Dharmsinh Desai University, Nadiad

Faculty Of Technology, Department of Computer Engineering

Subject :- Advanced Technologies(CE-515)

Project Report of

Beat Buzzz (Online Music Player)

Prepared By :-

Mahammadayan S. Shaikh (CE121)

ID :- 18CEUOG004

Guided By :-

Prof. Prashant M. Jadhav

Prof. Ankit P. Vaishnav

Dharmsinh Desai University, Nadiad

Faculty Of Technology, Department of Computer Engineering

Subject :- Advanced Technologies(CE-515)

CERTIFICATE

This is to certify that the **Advanced Technologies** project entitled **"Online Music Player"** is bonafide work of Mahammadayan S. Shaikh (CE121, ID:- 18CEUOG004) of Department Of Technology, Semester 5, academic year **2020-2021,** under our supervision and guidance.

Guide	**Guide**	**Head Of Department**
Prof. Prashant M. Jadhav	**Prof. Ankit P. Vaishnav**	**Prof. C. K. Bhensdadia**
Assistant Professor, Department of Computer Engineering Dharmsinh Desai University.	Assistant Professor, Department of Computer Engineering Dharmsinh Desai University.	Head Of Department, Department of Computer Engineering Dharmsinh Desai University.

Index

Abstract

Music is very important part of daily life. It has capacity to heal the wounds which even medicines can’t touch.

This Online Music Player gives various functionalities to users. Users can login to the website and can play various songs uploaded by admin. They can view playlists uploaded by admin. They can create their own playlists as well. They can search particular song.

Admin will be managing all songs and playlists. Admin can create new category of songs and can modify existing category. He/she can upload new songs and edit uploaded song. He/ She can create playlists from the songs uploaded.

-Developer Team

(Mahammadayan Shaikh)

Introduction

Aim of this project or website is to provide a user friendly platform to users for playing and managing their favourite music at one place.

This web application manages details of all the registered users. It provides forgot password functionality to users so if any user has forgotten his/ her password then they can reset it using this functionality. New users can register themselves to this web application. Registered users can login to the application and enjoy free music. Users can also see recently uploaded 10 songs. This application manages songs at admin end. Application stores songs' details like name, artist's name, category in database. It also manages playlists created by admin and stores data of playlists into database. It also takes care of invalid data for example admin can't upload file except .mp3 or .mpeg for song.

Tools – Technologies Used

- **Technologies :-**

  - HTML5 for creating web pages.
  - CSS3 (Cascading Style Sheet) for styling HTML pages.
  - Bootstrap4 for making web pages responsive and well structured.
  - Javascript for making web pages dynamic and for applying some client side validations.

- Jquery to manipulating DOM easily.
- Angular 10 for creating single page application and managing front-end.
- Typescript to create classes and models in angular application.
- Nodejs for managing back-end and providing API to front-end by interacting with database.
- Express for managing web application in nodejs.
- MongoDb as database to store data.

- **Tools :-**

  - Visual Studio Code as IDE
  - MongoDb compass
  - Browser to run application

- **Platform :-**

  - Windows 10

Software Requirement Specification (SRS)

For

Online Music Player

- **Description :-**

  This is an web project for an Online Music Player in which various users can register to the website. Then After successfully registering, they can log in to the website and use different functionalities provided. They can listen songs, create and manage playlists of their favourite songs, search particular songs etc. And Admin can upload , provide and manage different songs & playlists.

- **Functional Requirements of the system** :-

R1. Manage User-Module .

R1.1 Manage Users' Registeration

Description :- New user can register to the system.

Input :- User's Email-Id & Password

Output :- Confimation message regarding successful registration.

R 1.2 Manage Login

Description :- Registered users can login to the system.

Input :- User’s Email-Id & Password

Output :- Redirection to Home Page after successful login.

R1.3 Manage Logout.

Description :- Logged User can logout from the system.

Input :- User selection.

Output :- Success message and redirection to home page.

R1.3 Forgot Password.

Description :- If user has forgotten his/ her password, then he/ she can reset it

Input :- User’s registered mail-id.

Output :- Verification code for resetting the password through mail.

R2. Manage Songs.

Description :- User can play, download, search different songs.

R2.1 Search Song.

Description :- User can search particular song.

Input :- Song name.

Output :- Search list according to input.

R2.2 Play Songs.

Description :- User can play song.

Input :- User selection.

Output :- Play the selected song.

R2.3 Like Song.

Description :- User can like particular song.

Input :- User selection.

Output :- Success message.

R2.5 Share Song.

Description :- User can share a song.

Input :- Song to be shared.

Output :- Link of the song.

R3. Manage playlists.

Description :- User can create, update & manage playlists.

R3.1 Create playlist.

Description :- User can create his/her new playlist and add songs into it.

Input :- Songs to be added in the playlist & poster for playlist

Output :- Success message regarding creation of playlist.

R3.2 Update playlist.

Description :- User can edit already created playlist.

Input :- Playlist that is to be updated.

Output :- Success message regarding updation of playlist/

R4. Manage Favourites .

Description :- User can see songs liked by him/her.

Input :- User selection.

Output :- List of liked songs.

R5. Manage Admin Panel.

Description :- Admin can perform certain tasks like uploading songs, uploading playlists, view statistics etc.

R5.1 Admin Login

Description :- Admin can login to the system

Input :- Admin's Email & password.

Output :- Redirection to home page after successful login.

R5.2 Upload Song.

Description :- Admin can upload new song.

Input :- Song details.

Output :- Success message.

R5.3 Update song.

Description :- Admin can update the uploaded song

Input :- Song’s name & identity which is to be updated.

Output :- Success message regarding updation.

R5.4 Manage categories.

R5.4.1 Add categories.

Description :- Admin can add new categories.

Input :- Category’s name

Output :- Success message.

R5.4.2 Update categories.

Description :- Admin can update added category.

Input :- Category name & identity.

Output :- Success message regarding updation of playlist.

R5.5 Manage statistics.

Description :- Admin can view statistical data like total songs, total users, total likes etc.

Input :- User Selection.

Output :- Corresponding statistical data.

Design

i. DTD :-

1. Users

```
<?xml version="1.0"?>
<!DOCTYPE users [
    <!ELEMENT users (user+)>
    <!ELEMENT user (id,email,password)>
    <!ELEMENT id (#PCDATA)>
    <!ELEMENT email (#PCDATA)>
    <!ELEMENT password (#PCDATA)>
]>
<users>
    <user>
        <id>ObjectId(123456)</id>
        <email>mahammadayan18@gmail.com</ email >
        <password>**** </password>
    </user>
```

```
</users>
```

2. Songs

```
<?xml version="1.0"?>
<!DOCTYPE songs [
    <!ELEMENT songs(song+) >
    <!ELEMENT song (id,ArtistName,SongName,SongPath,PosterPath,Category,UploadTime)>
    <!ELEMENT id (#PCDATA)>
    <!ELEMENT SongName (#PCDATA)>
    <!ELEMENT SongPath(#PCDATA)>
    <!ELEMENT PosterPath(#PCDATA)>
    <!ELEMENT Catagory(#PCDATA)>
    <!ELEMENT UploadTime(#PCDATA)>

]>
<songs>
    <song>
        <id>ObjectId(123456)</id>
        <SongName>Kun Faya Kun</ SongName >
        <SongPath> Kun_faya_kun.mp3 </SongPath>
        <PosterPath> img1.png </PosterPath>
        <Catagory> Hindi </Catagory>
        <UploadTime> 2020-10-31T04:06:03.562+00:00 </UploadTime>
    </song>
</songs>
```

3. Catagorys

```
<?xml version="1.0"?>
<!DOCTYPE catagorys [
    <!ELEMENT catagorys(category+)>
    <!ELEMENT catagory(id,email,password)>
    <!ELEMENT id (#PCDATA)>
    <!ELEMENT name (#PCDATA)>
]>
<catagorys>
    <catagory>
        <id>ObjectId(123456)</id>
        <name>Hindi</ name>
    </catagory>

</catagorys>
```

