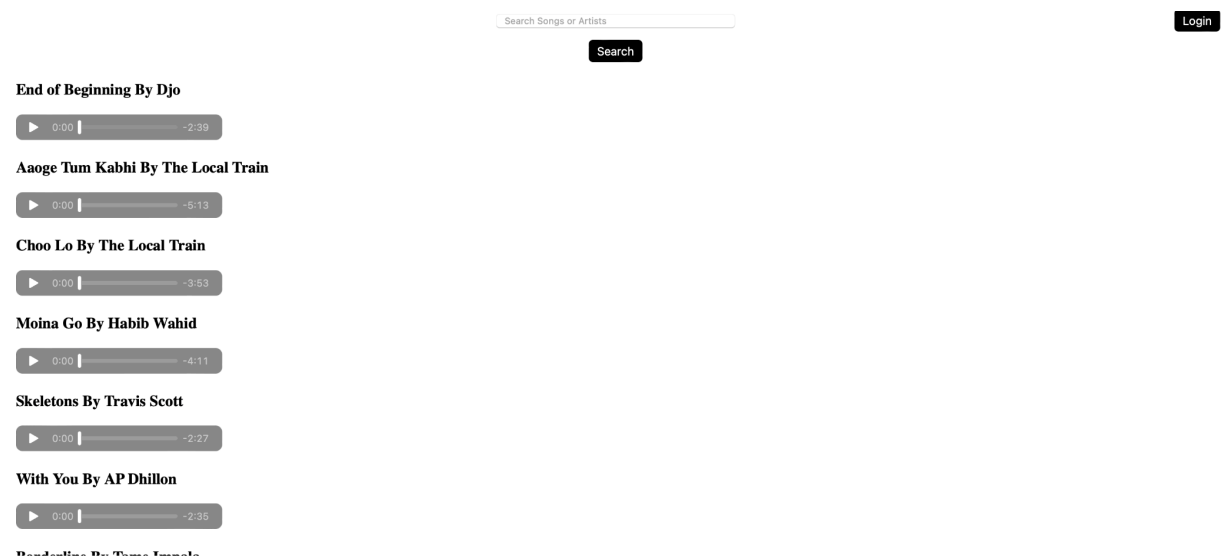


Schema Diagram

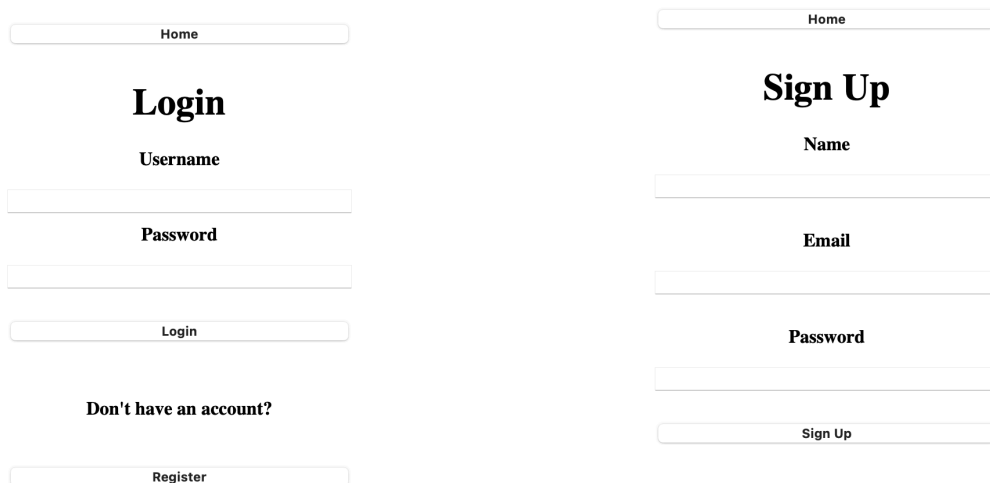


Frontend Development

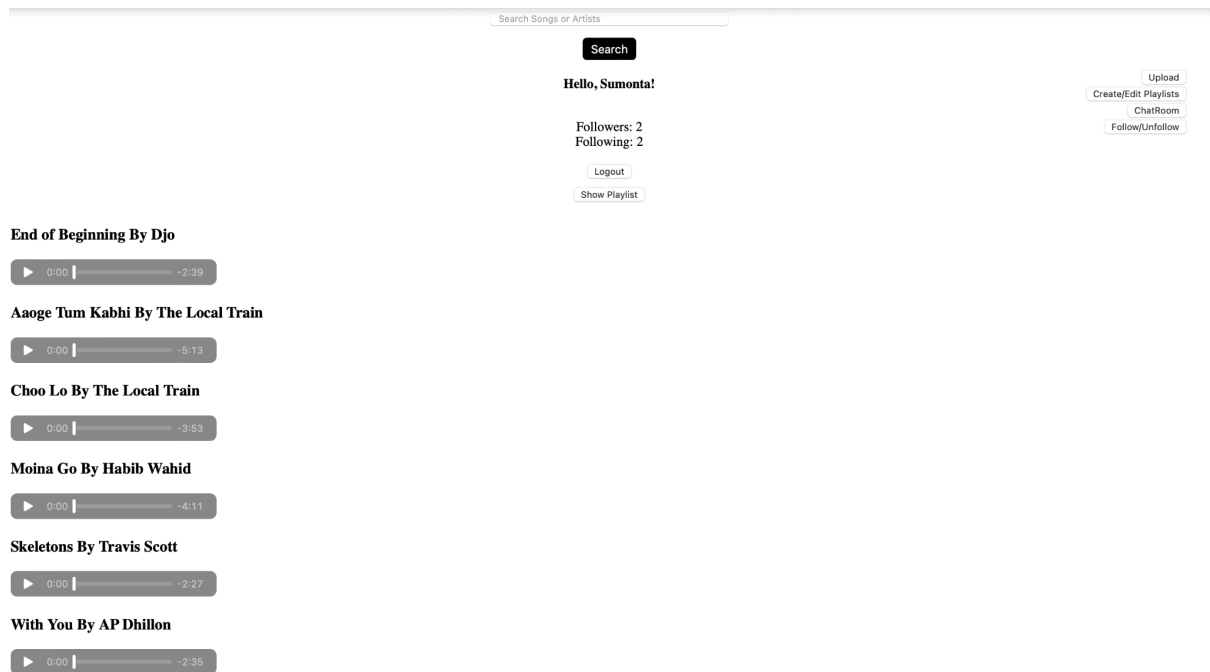
The home page has all the songs in the database. Alongside, there is also a search bar at the top enabling the search of artists/songs within the database and a login button at the top right.



The login page lets the user go back to the home page, login and if they do not have an account, register for one. The sign up page lets the user make a new account.



When logged in there are song upload functionality, create/edit, show playlist options, the total followers & following of the user, follow/unfollow users, open/create chat rooms and logout.



The search button allows the user to search for songs/artists and all the similar songs will show up in the page.

Home

Search Results for: 'The local'

Aaoge Tum Kabhi By The Local Train



Choo Lo By The Local Train



The upload page allows the user to select a .mp3 file, title, album and artist for a song and uploads that to the database.

[Home](#)

Select Song

no file selected

Title

Album

Artist

The create/edit playlist lets users create a new playlist (playlist name must be unique). Add Songs enables users to see a list of all the songs and playlists in the database and add the songs to the playlist based on their input.

