CHITKARA
UNIVERSITY
HIMACHAL PRADESH - NAAC Accredited

PROJECT REPORT

IN

“COMPUTER SCIENCE ENGINEERING”

ON

“RESOUND MUSIC PLAYER”

SUBMITTED IN PARTIAL FULFILLMENT OF THE DEGREE

OF

BE (CSE)

Under the Guidance of:
Name: Dr. Abhishek Thakur
Designation: Mentor
Department: Computer Science Engineering

Submitted By:
Madhav Goyal (2011981262)
Akhilesh Thakur (2111981020)
Akshita (2111981026)
Anjori (2111981033)

Chitkara University, Himachal Pradesh, India

CONTENTS

Title **Page No.**

DECLARATION

We hereby declare that the project work titled, "**Resound | An Interactive Music Sharing and Streaming Platform**" submitted as part of Bachelor's degree in CSE, at Chitkara University, Himachal Pradesh, is an authentic record of our own work carried out under the supervision of Dr. Abhishek Thakur.

Signature(s):

ACKNOWLEDGEMENT

We would like to express our deepest appreciation to our project supervisor, Dr. Abhishek Thakur, for his guidance, support, and expertise throughout the development of our music player. We are grateful for his invaluable insights and feedback that helped us refine our ideas and make critical decisions.

We would also like to thank our team members, Madhav Goyal, Akhilesh Thakur, Akshita and Anjori, for their hard work and dedication in bringing this project to fruition. Each team member brought unique skills and perspectives to the project, and their collaboration was essential to the success of the music player.

We extend our gratitude to the outside consultants who generously shared their expertise with us. Their insights and feedback helped us refine the functionality and usability of our music player. We would like to express our gratitude towards our parents member of CSE Department for their kind co-operation and encouragement which helps us in the completion of the project.

Lastly, we thank all of the users who provided valuable feedback that helped us improve the music player. We are proud of what we have accomplished and we hope that our music player will bring joy and entertainment to music lovers everywhere.

Thank you all for being a part of this journey and for making this music player project a success.

LIST OF FIGURES AND TABLES

PROJECT TITLE

Resound | An Interactive Music Sharing and Streaming Platform

Live Project Link: https://www.madhavgoyal.me/resound

Project Code Link: https://www.github.com/BartBruh/resound

INTRODUCTION

Welcome to the world of Music! Music is an integral part of our lives and has the power to evoke emotions, memories, and moods. In this project we design and develop a music player that goes beyond playing songs based on just the user's streaming history and also takes into consideration the user's real-time emotions and present them in an easy-to-use interface. With the integration of emotion detection technology, the music player will detect the user's emotional state and play songs accordingly. Our goal is to develop a music player that will not only play songs but also improve the user's mental well-being by playing songs that are appropriate for their mood. So, sit back, relax, and immerse yourself in the world of music with our music player. We are excited to embark on this journey of creating a music player that will enhance the user's music listening experience and contribute to their overall well-being.

Project Category:

In my view this project would fall under the category of web development, with a focus on front-end development. The front-end development aspect of the project would involve designing and developing the user interface of the website, including creating a navigation menu, designing the music player interface.

For the music playback functionality, the project could involve using an open-source music player library or creating a custom music player using HTML5 audio tag and JavaScript. The website could also include features such as searching for songs and displaying song lyrics. In this project, we need to design the user interface and user experience of the application, including features such as playlists, music library management, and playback controls.

ABSTRACT

The aim of this project is to develop a music player website that provides users with a comprehensive and user-friendly platform to listen to their favourite music. The website will be designed to allow users to access a vast collection of songs, while having advanced Emotion Detection features allowing the user to sit back and relax while the website plays songs according to their real-time mood. Our approach involves integrating cutting-edge AI, ML and audio technology with intuitive design principles to create a music player that meets the needs of music lovers.

The project provides seamless access to the music player, allowing users to control their music playback and access their music library on the go. Our project has the potential to improve the music listening experience for people around the world and set a new standard for music player design and functionality. To enhance the user experience, the website will include a search function that enables users to search for songs based on artists, genres, and keywords. The website will also feature a song recommendation system that suggests new songs based on the user's listening history.

The outcome of the project will be a functional music player website that provides an enjoyable and immersive music listening experience for users. The website will be a valuable resource for music lovers, allowing them to discover new music.

WORK DONE

Overview:

User Interface Design: This involves designing an intuitive and user-friendly interface for the music player that allows users to easily browse, search, and play their music.

Audio Playback Functionality: This involves implementing music playback controls, volume adjustments, and playlist management using an open-source music player library or custom music player

Music Library Management: This involves managing the user's music library, including adding/removing music, organizing music by artist/album/genre.

Audio Equalization: This involves adding equalization controls to the music player, which allows users to adjust the frequency response of the audio output to suit their preferences.

Front-end Development: This involves designing and developing the user interface and user experience of the website using HTML, CSS, and JavaScript. For a music player website, front-end development includes designing the music player interface, playlist management, and search functionality.

Testing and Optimization: This involves testing the website's functionality, performance, and security to ensure a seamless user experience. It also includes optimizing the website's design and functionality.

Purpose:

l. To allow users to play and listen to music.

2. Allow users to listen to music whenever and wherever they want.
3. To customize their listening experience to suit their preferences.
4. Easy access to a wide variety of music: A music player website provides users with easy access to a vast collection of songs, including popular music and niche genres.
5. Gives Convenience to the user.
6. User-friendly platform to listen to their favourite music and discover new songs.

Overall Description:

1. Product Perspective:

This software system is user friendly. The product has various features, such as the ability to manage a music library and adjust audio settings.

Some key aspects of the product perspective include:

Market analysis: Understanding the market for online music streaming services, including market size, competition, and user demographics.

Product features: Identifying the key features that make the music player website unique, such as personalized playlists, social sharing, and music recommendation algorithms.

User experience: Focusing on creating a seamless and enjoyable user experience that is easy to navigate and use.

Scalability: Ensuring that the website is scalable to handle increasing user traffic and a growing music library.

Business model: Identifying the revenue streams for the music player website, such as advertising, subscription fees, or a combination of both.

2. Product Function:

The product function of a music player is to allow users to play and listen to music. The core function of a music player is to read audio files, decode them, and play them.

Music library: The website should provide access to a vast collection of songs that users can search and browse.

Playlists: Users should be able to create customized playlists with their favourite songs and organize them in a way that makes sense to them.

Playback controls: The website should provide playback controls such as play, pause, skip, and volume control.

User accounts: Users should be able to create accounts to save their preferences, playlists, and listening history.

3. **Operating Environment:**

   This project works on the following:

   1. Operating System: Windows 7 and higher.
   2. Text-Editor: VS Code.
   3. Technologies used: HTML, CSS, JavaScript..

4. **Design and Implementation Constraints:**

   1. Platform Compatibility
   2. Audio Format Support
   3. Resource Limitations
   4. Licensing and copyright
   5. Storage Capacity

External Interface Requirements:

1. **User Interface:**
   - It will be easy to use and user friendly.
   - It has standard playback controls.
   - User-friendly platform to listen to their favourite music and discover new songs.
   - To customize their listening experience to suit their preferences.
   - Easy access to a wide variety of music: A music player website provides users with easy access to a vast collection of songs, including popular music and niche genres.