4. Playlists

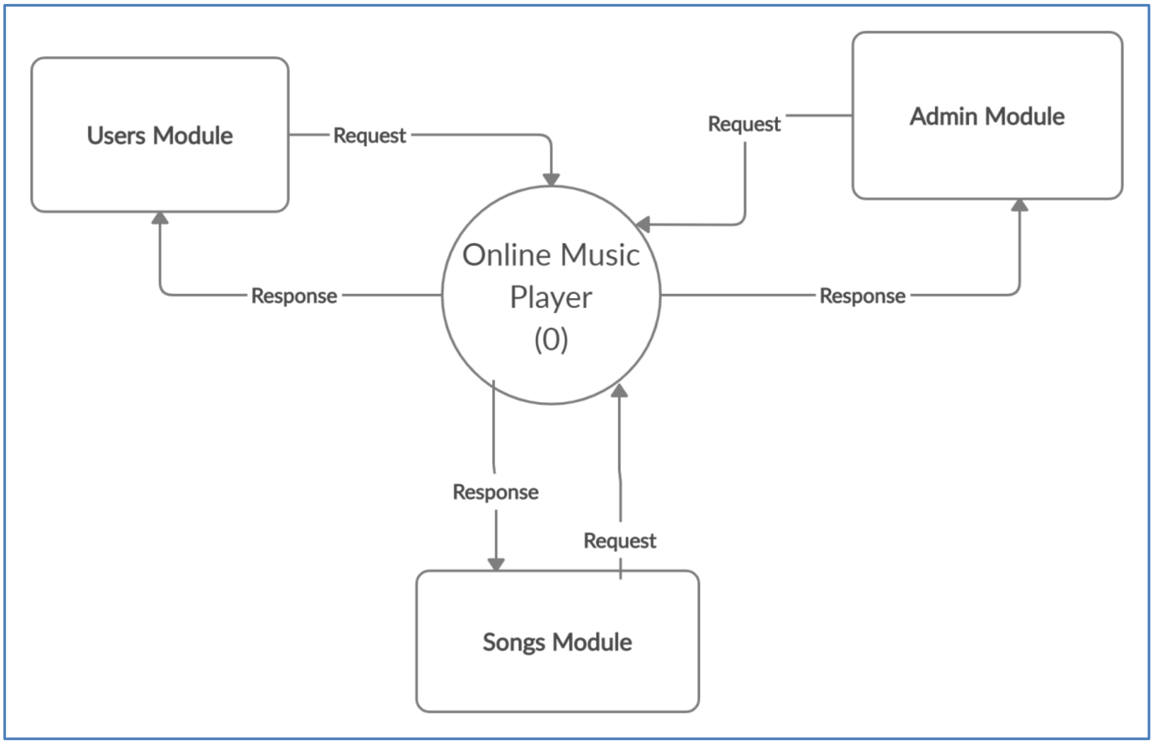
```
<?xml version="1.0"?>
<!DOCTYPE playlists [
    <!ELEMENT playlists(playlist+)>
    <!ELEMENT playlist(id,name,posterpath,songs)>
    <!ELEMENT id (#PCDATA)>
    <!ELEMENT name (#PCDATA)>
    <!ELEMENT posterpath (#PCDATA)>
    <!ELEMENT songs (song+)>
    <!ELEMENT song (#PCDATA)>
]>
<playlists>
    <id>ObjectId(123456)</id>
    <name>2000s songs</ name>
    <posterpath>poster1.jpg</ posterpath>
    < songs>
            <song>Zindagi Ye Safar</song>
            <song>Kun Faya kun</song>
    </ songs >

</playlists>
```

5. MyPlaylists

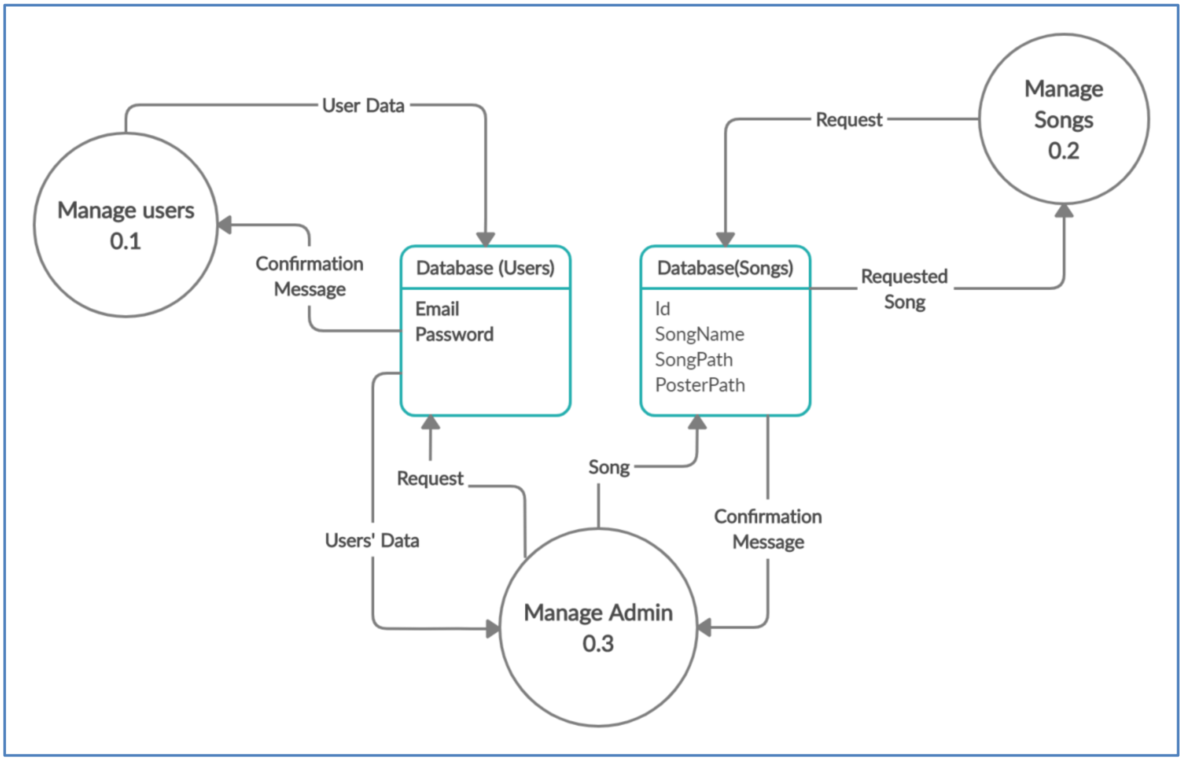
```
<?xml version="1.0"?>
<!DOCTYPE myplaylists [
    <!ELEMENT myplaylists(playlist+)>
    <!ELEMENT myplaylist(id,userid,name,posterpath,songs)>
    <!ELEMENT id (#PCDATA)>
    <!ELEMENT userid (#PCDATA)>
    <!ELEMENT name (#PCDATA)>
    <!ELEMENT posterpath (#PCDATA)>
    <!ELEMENT songs (song+)>
    <!ELEMENT song (#PCDATA)>
]>
<myplaylists>
    <id>ObjectId(123456)</id>
    <userid>ObjectId(f13237)</userid>
    <name>Morning Songs</ name>
    <posterpath>poster2.jpg</ posterpath>
    < songs>
            <song>Vaalam Aavo Ne</song>
            <song>Kun Faya kun</song>
    </ songs >

</myplaylists>
```