[Home](#)

Create Playlist

Playlist Name:

[Home](#)

Add Song to Playlist:

Playlist Name:

Song ID:

List of Songs:

End of Beginning (SongID: 1)
Aaoge Tum Kabhi (SongID: 2)
Choo Lo (SongID: 5)
Moina Go (SongID: 6)
Skeletons (SongID: 21)
With You (SongID: 22)
Borderline (SongID: 23)
After Hours (SongID: 24)
I Wanna Be Yours (SongID: 25)
Summertime Sadness (SongID: 26)

List of Playlists:

Playlist Name: Vibes

The show playlists page, shows a list of all the playlist and allows the user to search for a playlist, all the songs in the playlist will show up once searched.

[Home](#)

Enter the name of the playlist: [Submit](#)

Playlists:

Playlist Name: Vibes

Playlist Name: Chill

Playlist Name: jamalfies

Songs in playlist: Vibes

Aaoge Tum Kabhi By The Local Train



Choo Lo By The Local Train



Moina Go By Habib Wahid



Chatroom page allows users to open a new chat room with other users or join an existing chat room (only if they are authorized to do so).

[Home](#)

Chatrooms

Create New Chatroom

Username of the other user: [Create Chatroom](#)

Existing Chatrooms

[Chatroom ID: 1 Users: Sumonta & tim](#)

[Chatroom ID: 5 Users: virat & Sumonta](#)

Inside of the chat rooms, the users can send/receive texts from the other users, it shows the chat room ID and the users of the chat room. The page needs to be refreshed to get new messages (as Database does not allow real time updating).

Home

Chatroom (ID: 1)

Users in this chatroom:
Sumonta
tim

Refresh the page to see new messages

Sumonta: Hello
Sumonta: yoo
tim: Hi
tim: wsg bro
Sumonta: chillin
tim: peace
Sumonta: ight g
Sumonta: bruh
tim: All good
tim: giv me vbuck
Sumonta: no
Sumonta: hello
tim: hiii

Your message:

Send

The follow/unfollow page shows all the available users and gives the user an option to follow/unfollow them.

Home

Follow/Unfollow

Logged in as: **Sumonta**

Available Users:

tim

virat

In the login page if the admin logs in, it takes the admin to the special access page where the admin can look up all the values in the entity and choose to delete any data entries.

[Home](#)

Admin Delete Access

Available Entities:

Artist, ChatMessages, ChatRoom, Follow, Playlist, SearchHistory, Song, Users

Enter Entity (Table) Name:

[Submit](#)

Data from Users:

| Key | Other Columns | | | | |
|-----|---------------|---------|-------------------|---------|------------------------|
| 1 | 1 | Sumonta | sumonta@gmail.com | Sumonta | Delete |
| 2 | 2 | tim | david@gmail.com | david | Delete |
| 3 | 3 | virat | virat@gmail.com | kohli | Delete |

Backend Development

Created the database with entities based on the relational schema and added all the appropriate attributes.

| Table | Action |
|--|-------------------|
| <input type="checkbox"/> Artist | Browse Structure |
| <input type="checkbox"/> ChatMessage | Browse Structure |
| <input type="checkbox"/> ChatRoom | Browse Structure |
| <input type="checkbox"/> COMPRISES | Browse Structure |
| <input type="checkbox"/> Follow | Browse Structure |
| <input type="checkbox"/> OF | Browse Structure |
| <input type="checkbox"/> Playlist | Browse Structure |
| <input type="checkbox"/> SearchHistory | Browse Structure |
| <input type="checkbox"/> Song | Browse Structure |
| <input type="checkbox"/> Users | Browse Structure |
| 10 tables | Sum |

When the user signs up, the name, email and password gets saved to the database in the Users entity.

| UserID | Name | email | password |
|--------|---------|-------------------|----------|
| 1 | Sumonta | sumonta@gmail.com | Sumonta |
| 2 | tim | david@gmail.com | david |
| 3 | virat | virat@gmail.com | kohli |

The uploaded songs get uploaded to “uploads/” directory and their title, artist and album names get added to the Song entity.