2. **Software Interface:**

   This whole project works on browsers like Chrome, Firefox etc. and is based on the technologies like HTML, CSS and JavaScript. It has the following features:

   - Navigation: The website should have a clear and easy-to-use navigation system that allows users to access different functions and features.

- Search: The website should have a search bar that allows users to search for songs, artists, albums, and playlists.
- Playlists: Users should be able to manage playlists easily.
- Playback controls: The website should provide clear and accessible playback controls, such as play, pause, skip, and volume control.
- User accounts: Users should be able to create and manage their accounts easily, with the ability to view their listening history, manage their preferences.
- Design: The website should have an attractive and visually appealing design that is consistent across all pages and functions.

Other Non-Functional Requirements:

1. **Performance Requirements**
   - It gives quick responses.
   - It is time saving.
   - It is easily accessible to the user.
   - It has Log In/Log Out option.

2. **Portability Requirements**

   Cross-browser compatibility: The website should be designed to work seamlessly on different web browsers, including Chrome, Firefox, Safari, and Edge.

   User accounts: Users should be able to create accounts to save their preferences, playlists, and listening history, allowing them to access their music from any device.

3. **Availability Requirements**

   This music player is up and running whenever needed. The users can access the website at all times, without any significant interruptions or downtime. By meeting these requirements, a music player website can provide a reliable and trustworthy user experience, leading to increased engagement and user retention.

4. **Scalability Requirements**

Scalability requirements refer to the ability of a music player website to handle increasing user traffic, a growing music library, and new features. A music player website should be scalable to ensure that it can meet the demands of its users, without sacrificing performance or user experience. It meets the needs for which it was build.

5. **Security Requirements**

This system provides basic security authentication. Security requirements for a music player website are critical to ensure that user data is protected, and the website is not vulnerable to attacks or breaches. Some key security requirements for a music player website include:

- Authentication and authorization: The website should have secure user authentication and authorization mechanisms, ensuring that only authorized users can access the website and their data.
- Data encryption: The website should use encryption to protect user data, such as passwords, credit card information, and personal information.
- Secure communication: The website should use secure communication protocols, such as HTTPS, to protect user data in transit.
- Access controls: The website should have appropriate access controls to limit user access to sensitive data and functionality.

6. **Safety Requirements**

Safety requirements for a music player website typically focus on protecting users from harm while using the website. It keeps the records. Also, it is safe from various attacks.

User privacy: The website should have privacy controls and data protection measures in place to ensure that users are not at risk of identity theft, fraud, or other types of cybercrime.

7. Usability Requirements

- Intuitive user interface: The website should have a clear and intuitive user interface that allows users to easily navigate the website and find the content they are looking for.
- Simple controls: The website should have simple controls that are easy to understand and use, such as play/pause, skip, volume, and playlist controls.
- Search functionality: The website should have a robust search function that allows users to easily find specific songs, albums, or artists.

8. Reliability Requirements

Reliability requirements for a music player website focus on ensuring that the website operates reliably and consistently, without downtime or errors. Some key reliability requirements for a music player website include:

- Uptime: The website should be available and accessible to users at all times, without extended periods of downtime. It is available to the user all the time.
- Performance: The website should perform well and respond quickly, without lag or delays, to user interactions such as play/pause, skip, and playlist management.

9. Supportability Requirements

It is supported by all the browsers like chrome, Firefox etc.

10. Efficiency Requirements

It works with good efficiency.

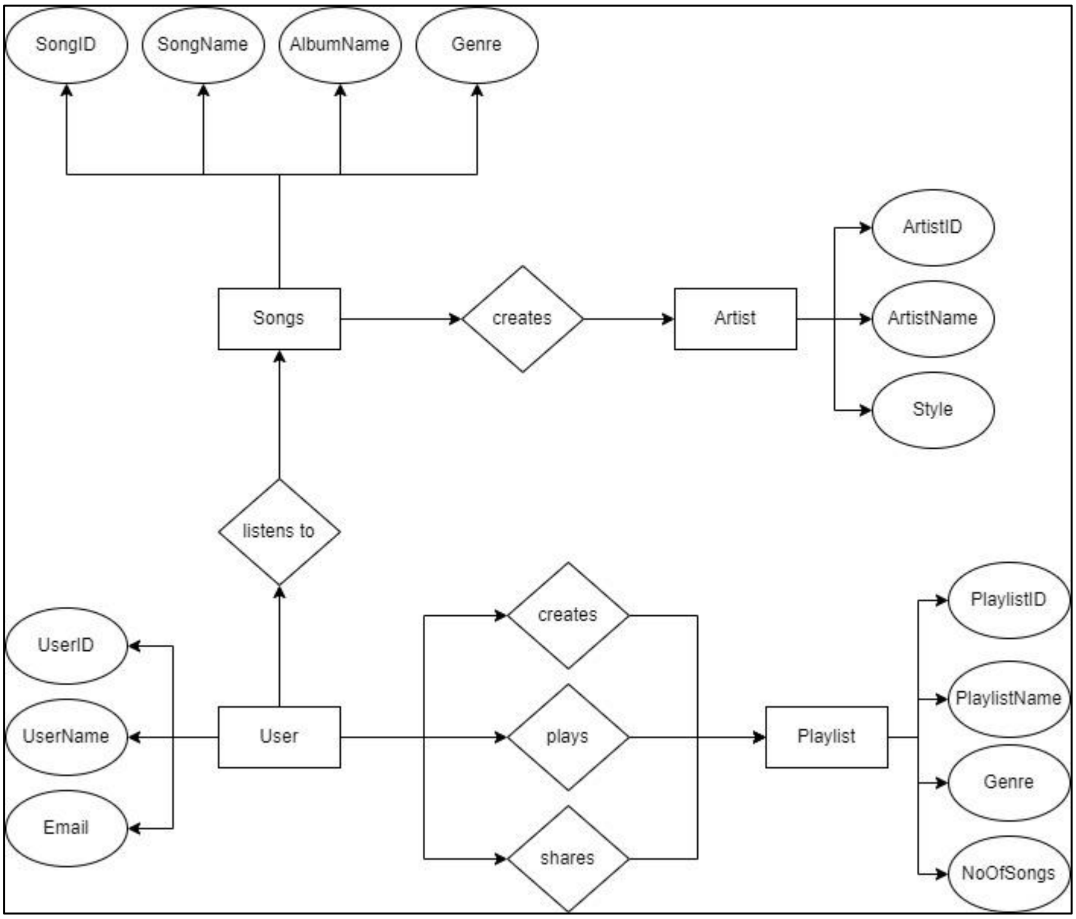
11. Interface Requirements

- Navigation: The website should have a clear and easy-to-use navigation system that allows users to access different functions and features.
- Search: The website should have a search bar that allows users to search for songs, artists, albums, and playlists.

- Playlists: Users should be able to manage playlists easily.
- Playback controls: The website should provide clear and accessible playback controls, such as play, pause, skip, and volume control.
- User accounts: Users should be able to create and manage their accounts easily, with the ability to view their listening history, manage their preferences.