ii. DFD(Data Flow Diagram) :-

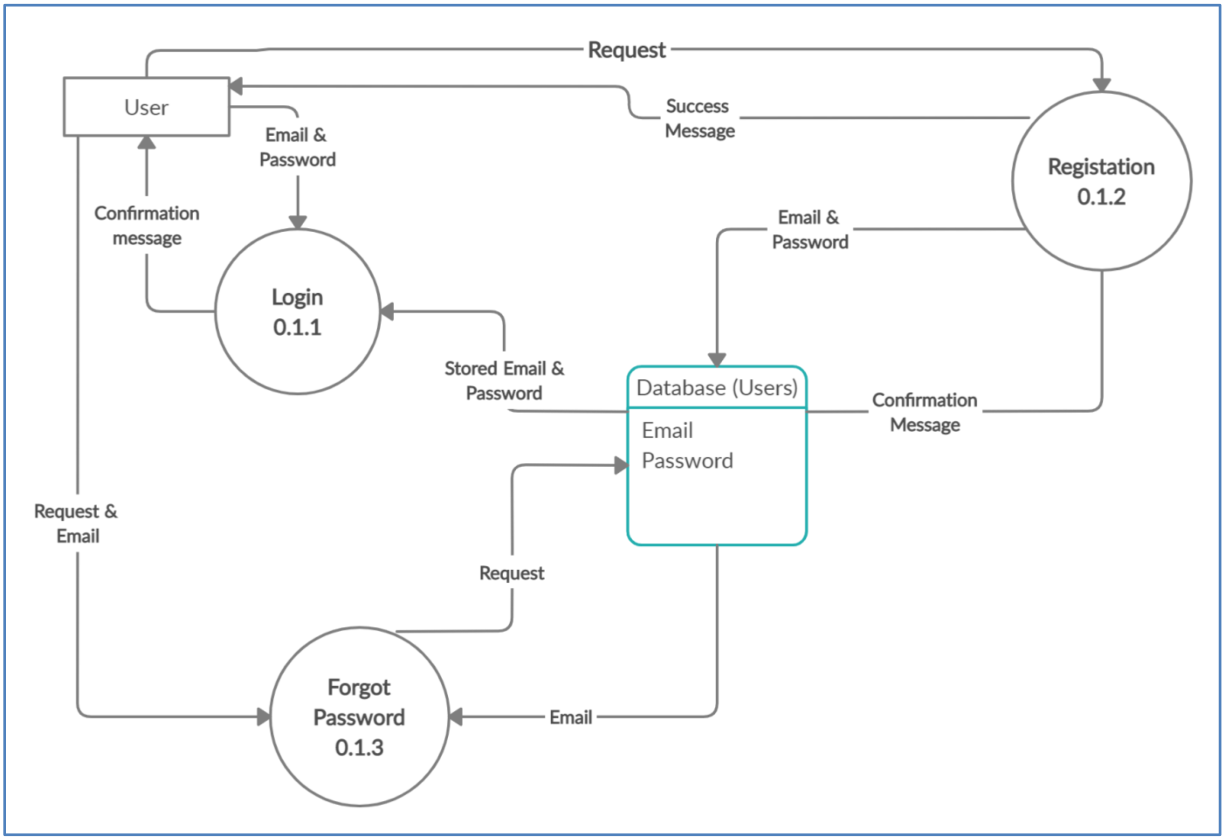
- Level 0 :-



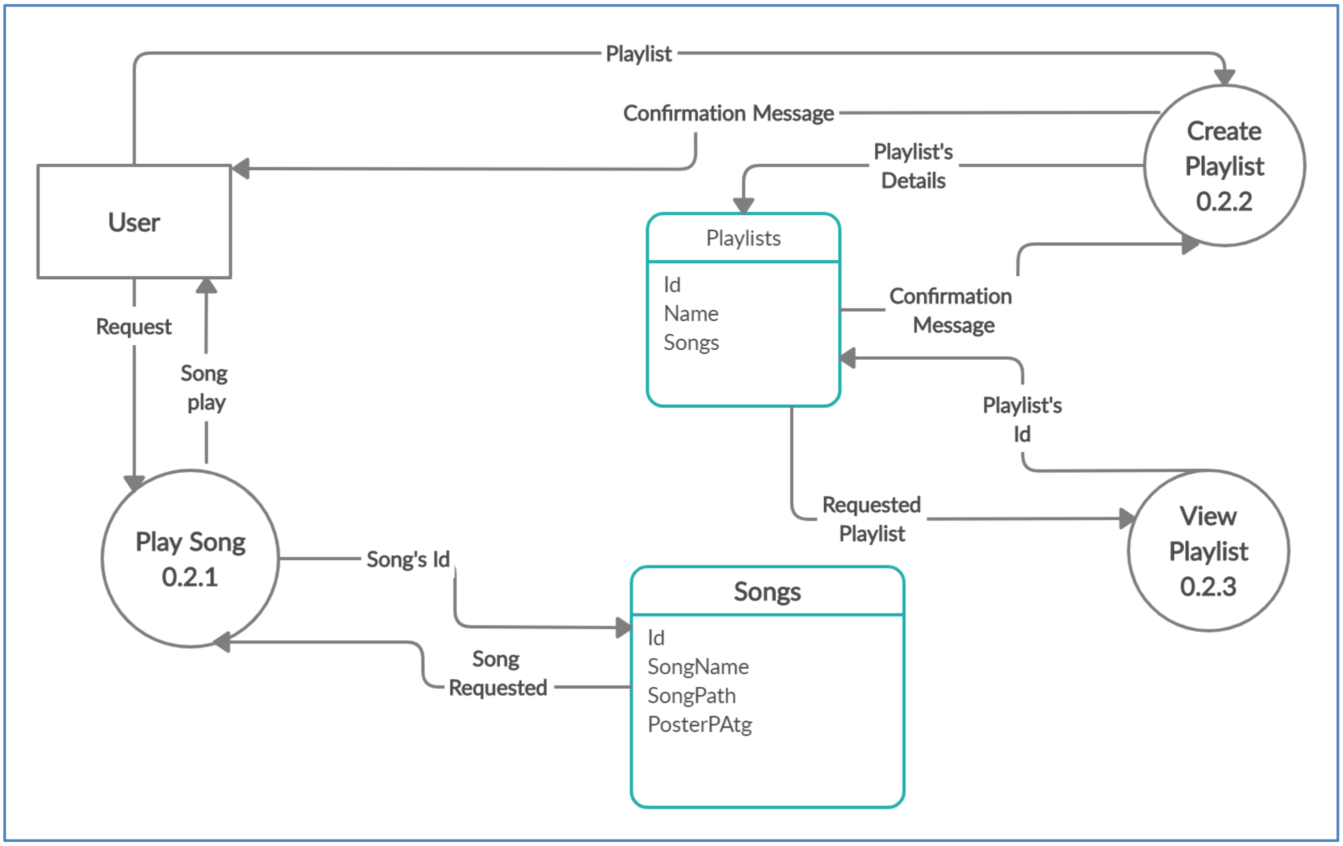
- Level 1 :-



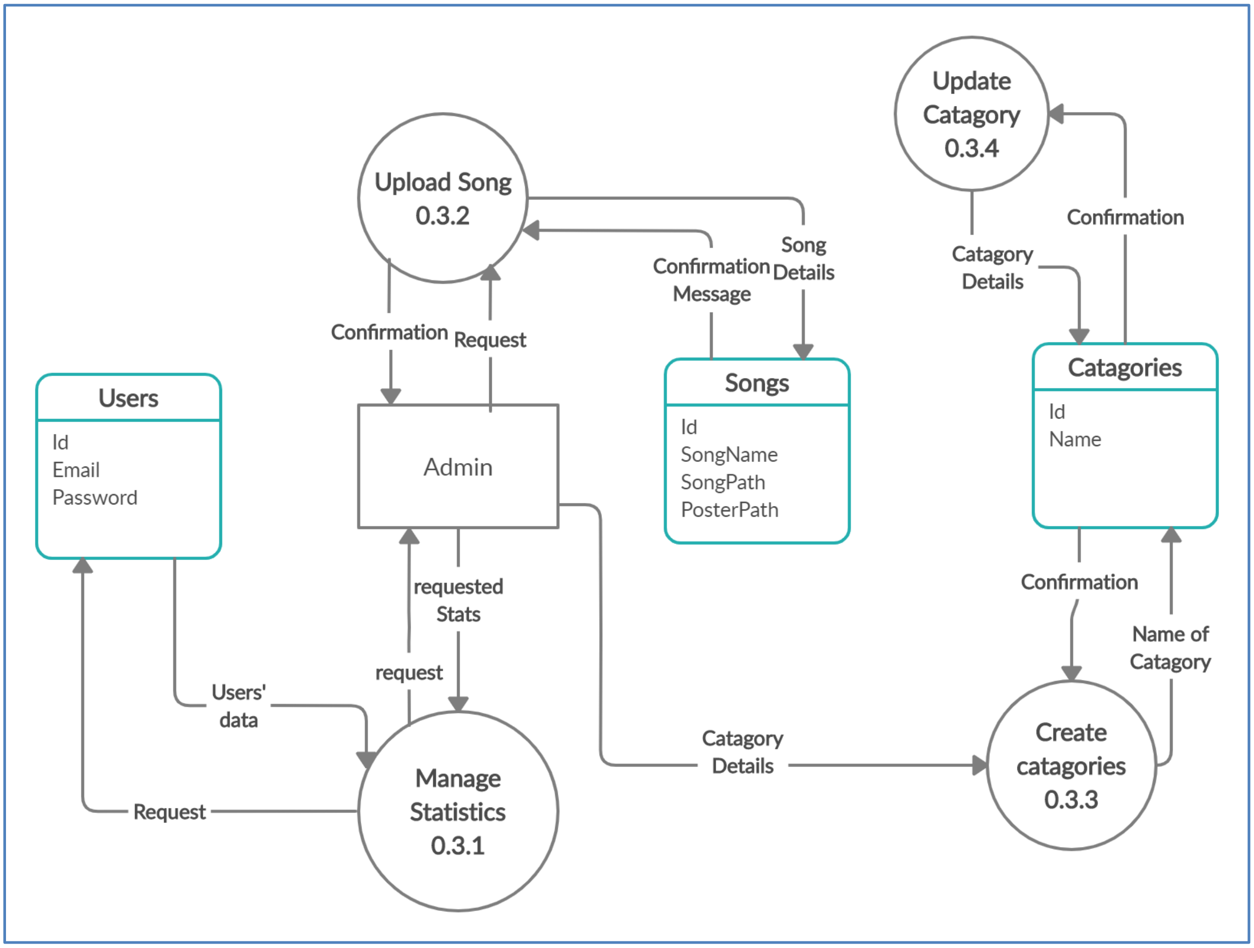
- **Level 2 for manage users :-**



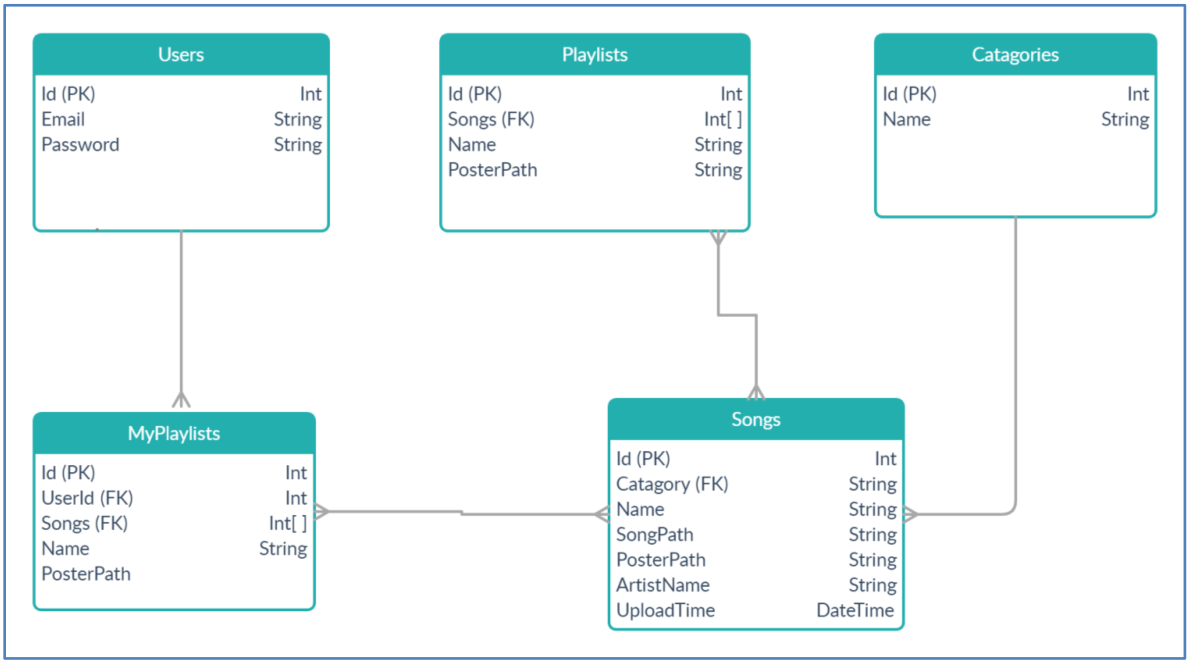
- Level 2 diagram for manage songs :-



- Level 2 diagram for manage admin : -



iii. ER Diagram :-



iv. Data-Dictionary :-

- **Users :-**

Sr. No	Field Name	Data Type	Required	Unique	PK/F	Reference
1	Id	Int	Yes	Yes	Yes	-
2	Email	String	Yes	Yes	No	-
3	Password	String	Yes	No	No	-

- **MyPlaylists :-**

Sr. No	Field Name	Data Type	Required	Unique	PK/F	Reference
1	Id	Int	Yes	Yes	Yes	-
2	UserId	Int	Yes	No	Yes	Id(User)
3	Songs	Int [ ]	Yes	No	Yes	Id(Songs)
4	Name	String	Yes	No	No	-
5	PosterPath	String	Yes	No	No	-

- **Playlists :-**

Sr. No	**Field Name**	**Data Type**	**Required**	**Unique**	**PK/F**	**Reference**
1	Id	Int	Yes	Yes	Yes	**-**
2	Songs	Int [ ]	Yes	No	Yes	Id(Songs)
3	Name	String	Yes	No	No	-
4	PosterPath	String	Yes	No	No	-

- **Songs :-**

Sr. No	**Field Name**	**Data Type**	**Required**	**Unique**	**PK/F**	**Reference**
1	Id	Int	Yes	Yes	Yes	**-**
2	SongName	String	Yes	No	Yes	-
3	SongPath	String	Yes	Yes	No	-
4	ArtistName	String	Yes	No	No	-
5	PosterPath	String	Yes	No	No	-
6	UploadTime	DateTime	Yes	Yes	No	-

- **Catagories :-**

Sr. No	**Field Name**	**Data Type**	**Required**	**Unique**	**PK/F**	**Reference**
1	Id	Int	Yes	Yes	Yes	**-**
2	Name	String	Yes	No	Yes	-

Implementation Details

- **Modules :-**

  - **Users Module :-** This module manages the user’s login, signup & forgot passwords functionalities. If user wants to login or logout to the system, they can do that using this module’s feature. If user has forgotten password, then he can reset it. Verification code will be sent to user via mail. If user enter correct verification code, then he/she will be allowed to reset password. New User can register to the system. If email provided by user is already registered then user will get error message.

  - **Admin module :-** Admin can view overall statistical data of the system like total songs, category wise songs, playlist wise songs. He can upload new songs, edit song and delete song. He can upload new categories, edit or delete existing category. He can create new playlist from the uploaded songs.

  - **Song Module :-** User can view and play all the songs uploaded by admin. User can also search particular song by entering song’s name or artist’s name.

  - **Playlist Module :-** User can view playlists created by the admin. And can see and play all songs in the playlist. Users can create their own playlists from available songs, too.

- **Major functions :-**

  - **Admin’s functionalities :-**
    - Login
    - Upload song
    - Edit/Delete song
    - Create Playlist
    - View Statistics
    - Create category
    - Update/ Delete category.

