



SIMATS SCHOOL OF ENGINEERING
SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL SCIENCES
CHENNAI-602105



Spotify Application

A CAPSTONE PROJECT REPORT

Submitted in the partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by

Punith Sai. R (1922110112)

Santhosh. N. V (192211881)

Under the Supervision of

Ms.B.Jeevashri

JULY 2024

DECLARATION

We, **Punith Sai, Santhosh** students of **Bachelor of Engineering in CSE**, Department of Computer Science and Engineering, Saveetha Institute of Medical and Technical Sciences, Saveetha University, Chennai, hereby declare that the work presented in this Capstone Project Work entitled **Spotify application** is the outcome of our own bonafide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics.

1. Punith Sai. R (1922110112)

2. Santhosh. N. V (192211881)

Date: 31/07/2024

Place: Chennai

CERTIFICATE

This is to certify that the project entitled “**Spotify Application**” submitted by **Punith Sai, Santhosh** has been carried out under my supervision. The project has been submitted as per the requirements in the current semester of B.E. Computer Science Engineering.

Teacher-in-charge

Ms.B.Jeevashri

Table of Contents

S.NO	TOPICS
	Abstract
1	Introduction
2	Project Description
3	Problem Description
4	Tool Description
5	Operations
6	Approach / Module Description / Functionalities 6.1 User Authentication Module 6.2 Quiz Management 6.3 Quiz Taking 6.4 Performance Analysis 6.5 User Profile 6.6 Admin
7	Implementation
8	Result
9	Conclusion
	References

ABSTRACT

Spotify is a digital music streaming service that provides users access to millions of songs, podcasts, and other audio content from artists worldwide. Launched in 2008 by Swedish entrepreneurs Daniel Ek and Martin Lorentzon, Spotify offers both free and premium subscription options. It boasts a vast library of over 70 million songs, including tracks from major and independent labels. Users can create, share, and follow playlists, and Spotify offers curated playlists based on genres, moods, and activities. Personalized playlists like Discover Weekly and Release Radar help users discover new music tailored to their tastes.

Spotify also features a wide range of podcasts, including exclusive content. Social features allow users to follow friends, share music, and see what others are listening to in real-time. Available on various devices, including smartphones, tablets, computers, and smart speakers, Spotify supports offline listening for premium subscribers, saving data and providing access without an internet connection. The service uses advanced algorithms and machine learning to recommend music based on listening habits and preferences. Operating on a freemium model, Spotify offers a basic, ad-supported free version and a premium subscription that removes ads and provides additional features like higher sound quality and offline listening. Revenue is generated from both ads and subscriptions.

1. Introduction

Spotify is a leading digital music streaming service that revolutionizes the way we listen to music and podcasts. Launched in 2008 by Swedish entrepreneurs Daniel Ek and Martin Lorentzon, Spotify offers users instant access to a vast library of over 70 million tracks and a wide range of podcasts from artists and creators around the world. With both free and premium subscription options, Spotify caters to a diverse audience, providing personalized playlists, curated content, and advanced recommendations powered by sophisticated algorithms and machine learning.

Spotify is a leading digital music streaming service that has revolutionized the way we listen to music and podcasts. Launched in 2008 by Swedish entrepreneurs Daniel Ek and Martin Lorentzon, Spotify offers users instant access to a vast library of over 70 million tracks and a wide range of podcasts from artists and creators around the world. With both free and premium subscription options, Spotify caters to a diverse audience, providing personalized playlists, curated content, and advanced recommendations powered by sophisticated algorithms and machine learning. Available on various devices, including smartphones, tablets, computers, and smart speakers, Spotify's user-friendly interface and innovative features make it a popular choice for music and podcast enthusiasts globally.

One of Spotify's standout features is its ability to personalize the listening experience. Through features like Discover Weekly and Release Radar, users receive tailored recommendations that introduce them to new music based on their listening habits. The platform's social features allow users to share music, follow friends, and see what others are listening to, fostering a community of music lovers. Spotify also offers a wide range of curated playlists designed for various moods, activities, and genres, ensuring that there is something for everyone.

2. Project Description

"Spotify application" is a comprehensive web application developed to streamline music . The application includes:

Proposed Method

- **Frontend Development:** Utilizing Visual Studio for designing responsive and intuitive user interfaces.
- **Backend Development:** Using XAMPP stack (Apache, MySQL, PHP) to handle server-side scripting, database management via phpMyAdmin, and ensuring secure data storage and retrieval.