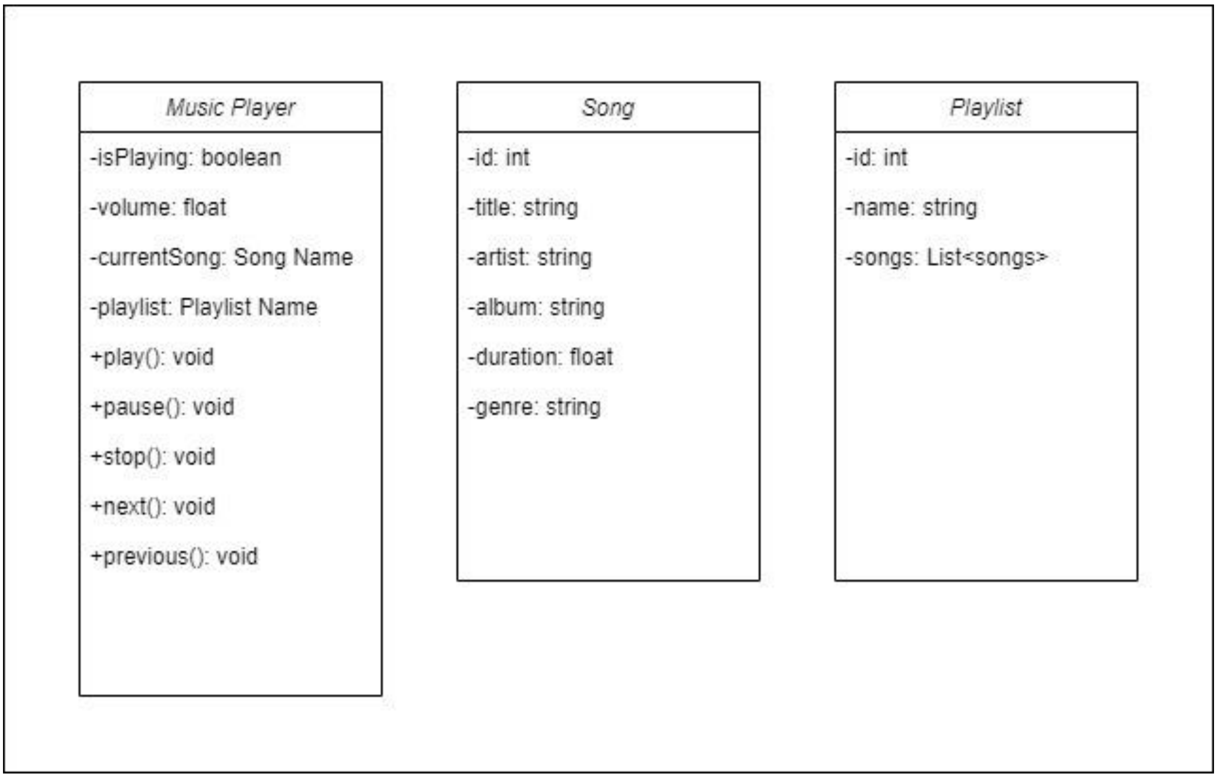
DIAGRAMS:

1. ER-Diagram:



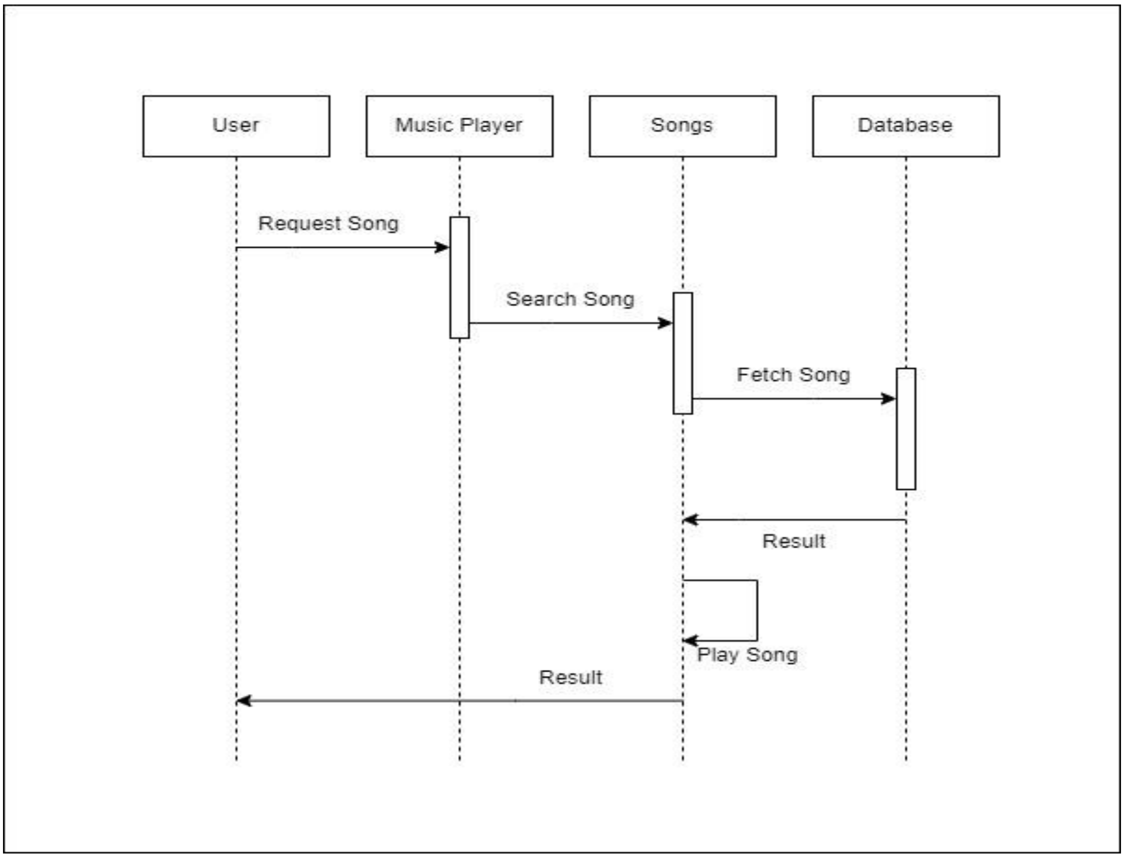
- An ER diagram of a music player includes entities such as Songs, Albums, Artists, and Playlists, each with their relevant attributes.
- These entities are related to one another through relationships such as "belongs to," "performed by," "contains," and "has," with varying cardinalities depending on the specific requirements of the music player.
- Attributes associated with these entities could include unique IDs and user preferences.
- Additionally, other considerations such as users, subscription plans, and social features could also be included in the diagram to reflect the complexity of the music player.