  - **Users’ functionalities :-**
    - Login
    - Signup
    - Forgot password
    - View list of songs
    - Play Song
    - Search song
    - View Playlists
    - Create own playlists

Testing

Black box testing of functions :-

- **Function 1 :- Authorization.**
  - Case 1 :- Trying to access dashboard page using url without login.
  - O/P :- Redirected to login page.

  - Case 2 :- Trying to access dashboard page with url after login.
  - O/P :- Allowed to access page.

- **Function 2 :- Login.**
  - Case 1 :- Login with invalid credentials.
  - O/P :- Error message regarding invalid login.

  - Case 2 :- Login with proper credentials.
  - O/P :- Redirected to dashboard page.

- **Function 3 :- Signup**
  - Case 1 :- Signup with already registered email-id.
  - O/P :- Error message saying “Email-Id is already registered”.

  - Case 2 :- Signup with unique Email-Id.
  - O/P :- Successfully registered.

- **Function 4 :- Upload Song by admin.**
  - Case 1 :- Selecting .pdf file in song input.
  - O/P :- Error message saying “only .mp3 & .mpeg files supported”.

  - Case 2 :- Selecting .docx file in Poster image input.
  - O/P :- Error message saying “only .jpg ,.jpeg or .png files supported”.

  - Case 3 :- Selecting .mp3 for song, .jpeg for poster & all other fields filled up.
  - O/P :- Song successfully uploaded.

- **Function 5 :- Add category by admin :-**
  - Case 1 :- Adding category with existing Category name.
  - O/P :- Error message saying “Category is already there”.

  - Case 2 :- Adding category with unique name.
  - O/P :- Category successfully added.

- **Function 6 :- Search Song.**
  - Case 1 :- Entering song name or Artist’s which is not uploaded.
  - O/P :- No Song found.

  - Case 2 :-Entering song name or Artist’s which is there in database.
  - O/P :- List of songs and artist containing user input.

- **Function 7 :- Forgot password.**
  - Case 1 :- Entering email which isn’t registered.
  - O/P :- Error message saying “Invalid Email”.

- Case 2 :- Entering valid registered email.
- O/P :- Verification code sent via mail.

➢ **Function 8 :- Verification code's verification.**

- Case 1 :- Entering Invalid Verification code.
- O/P :- Error message saying "Invalid verification code".

- Case 2 :- Entering valid verification code.
- O/P :- Redirected to reset password page.

➢ **Function 9 :- Reset Password page.**

- Case 1 :- Trying to access reset password page without going through verification.
- O/P :- Error message.

- Case 2 :- Accessing reset password page after verifying verification code.
- O/P :- Accessed allowed for resetting password.

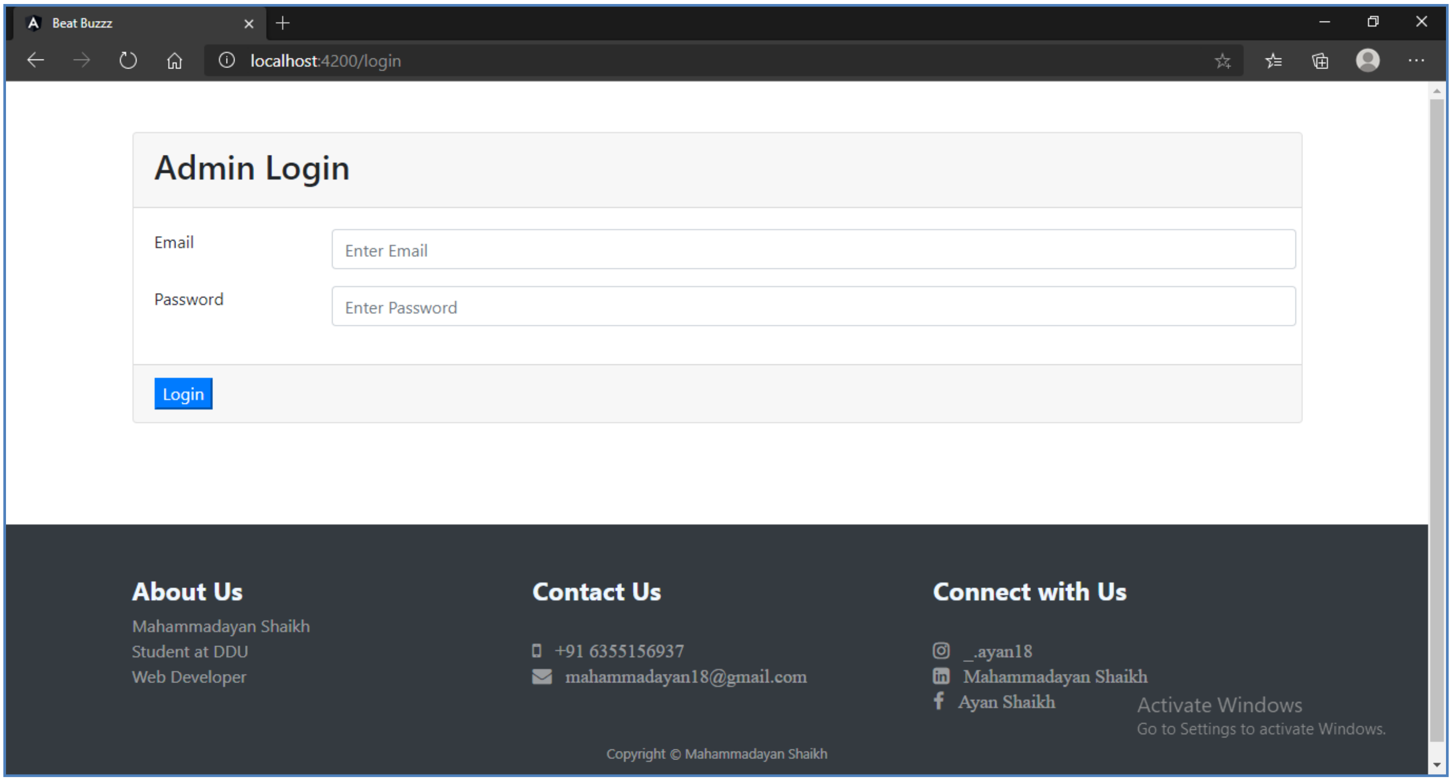
➢ **Function 10 :- Accessing authorised pages after logout.**

- Case 1 :- Pressing back button in browser after clicking on logout.
- O/P :- Access is denied and redirected to login page.