| SongID | Name | Artist | Album |
|--------|--------------------|-----------------|--------------------|
| 1 | End of Beginning | Djo | End of Beginning |
| 2 | Aaoge Tum Kabhi | The Local Train | Aaoge Tum Kabhi |
| 5 | Choo Lo | The Local Train | Choo Lo |
| 6 | Moina Go | Habib Wahid | Moina Go |
| 21 | Skeletons | Travis Scott | Skeletons |
| 22 | With You | AP Dhillon | With You |
| 23 | Borderline | Tame Impala | Borderline |
| 24 | After Hours | The Weeknd | After Hours |
| 25 | I Wanna Be Yours | Arctic Monkeys | I Wanna Be Yours |
| 26 | Summertime Sadness | Lana Del Rey | Summertime Sadness |

All the searches get stored in the SearchHistory entity.

| srcHistoryID | Data |
|--------------|-------|
| 1 | local |
| 2 | l |
| 3 | a |
| 4 | hab |
| 5 | lo |
| 6 | local |
| 7 | lo |
| 8 | local |
| 9 | wahid |
| 10 | go |

Whenever a new playlist is created, the name of the playlist along with the auto generated PlaylistID gets stored in the Playlist entity. The songs in the playlist get stored through the COMPRISES entity.

| PlaylistID | PlaylistName | PlaylistID | SongID |
|------------|--------------|------------|--------|
| 1 | Vibes | 1 | 2 |
| | | 1 | 5 |
| | | 1 | 6 |
| 2 | Chill | 2 | 1 |
| | | 2 | 23 |
| | | 2 | 24 |
| | | 2 | 25 |

The artist entity stores the ArtistID and their names and the songs by the artists are stored through the OF entity.

| ArtistID | Name | SongID | ArtistID |
|----------|-----------------|--------|----------|
| 1 | Djo | 1 | 1 |
| 2 | The Local Train | 2 | 2 |
| 4 | Habib Wahid | 3 | 3 |
| 10 | Travis Scott | 4 | 3 |
| 11 | AP Dhillon | 5 | 2 |
| 12 | Tame Impala | 6 | 4 |
| 13 | The Weeknd | 7 | 5 |
| 14 | Artic Monkeys | | |
| 15 | Lana Del Rey | | |

The Follow entity stores the names of the follower and who they are following, it is also used to count the Follower and Following count in the Home page.

| followID | FollowerName | FollowingName |
|----------|--------------|---------------|
| 5 | tim | Sumonta |
| 6 | Sumonta | tim |
| 7 | virat | Sumonta |
| 8 | virat | tim |

The ChatRoom keeps track of all the rooms created and the ChatMessages keeps track of all the messages coming to that room. The messages are ordered ascendingly in respect to the MessageTime.

| chatRoomID | Username1 | Username2 |
|------------|-----------|-----------|
| 1 | Sumonta | tim |
| 5 | virat | Sumonta |

| ChatRoomID | SenderName | msgVal | MessageTime |
|------------|------------|----------|---------------------|
| 1 | Sumonta | Hello | 2024-04-26 00:00:00 |
| 1 | Sumonta | yoo | 2024-04-27 20:32:24 |
| 1 | tim | Hi | 2024-04-27 20:32:37 |
| 1 | tim | wsg bro | 2024-04-27 20:33:21 |
| 1 | Sumonta | chillin | 2024-04-27 20:33:32 |
| 1 | tim | peace | 2024-04-27 20:34:50 |
| 1 | Sumonta | ight g | 2024-04-27 20:57:24 |
| 1 | Sumonta | bruh | 2024-04-27 21:03:26 |
| 1 | tim | All good | 2024-04-27 21:16:37 |

Source Code Repository

https://drive.google.com/drive/folders/1ib_mBX5fFSGm_BI1xsoOcBuQRyJDPk5J?usp=share_link

Conclusion

This project was done using HTML, MySQL, PHP. The Chatting feature is not in real time as DBMS does not support that, it was done to simplify the project, same applies for password hashing.

References

- <https://www.javatpoint.com/>
- <https://www.w3schools.com/>