2. Class Diagram:



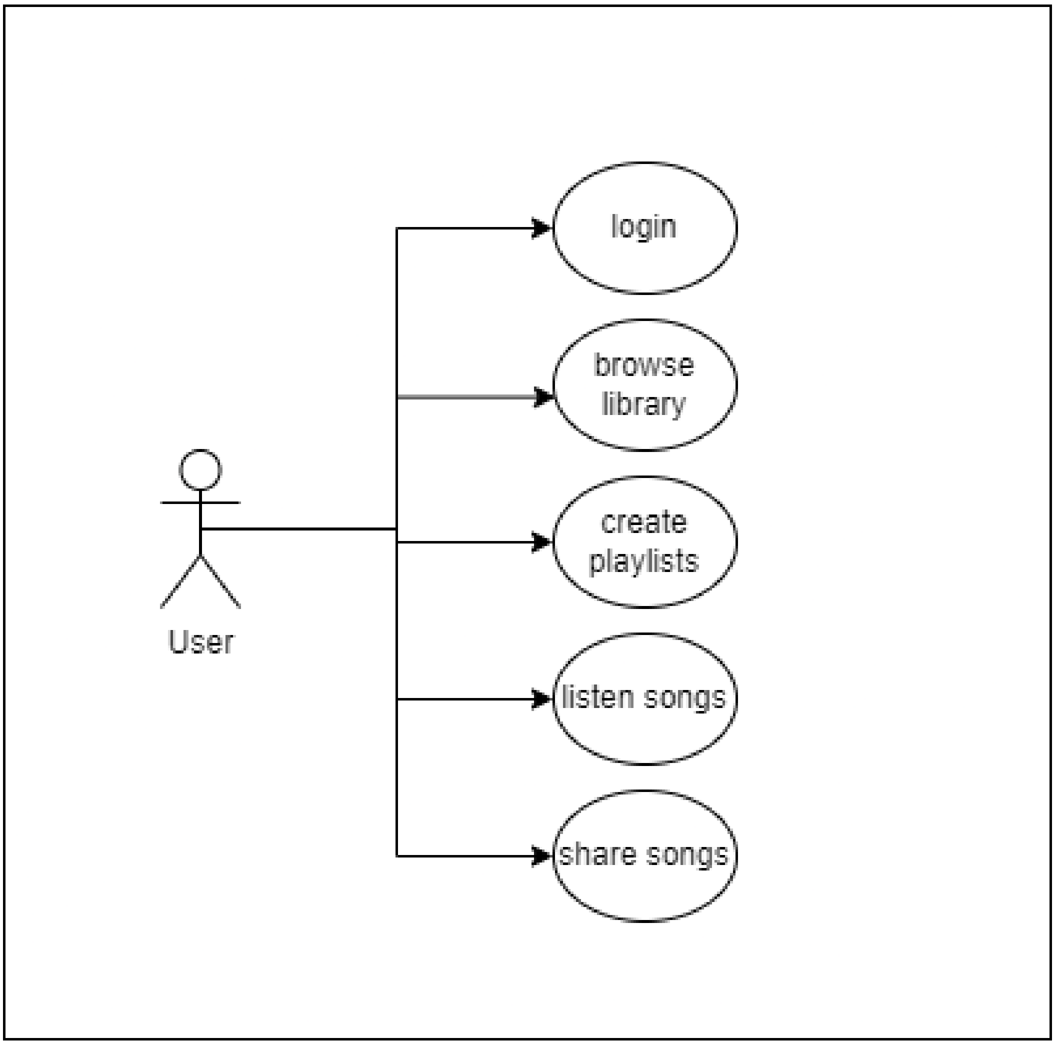
- A class diagram for a music player includes classes such as Music Player, Song, and Playlist, each with their relevant attributes and methods.
- Inheritance are used to model the relationships between these classes, with a Playlist composed of multiple Song objects.
- Encapsulation is used to restrict access to certain attributes and methods of each class.
- Other classes such as User, Subscription, and SocialFeatures are also be included in the diagram to reflect additional requirements.

3. Sequence Diagram



- A sequence diagram for a music player depicts actors such as the User, the Music Player application, Songs, and Database, with messages exchanged between these actors.
- The diagram also shows the timing and ordering of these messages, as well as any conditions or loops that could affect the sequence.
- Other actors and messages are included in the diagram to reflect the complexity of the system.

4. Use Case Diagram:



- A use case diagram for a music player depicts actors such as the user, the music player application, and external systems or APIs, with use cases related to these actors through relationships such as "extends" and "includes".
- The use cases include actions such as searching for songs, playing songs, creating and modifying playlists, and accessing external data sources.
- Depending on the specific requirements of the music player, other use cases and actors are included in the diagram to reflect the complexity of the system.

CONCLUSION AND FUTURE SCOPE

CONCLUSION:

This website will be a new hub for music lovers and a creative social media platform for music producers.

FUTURE SCOPE:

Integration of Social and Collaborative Features: Music is a highly social activity, and users often enjoy sharing their favourite songs and playlists with others. Therefore, future versions of the music player could integrate social and collaborative features such as the ability to create and share playlists with friends, follow other users and discover new music, or even collaborate with other users on a shared playlist.

Integration of Artificial Intelligence and Machine Learning: With the increasing sophistication of AI and machine learning technologies, music players could leverage these capabilities to provide more personalized and intelligent music recommendations, adapt to user preferences over time, and even generate new music based on user inputs.

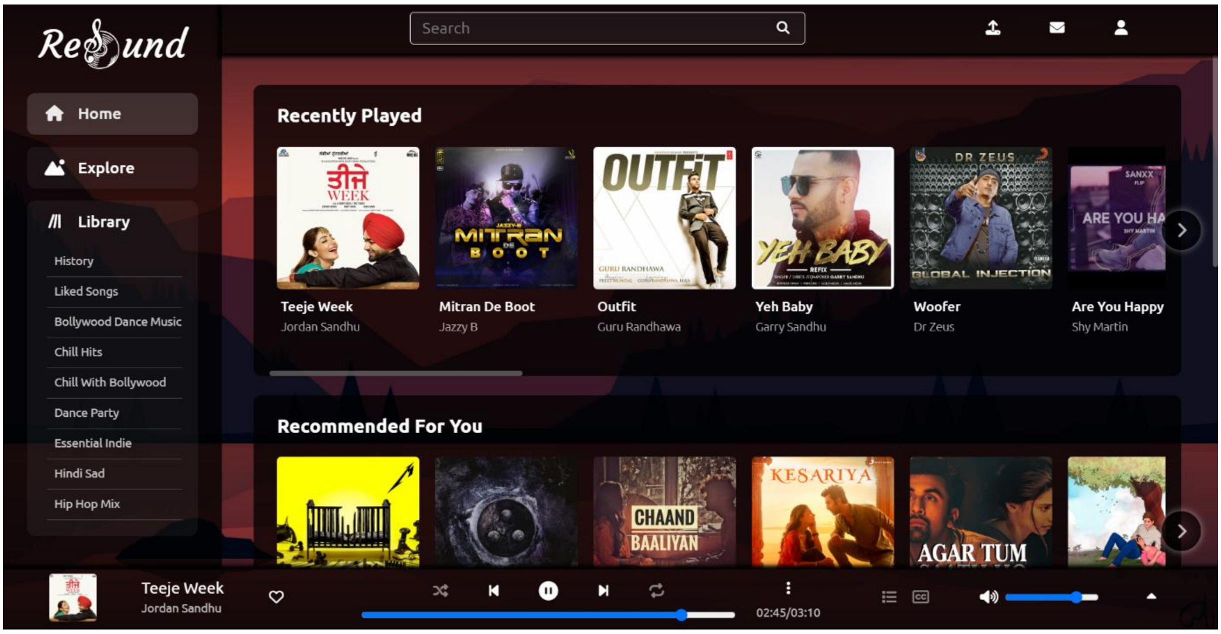
Augmented Reality and Virtual Reality Integration: Augmented reality and virtual reality technologies could be integrated into music players to create immersive and interactive music experiences. For example, users could experience concerts or live performances from the comfort of their homes, or even create their own virtual music environments.