Screen Shots :-

➢ Admin app :-

- Admin login :-



Admin Login

Email: Enter Email

Password: Enter Password

About Us

Mahammadayan Shaikh
Student at DDU
Web Developer

Contact Us

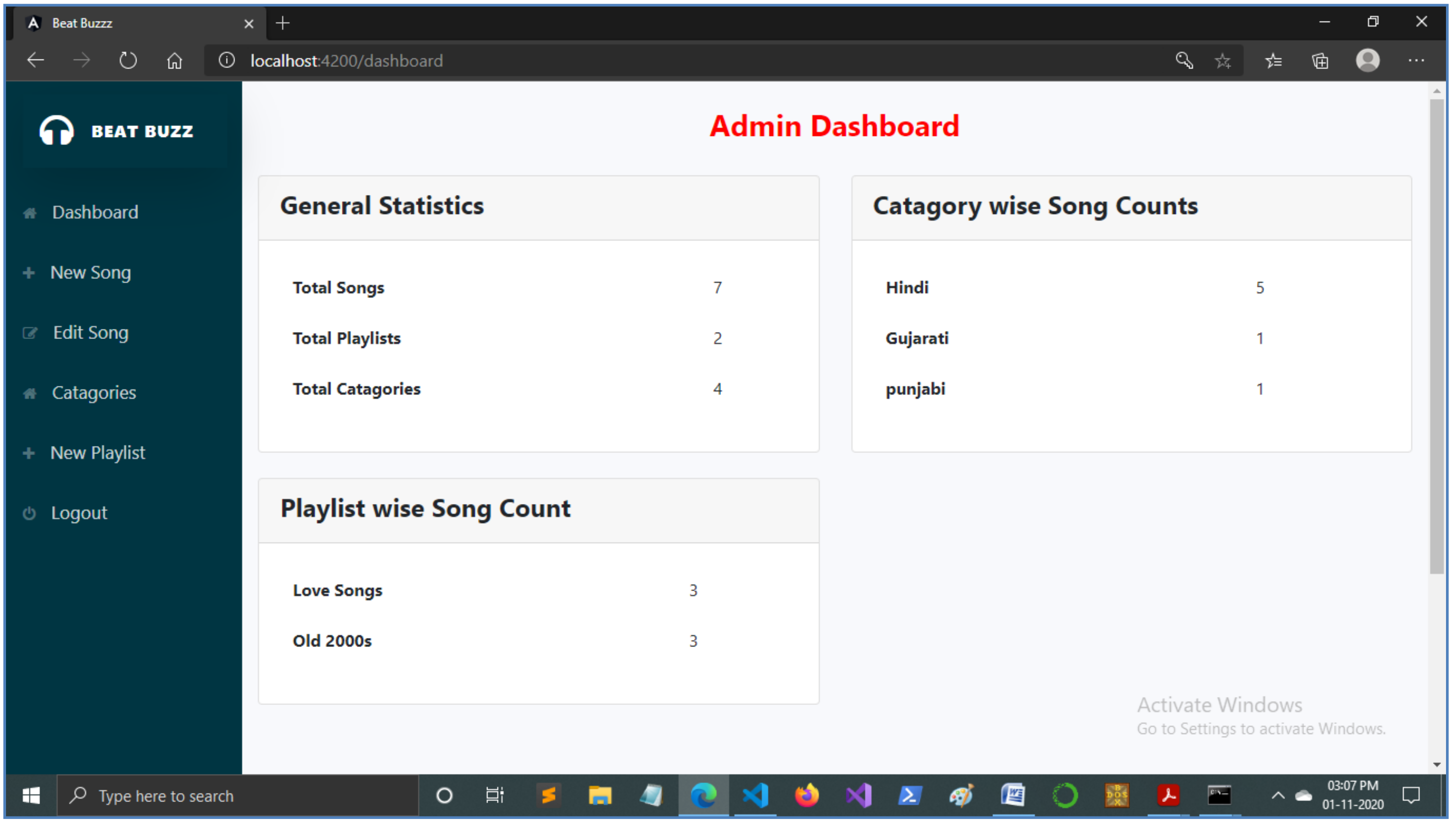
+91 6355156937
mahammadayan18@gmail.com

Connect with Us

_ayan18
Mahammadayan Shaikh
Ayan Shaikh

Copyright © Mahammadayan Shaikh

- Admin dashboard :-



Admin Dashboard

General Statistics

Total Songs	7
Total Playlists	2
Total Catagories	4

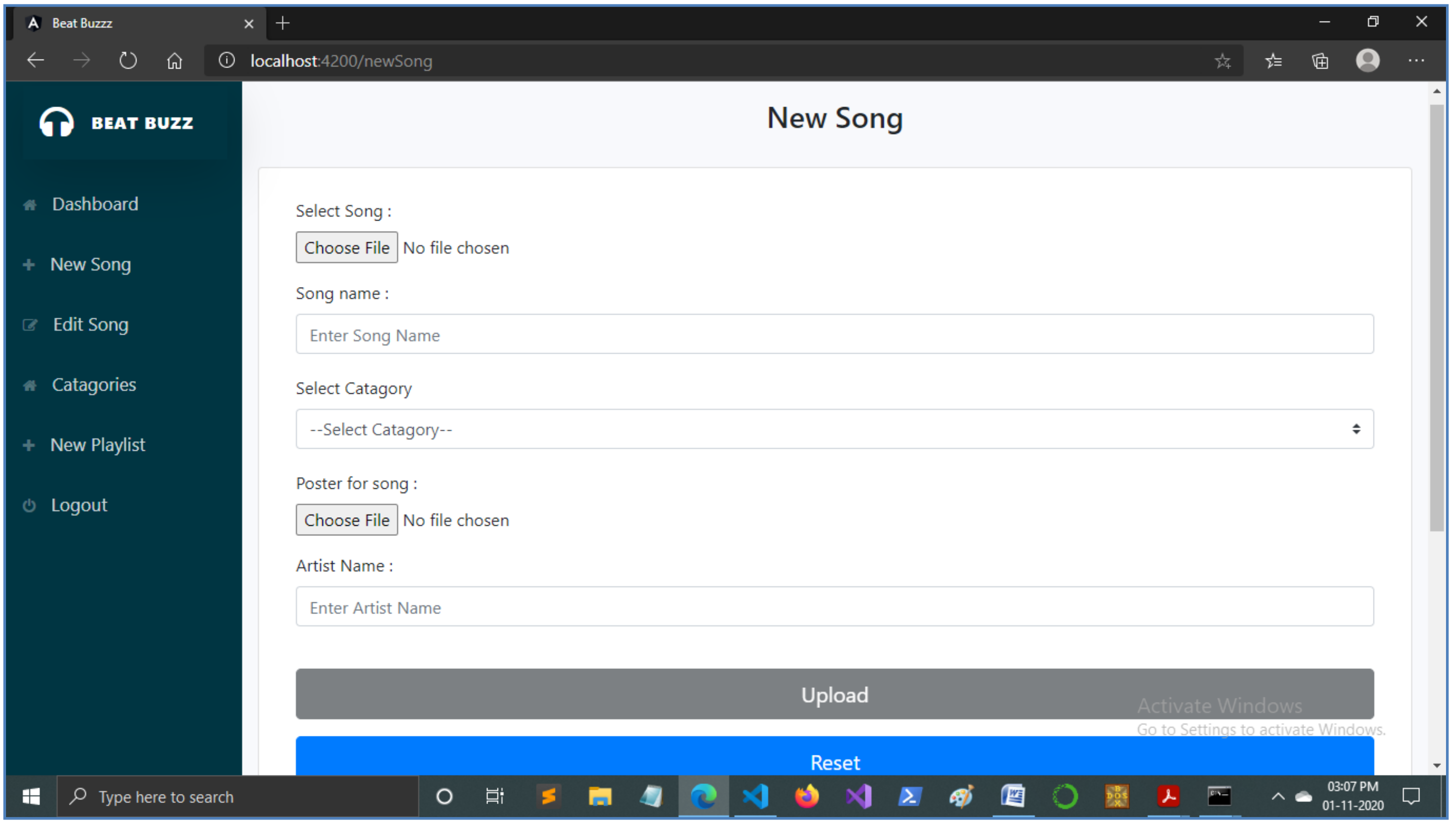
Catagory wise Song Counts

Hindi	5
Gujarati	1
punjabi	1

Playlist wise Song Count

Love Songs	3
Old 2000s	3

- New Song :-



New Song

Select Song : No file chosen