REFERENCES:

1. https://www.w3schools.com/
2. https://developer.mozilla.org/
3. https://www.geeksforgeeks.org/
4. https://stackoverflow.com/
5. https://github.com/justadudewhohacks/face-api.js
6. https://www.youtube.com/watch?v=CVClHLwv-4I&ab_channel=WebDevSimplified

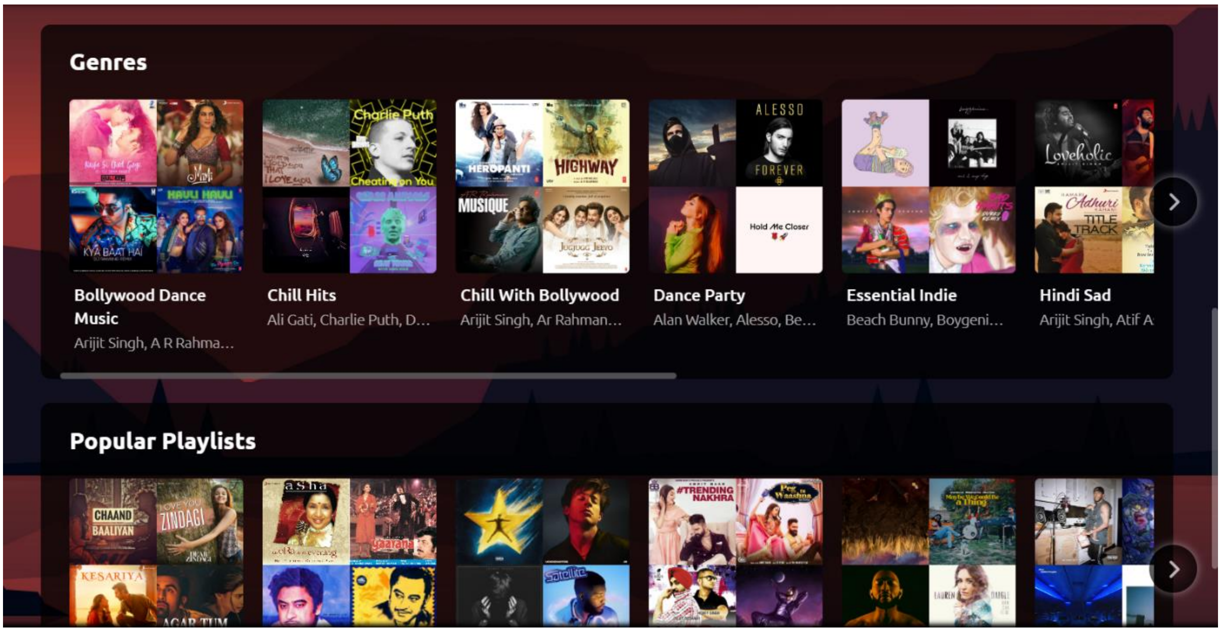
SNAPSHOTS

The Home Tab opens on website load and looks like this.



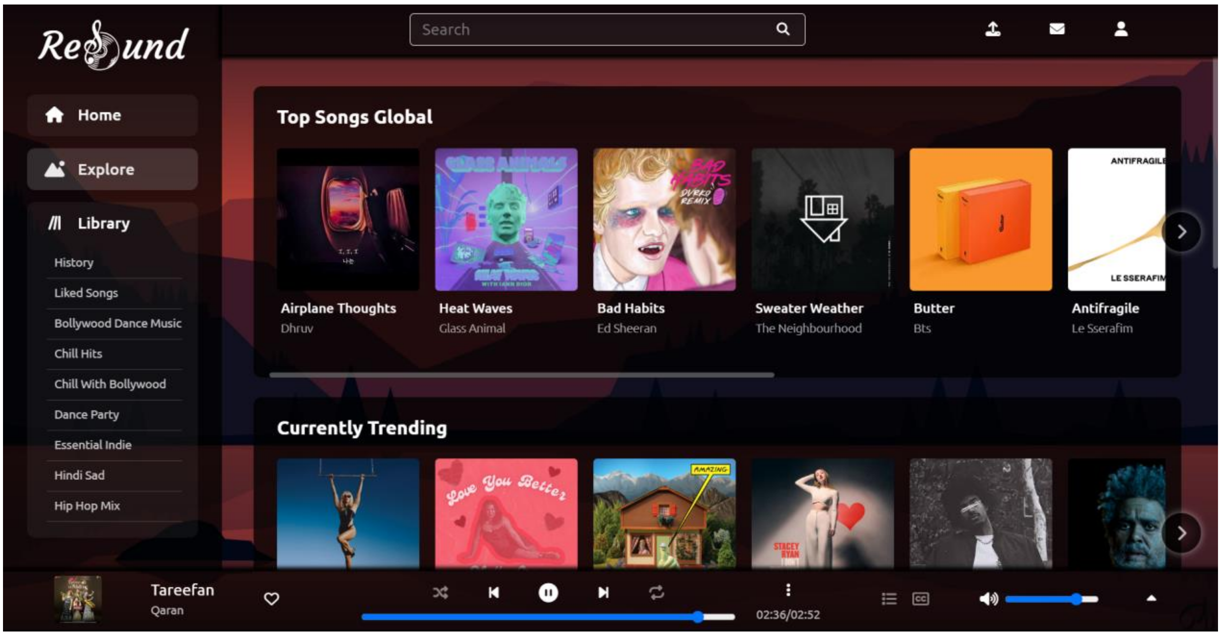
- The user’s recently played songs and playlists, recommended songs, popular songs and genres are found in the Home Tab.
- There is a Side Navbar to switch between Home, Explore and Library Tab.
- The Header on top has the Search Bar for the user to search songs and artists on the platform, and it also has an Emotion Detection Section and Profile button on the right to Login/Sign Up or Logout.
- Clicking any song plays the song and adds the song to the recently played songs.
- Clicking any playlist opens the Playlist View and gives options to Play the playlist, Shuffle Play, Add to Library, and Add to Queue.

Genres and Popular Playlists are found in the Home Tab.