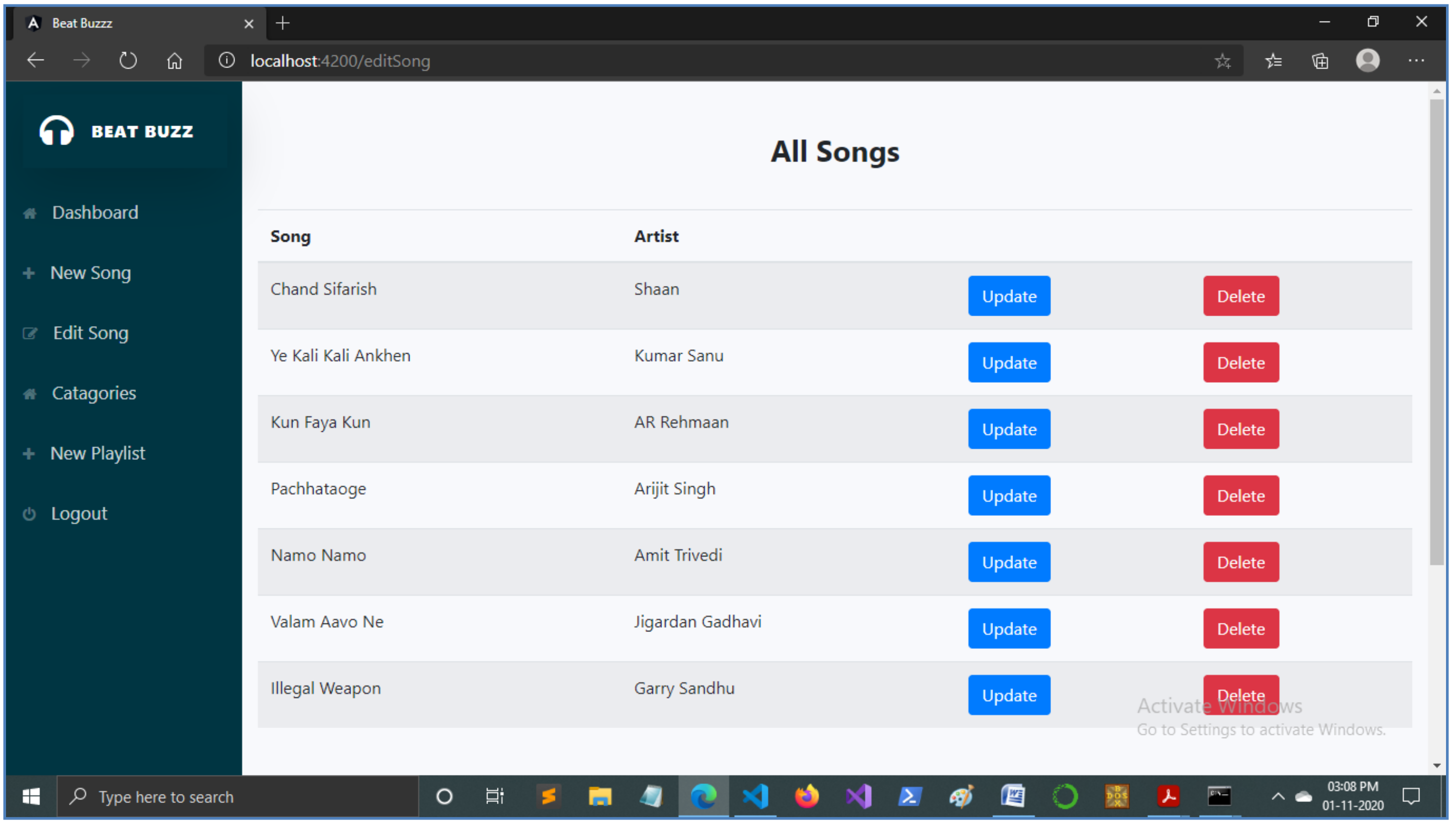
Song name : Enter Song Name

Select Catagory --Select Catagory--

Poster for song : No file chosen

Artist Name : Enter Artist Name

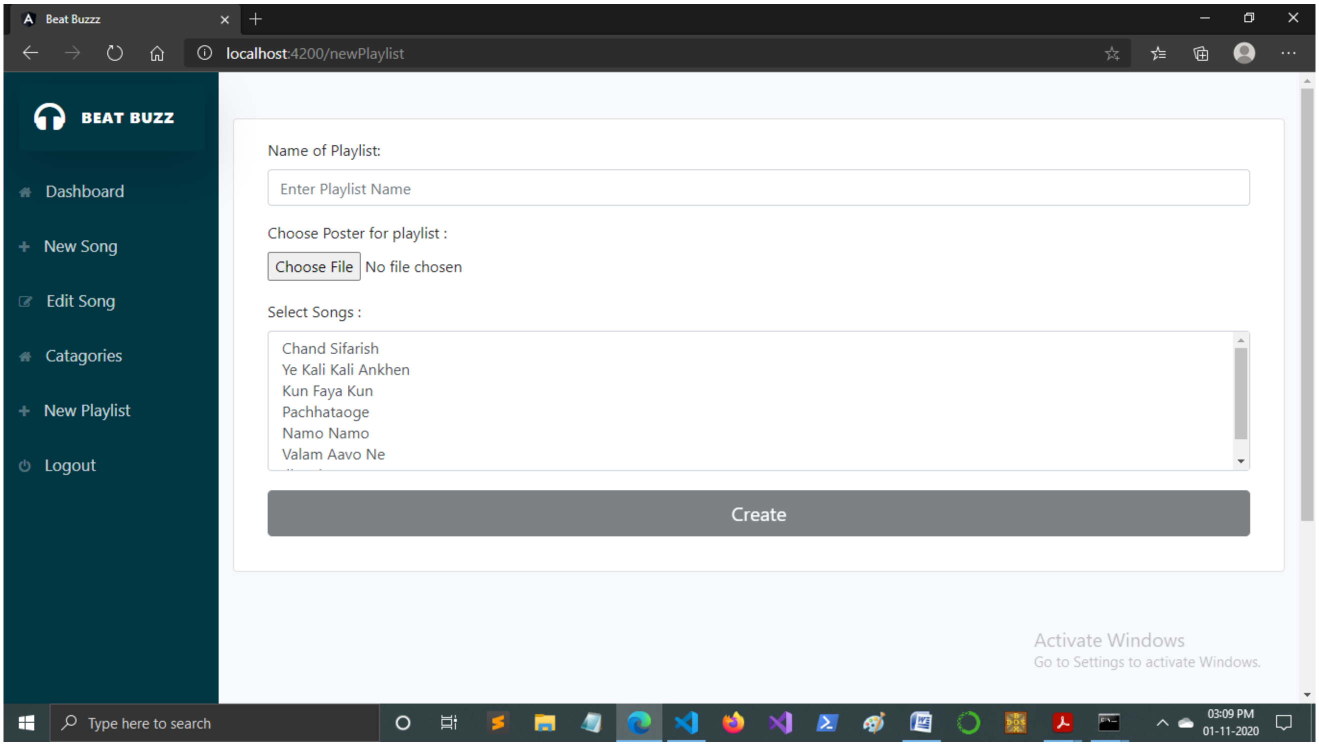
- Edit Song :-



All Songs

Song	Artist		
Chand Sifarish	Shaan	Update	Delete
Ye Kali Kali Ankhen	Kumar Sanu	Update	Delete
Kun Faya Kun	AR Rehmaan	Update	Delete
Pachhataoge	Arijit Singh	Update	Delete
Namo Namo	Amit Trivedi	Update	Delete
Valam Aavo Ne	Jigardan Gadhavi	Update	Delete
Illegal Weapon	Garry Sandhu	Update	Delete

- New Playlist :-



Name of Playlist:

Enter Playlist Name

Choose Poster for playlist :

Choose File No file chosen

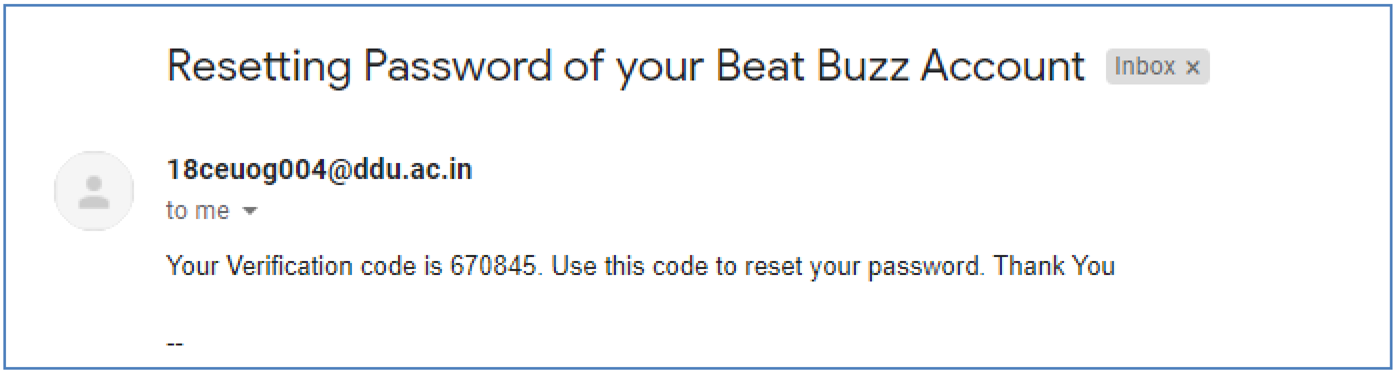
Select Songs :

Chand Sifarish
Ye Kali Kali Ankhen
Kun Faya Kun
Pachhataoge
Namo Namo
Valam Aavo Ne

Create

➢ User Application :-

- Mail sent for reset password :-



Resetting Password of your Beat Buzz Account Inbox ×

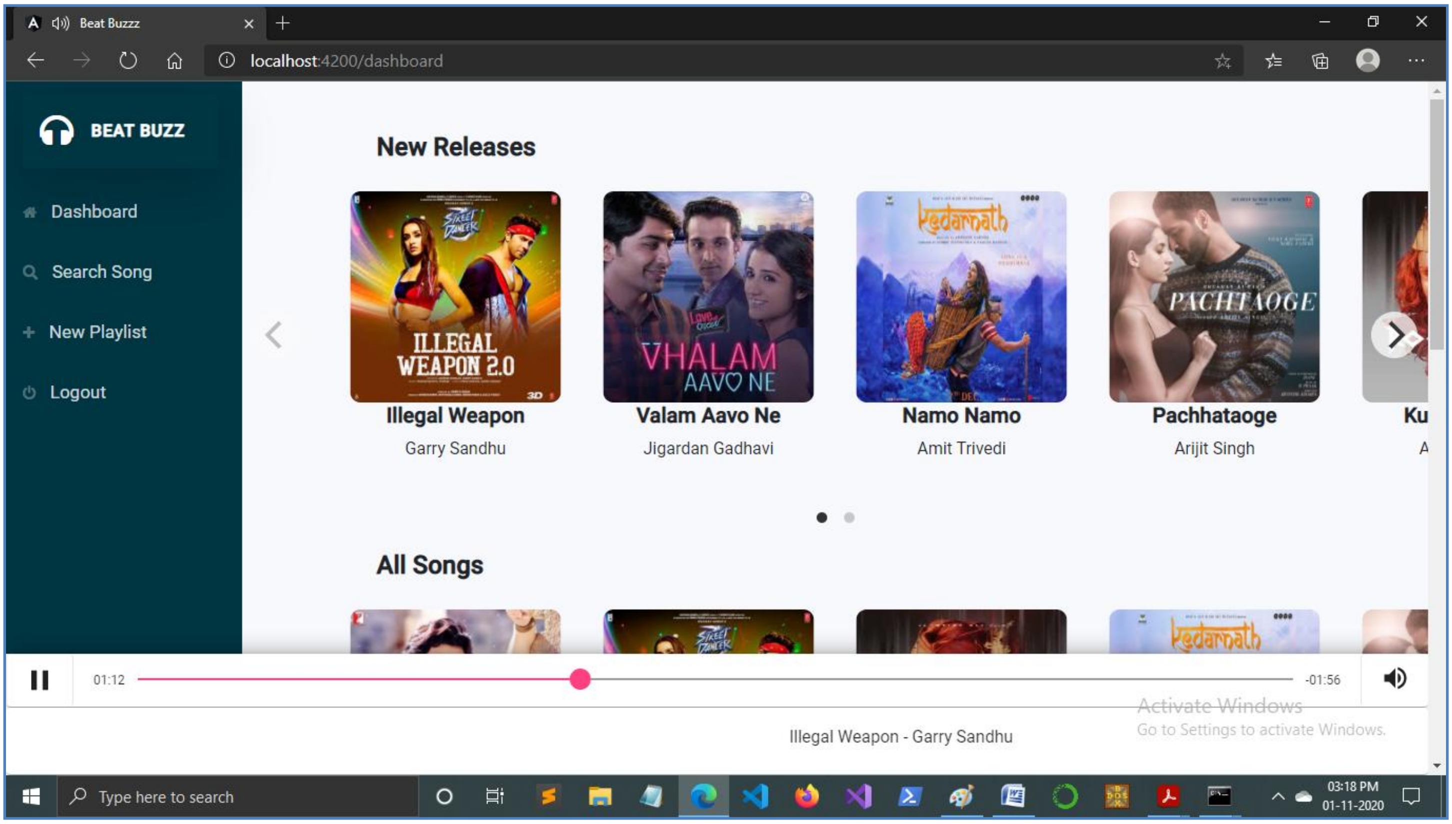
18ceuog004@ddu.ac.in

to me

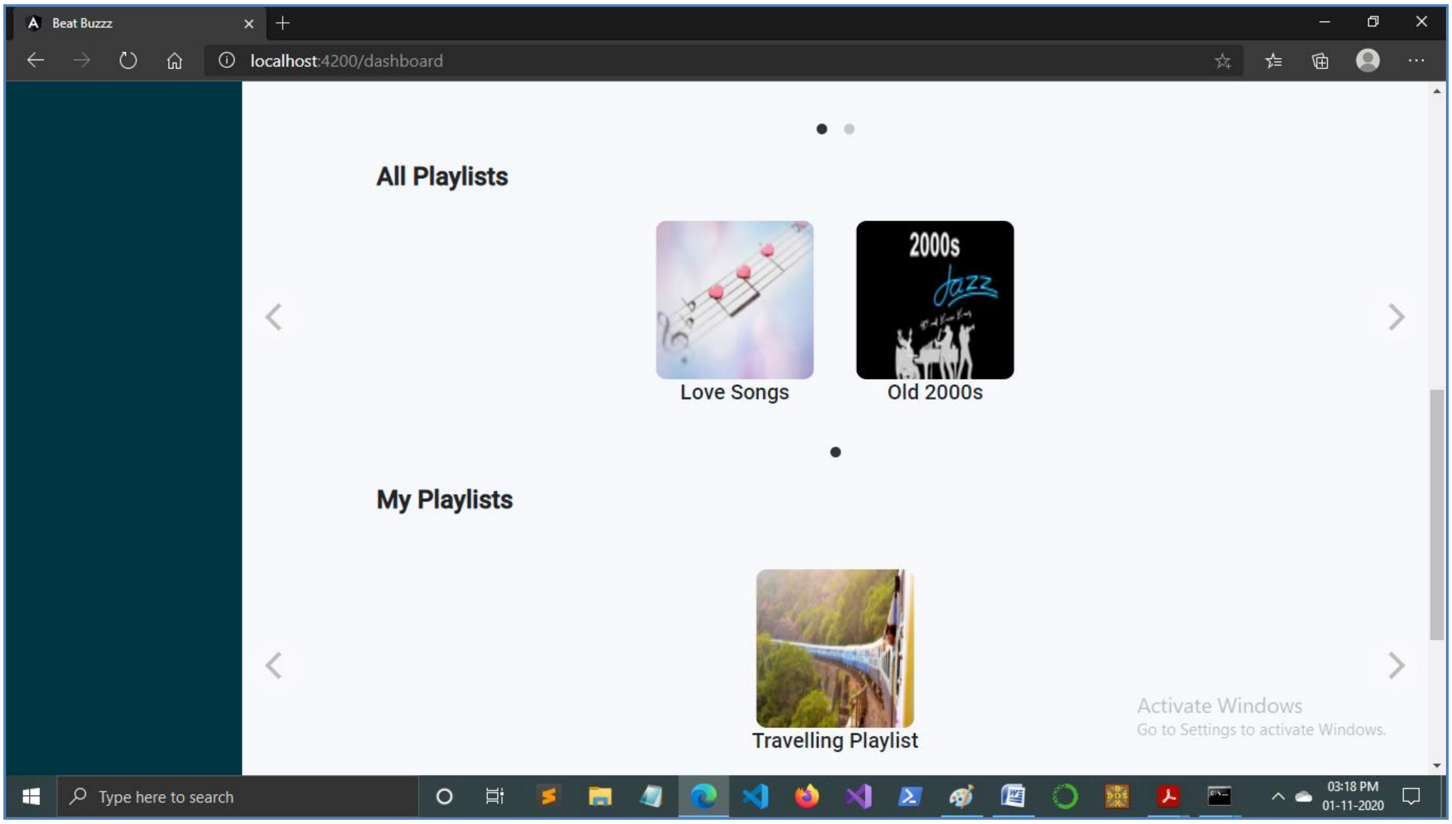
Your Verification code is 670845. Use this code to reset your password. Thank You