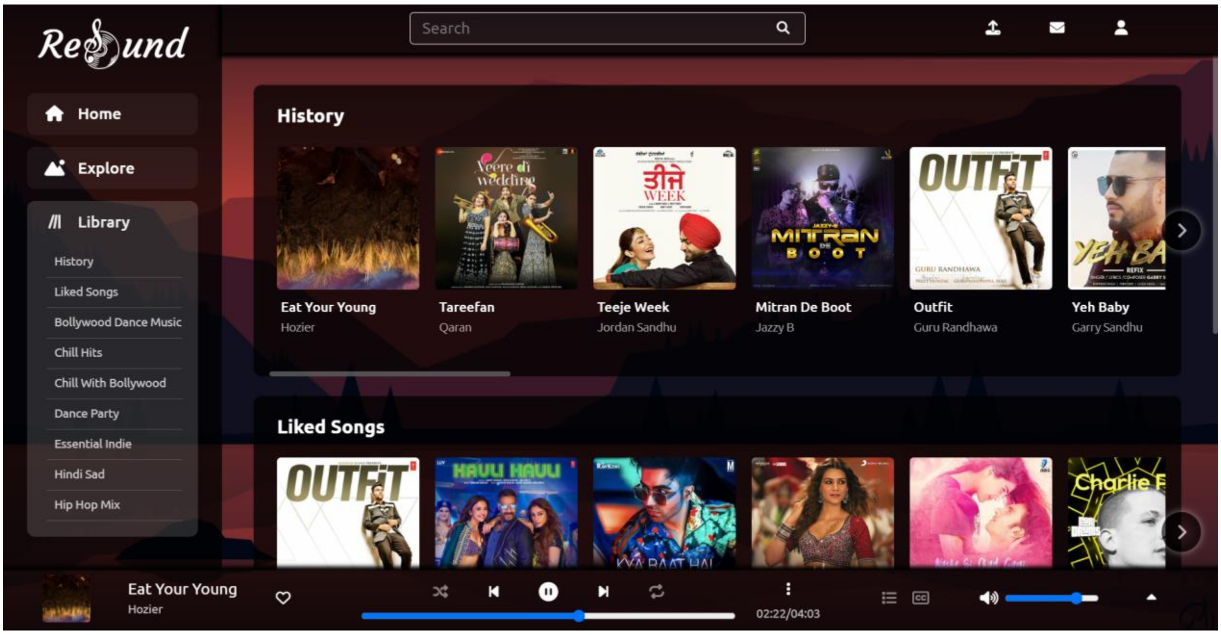
- You can also find multiple songs based on your preferred genre and various popular playlists in the Home Tab.
- You can click on the scroll buttons on the left and right of any category to scroll its content horizontally.
- Clicking the playlist cover opens the Playlist View, however, clicking the Play Icon Overlay on the cover directly plays the playlist from the Content Window.

The Explore Tab looks like this.



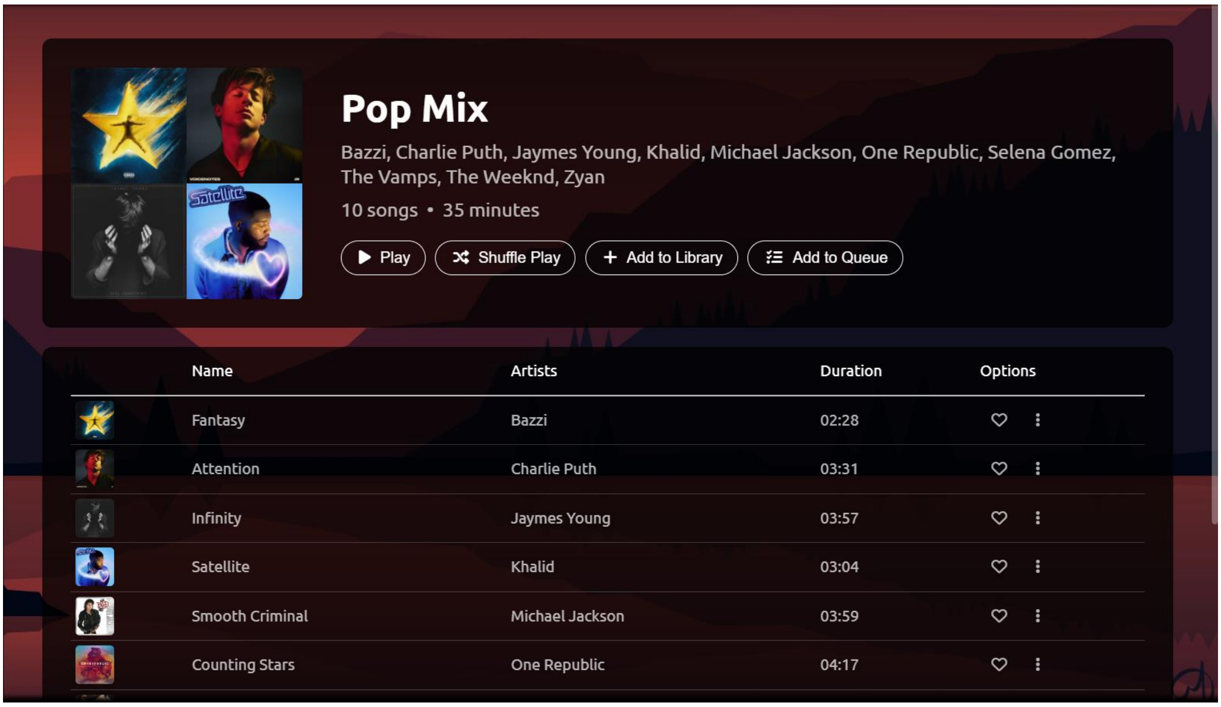
- Top songs on Global charts, Currently Trending Songs and Songs from Popular Artists are displayed in the Explore Tab.
- You can play songs, view playlists and scroll content in the same way as Home Tab on all tabs.
- The content on Explore Tab regular updates with time based on the user’s preferences and their listening history.

The Library Tab looks like this.



- The user's Listening History, Liked Songs and Saved Playlists are found in the Library Tab.
- As soon as the user plays a new song, the history gets updated and is displayed in the Library Tab.
- Same goes for Liked Songs i.e. as soon as the user likes or unlikes a song it gets added to the Liked Songs Playlist and can be viewed in the Library Tab.

Clicking any playlist opens the playlist view, which looks like this.