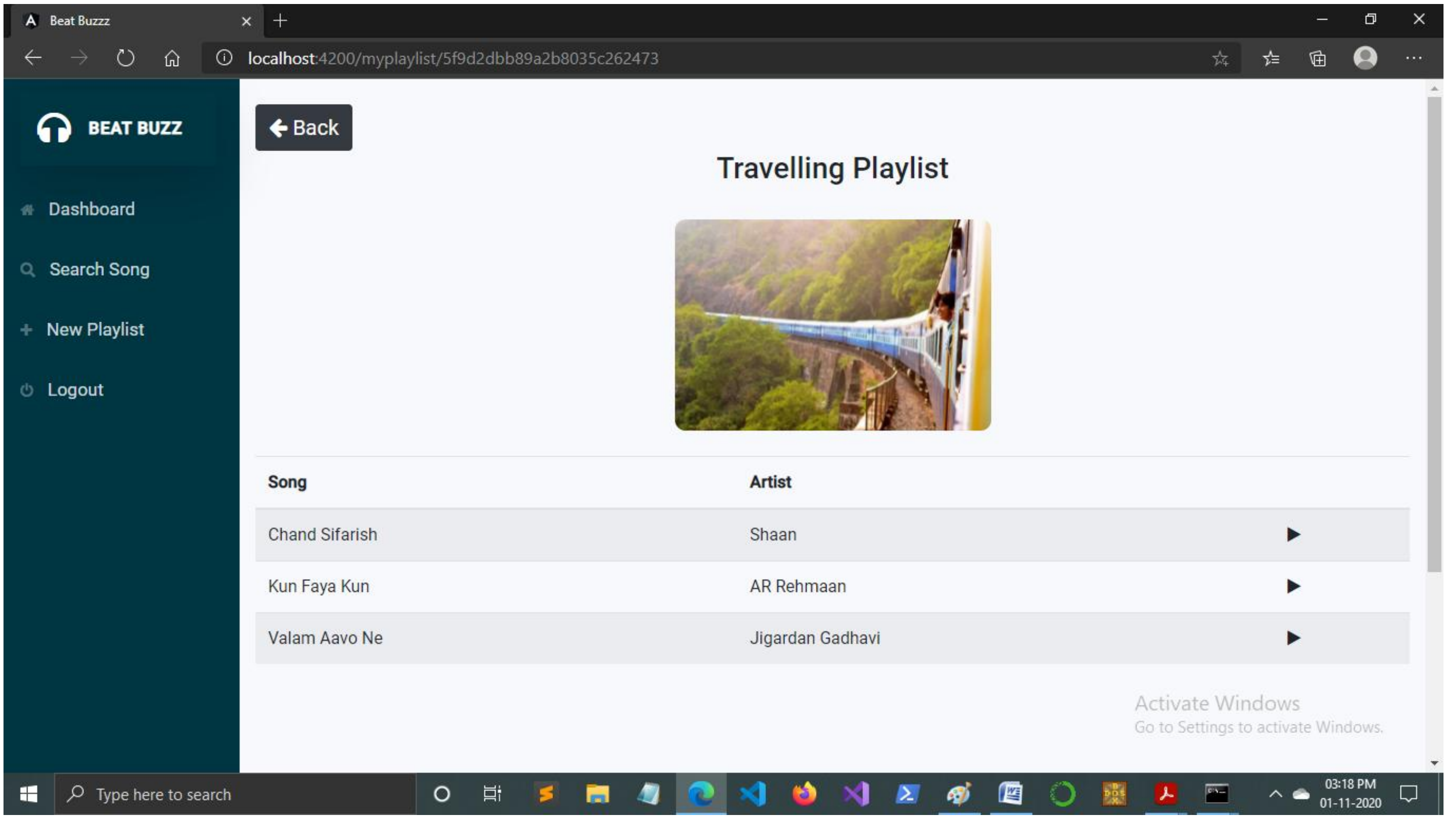
--

- User's dashboard :-



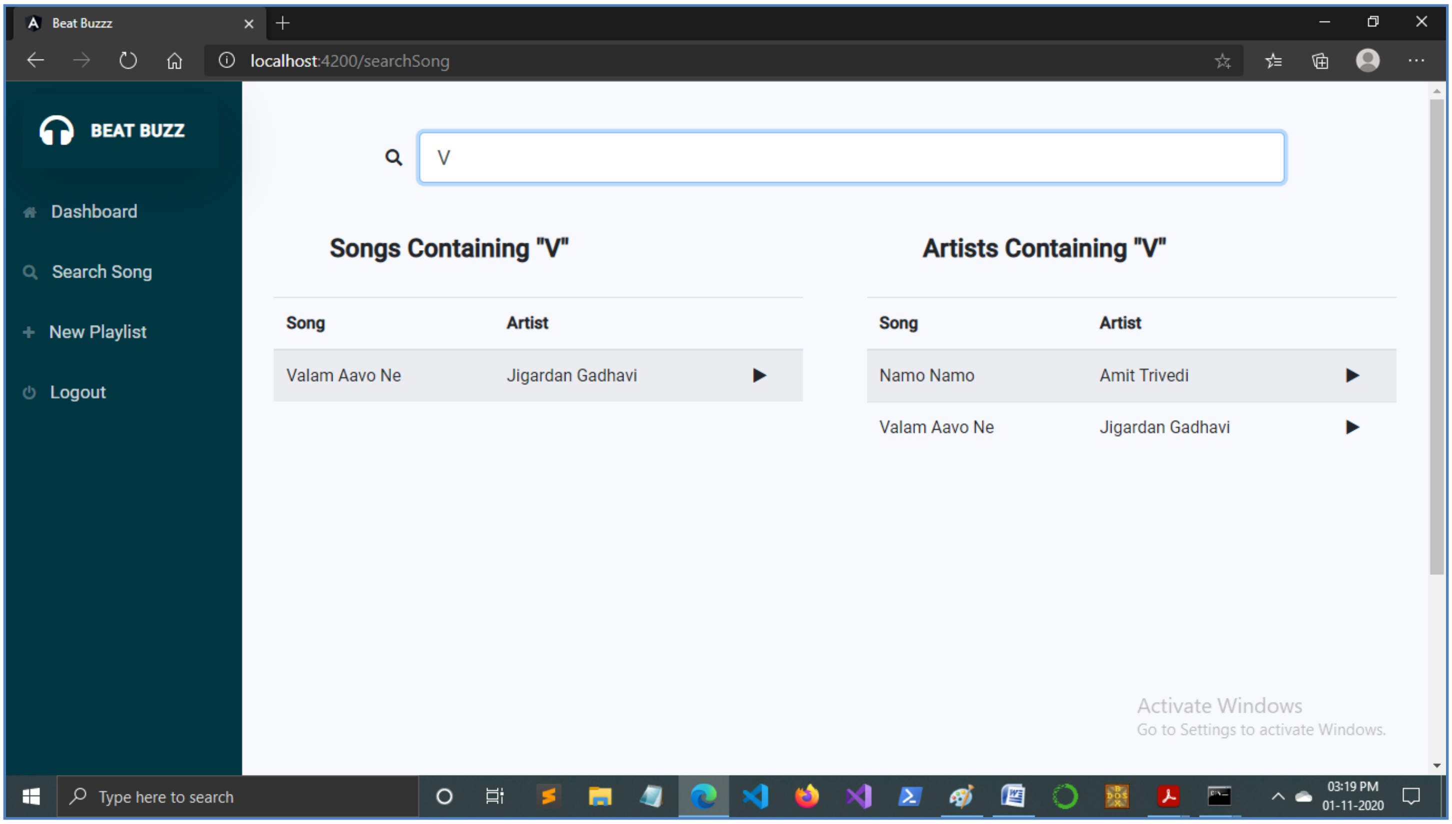
- Playlists :-





Song	Artist	
Chand Sifarish	Shaan	▶
Kun Faya Kun	AR Rehmaan	▶
Valam Aavo Ne	Jigardan Gadhavi	▶

- Search Song :-



Songs Containing "V"

Song	Artist	
Valam Aavo Ne	Jigardan Gadhavi	▶

Artists Containing "V"

Song	Artist	
Namo Namo	Amit Trivedi	▶
Valam Aavo Ne	Jigardan Gadhavi	▶

Conclusion

Login, Signup, forgot password, song upload, edit song, add category, edit categories ,View Songs, Edit songs, Create Playlist, View Playlist, View statistics – all these functionalities are implemented. Backend is implemented in nodejs and express. Data is fetched from MongoDb database. Nodejs work as api to provide data to angular's front end. Sending mail is implemented using Nodemailer. Uploading file is implemented using formidable.

Limitation and Future Extensions :-

- **Limitations :-**
  - Admin and users can’t update playlist.
  - Users can’t play entire playlist.

- **Not implemented functionality :-**
  - User can’t like songs and manage favourites.

- **Future Extensions :-**
  - Updating playlists.
  - Managing users’ favourites.
  - Playing songs in a queue.

Bibliography :-

- Stack Overflow: https://stackoverflow.com/

- W3Schools:
    - HTML5: https://www.w3schools.com/html/default.asp
    - CSS: https://www.w3schools.com/css/default.adp
    - JavaScript: https://www.w3schools.com/js/default.asp
    - Jquery: https://www.w3schools.com/jquery/default.asp
    - XML: https://www.w3schools.com/xml/default.asp
    - JSON: https://www.w3schools.com/js/js_json_intro.asp

- Bootstrap4: https://getbootstrap.com/docs/4.3/getting-started/introduction/

- Angular6: https://angular.io/docs

- Node.js: https://nodejs.org/en/

- Mongo DB: https://www.mongodb.com/