Pop Mix

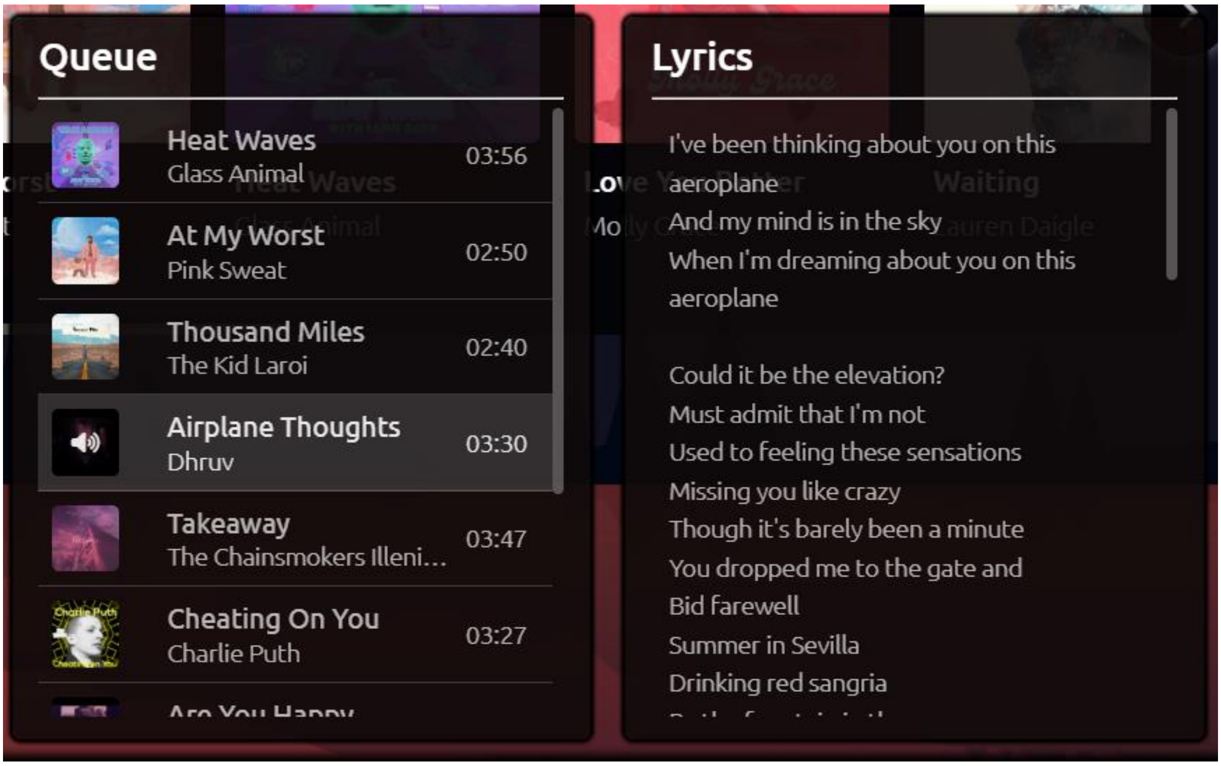
Bazzi, Charlie Puth, Jaymes Young, Khalid, Michael Jackson, One Republic, Selena Gomez, The Vamps, The Weeknd, Zyan

10 songs • 35 minutes

Name	Artists	Duration	Options
Fantasy	Bazzi	02:28	♡ ⋮
Attention	Charlie Puth	03:31	♡ ⋮
Infinity	Jaymes Young	03:57	♡ ⋮
Satellite	Khalid	03:04	♡ ⋮
Smooth Criminal	Michael Jackson	03:59	♡ ⋮
Counting Stars	One Republic	04:17	♡ ⋮

- This playlist view displays the Song Names along with their Covers, Artist Names, and Song Duration in the playlist contents.
- Options to Play the playlist, Shuffle Play, and Add to/Remove from Library are also available.
- The user can click on any song to play it, or like it to add it to their liked songs playlist or unlike to remove from the same.

Queue and Lyrics Floating Overlays show the queue and lyrics.



Queue

Heat Waves	Glass Animal	03:56
At My Worst	Pink Sweat	02:50
Thousand Miles	The Kid Laroi	02:40
Airplane Thoughts	Dhruv	03:30
Takeaway	The Chainsmokers Illeni…	03:47
Cheating On You	Charlie Puth	03:27
Are You Happy		

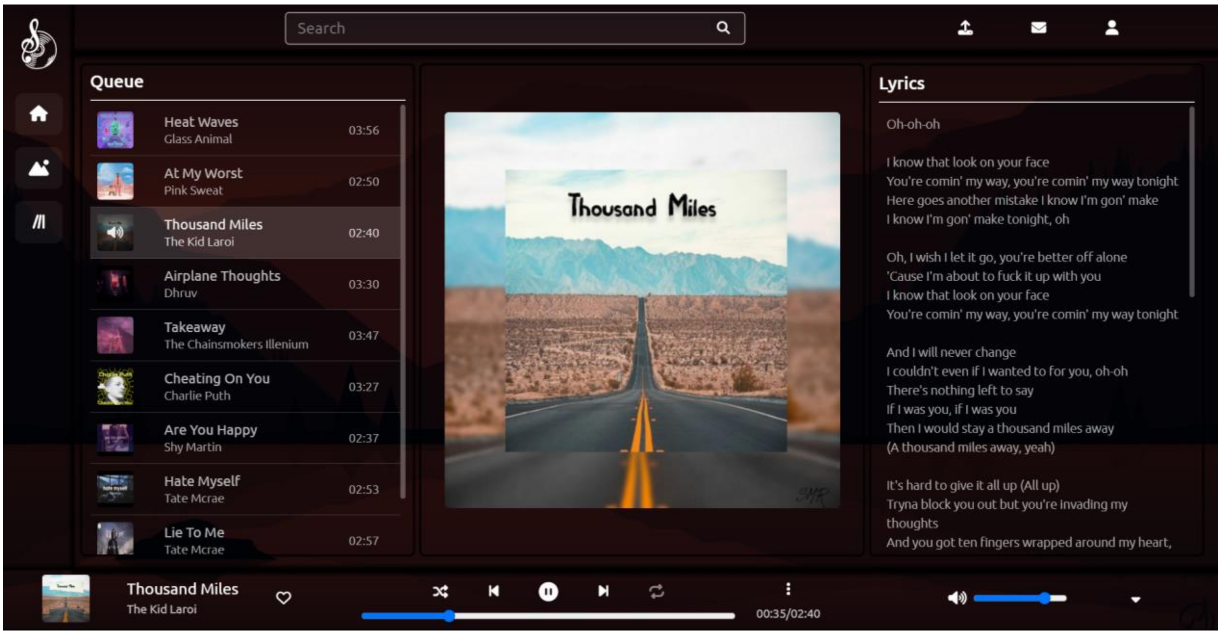
Lyrics

I've been thinking about you on this
aeroplane
And my mind is in the sky
When I'm dreaming about you on this
aeroplane

Could it be the elevation?
Must admit that I'm not
Used to feeling these sensations
Missing you like crazy
Though it's barely been a minute
You dropped me to the gate and
Bid farewell
Summer in Sevilla
Drinking red sangria

- These floating overlays can be toggled using the Queue and Lyrics buttons in the bottom player bar, or using the shortcut keys "Q" for Queue and "L" for Lyrics.
- The Queue Overlay shows the current playing queue songs with their covers, artists and duration.
- It also allows the user to play any other item in the queue by simply clicking it.
- The Lyrics Overlay simply displays the lyrics of the current playing song and gets updated as soon as a new song plays.

Extended Player for an immersive experience.



Queue

Heat Waves Glass Animal	03:56
At My Worst Pink Sweat	02:50
Thousand Miles The Kid Laroi	02:40
Airplane Thoughts Dhruv	03:30
Takeaway The Chainsmokers Illenium	03:47
Cheating On You Charlie Puth	03:27
Are You Happy Shy Martin	02:37
Hate Myself Tate Mcrae	02:53
Lie To Me Tate Mcrae	02:57

Lyrics

Oh-oh-oh

I know that look on your face
You're comin' my way, you're comin' my way tonight
Here goes another mistake I know I'm gon' make
I know I'm gon' make tonight, oh

Oh, I wish I let it go, you're better off alone
'Cause I'm about to fuck it up with you
I know that look on your face
You're comin' my way, you're comin' my way tonight

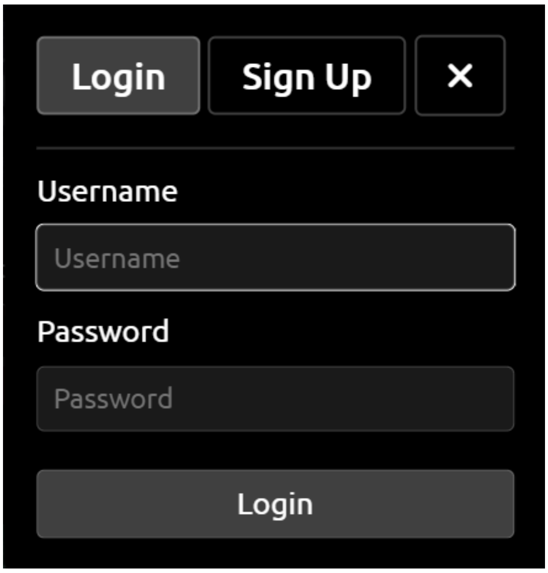
And I will never change
I couldn't even if I wanted to for you, oh-oh
There's nothing left to say
If I was you, if I was you
Then I would stay a thousand miles away
(A thousand miles away, yeah)

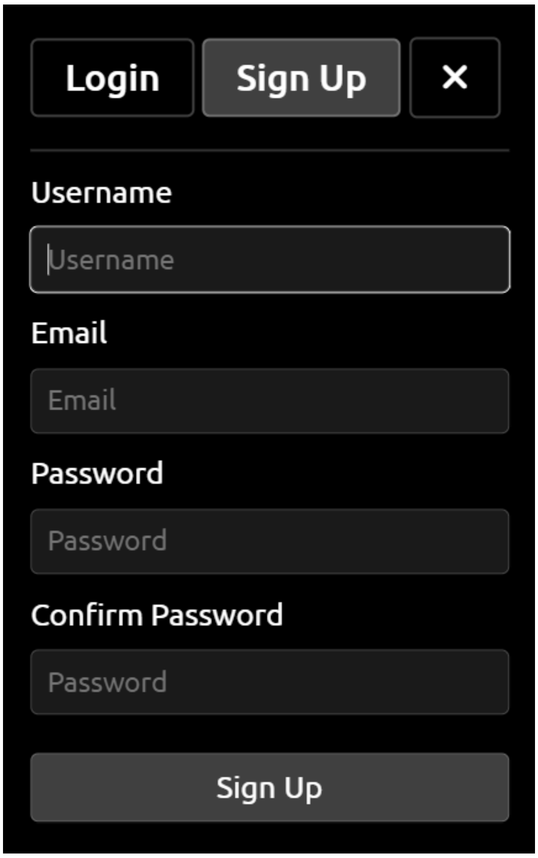
It's hard to give it all up (All up)
Tryna block you out but you're invading my
thoughts
And you got ten fingers wrapped around my heart,

Thousand Miles
The Kid Laroi

- The Extended Player can be toggled with the caret icon in bottom right and shows a larger view of the Queue, Album Cover and Lyrics of the playing song.
- The Queue Section in the Extended Player has the same functionality as the Floating Queue Overlay and allows the user to view the queue and play any other item in the queue.
- The Lyrics Section in the Extended Player shows a larger view of the lyrics to allow the user to easily read the lyrics while listening to their favourite songs.

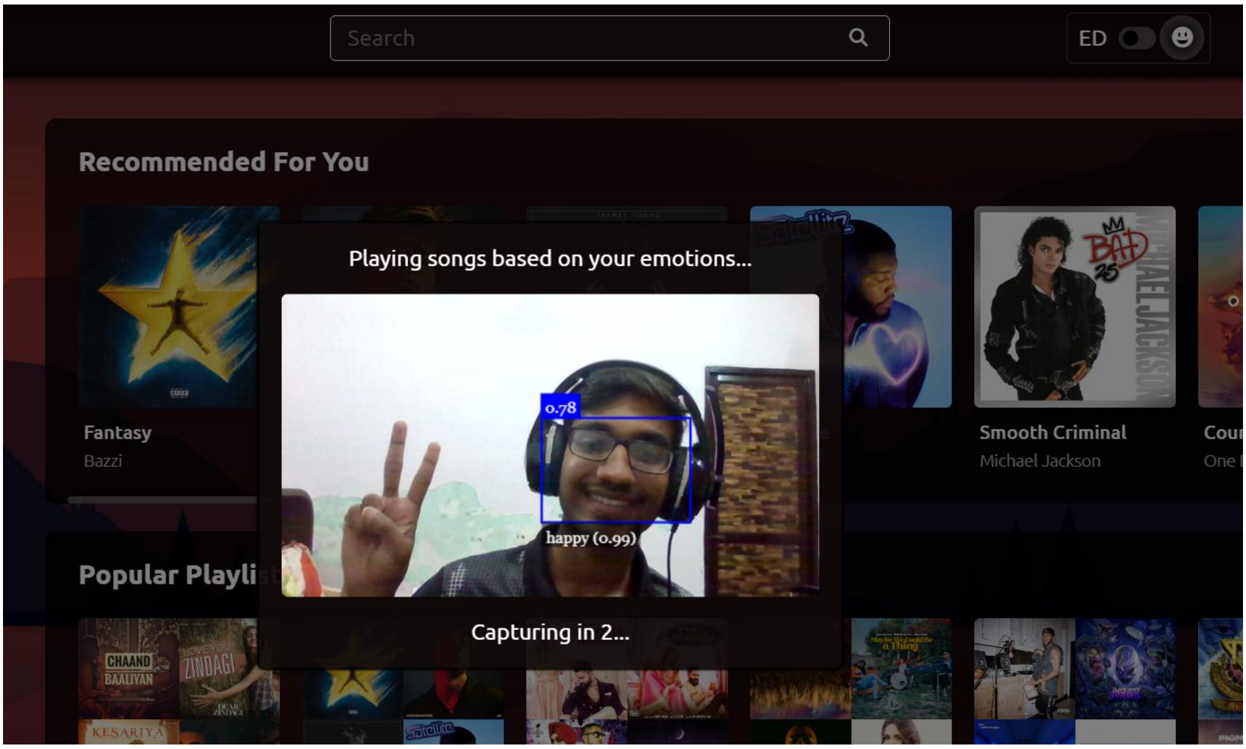
Login/Sign Up Popups to Log In or Sign Up.





- The user can Log In to their account or Sign Up to a new account allowing them to Like songs, check their Listening History and Save Playlists to their Library.
- Signing Up requires the user to enter a new unique Username, a valid Email Address, Password and then Confirm Password, all of which have to be valid.
- Logging in simply requires the user to enter their Username and Password, and then click the Login Button.

Emotion Detection Popup to detect the user's emotions.



Playing songs based on your emotions...

Capturing in 2...

- The user can allow camera access to the website to allow it to detect their emotions and play songs appropriate to their mood.
- The Emotion Detection panel in right portion of the Header has 2 buttons:
  - Automatic Emotion Detection Toggle: Enabling this automatically uses the webcam every time the playing queue ends to play the next set of songs based on the user's real-time mood.
  - Instant Emotion Detection: Clicking this button opens the emotion detection popup and webcam immediately to check for the user's real-time emotions and play songs accordingly.