2.1 About my project

Purpose and Scope

- The purpose of the Spotify web application is to provide a comprehensive and user-friendly platform for streaming music and podcasts. It aims to offer users instant access to a vast library of audio content, facilitate music discovery, and create a personalized listening experience. By leveraging advanced algorithms and machine learning, the application seeks to recommend new music and podcasts based on user preferences, enhancing the overall enjoyment and engagement of its audience.

Features and Functionality:

1.User Accounts and Subscriptions:

- Free tier with ad-supported streaming.
- Premium subscription with ad-free experience, higher sound quality, and offline listening capabilities.

2.Personalization and Recommendations:

- Personalized playlists such as Discover Weekly and Release Radar.
- Curated playlists based on genres, moods, and activities.
- Advanced recommendation engine using algorithms and machine learning.

3.User Interaction and Social Features:

- Ability to create, share, and follow playlists.
- Integration with social media for sharing music and podcasts.
- Real-time activity feed to see what friends are listening to.

3. Problem Description

Existing Method

Spotify uses several innovative methods and technologies to provide a seamless and personalized music streaming experience. The platform utilizes advanced algorithms and machine learning techniques to analyze user behavior and preferences, which helps in creating personalized playlists and recommendations like Discover Weekly and Release Radar. Spotify's extensive library management system ensures users have access to over 70 million tracks and a wide range of podcasts. The application supports various user interactions, such as creating and sharing playlists, following friends, and seeing real-time updates on what others are listening to. It also offers cross-platform accessibility, allowing users to stream music on multiple devices, including smartphones, tablets, computers, and smart speakers. By combining these features with a user-friendly interface and a robust backend infrastructure, Spotify ensures a high-quality, engaging, and personalized audio streaming experience for its users.

Spotify employs a multifaceted approach to deliver a superior music streaming experience. It analyzes vast amounts of data from user interactions to understand listening habits and preferences. This data analysis allows Spotify to create highly personalized content, such as the Discover Weekly playlist, which offers users a curated selection of songs every Monday, and Release Radar, which highlights new releases tailored to individual tastes. The platform also integrates social features to enhance user engagement. Users can connect their Spotify accounts to social media platforms, enabling them to share their favorite tracks and playlists with friends. Additionally, Spotify allows users to follow friends and artists, providing updates on their latest activities and releases.

4. Tool Description

Hardware and Software Tools

To develop and deploy the spotify web application, the following hardware and software tools were utilized:

Hardware Specifications

- **Laptop Model:** ASUS ROG Strix
- **Graphics Card:** NVIDIA GeForce RTX 3060, 4GB
- **Storage:** 1TB SSD
- **RAM:** 16GB
- **Processor:** AMD Ryzen 7 6800H

The ASUS ROG Strix laptop with its high-performance specifications provided an excellent environment for developing and testing the web application. The NVIDIA GeForce RTX 3060 graphics card ensured smooth rendering of graphics and multimedia content, enhancing the development experience, especially when dealing with high-resolution recipe images and user interface design. The 1TB SSD facilitated fast data read/write operations, significantly reducing load times for development tools and ensuring rapid access to project files. With 16GB of RAM, the laptop efficiently handled multiple development tools running concurrently, supporting a seamless multitasking environment. The AMD Ryzen 7 6800H processor, known for its powerful performance and energy efficiency, enabled quick compilation and execution of code, speeding up the development cycle.

Software Tools

- **Visual Studio Code:** An integrated development environment (IDE) used for writing and debugging code. Its extensions and integrated terminal enhanced the coding experience.
- **XAMPP:** A free and open-source cross-platform web server solution stack package developed by Apache Friends. It provided the necessary Apache, MySQL, PHP, and Perl support for local development and testing.
- **phpMyAdmin:** A free software tool written in PHP, intended to handle the administration of MySQL over the web. phpMyAdmin was used for database management, allowing for easy handling of the MySQL database used in the application.
- **GitHub:** Used for version control and collaborative development. The repository hosted the project's source code, enabling team collaboration and version tracking.
- **Google Chrome:** The primary web browser used for testing and debugging the web application. Developer tools in Chrome facilitated real-time inspection and modification of the front-end code.

The combination of powerful hardware and a robust set of development tools provided a conducive environment for the efficient development, testing, and deployment of the recipe management web application.

5. Operations

The Spotify web Application provides various operations for both administrators and users to manage to listen music effectively and ensure a smooth user experience. Below are the detailed operations based on the provided code and functionalities of the application:

5.1 Administrator Operations

User Management:

- **User Registration and Authentication:** Overseeing the registration process, ensuring secure authentication mechanisms, and managing user access.
- **Account Management:** Handling user accounts, including profile updates, password resets, and account recovery processes.
- **Subscription Management:** Managing subscription plans, payments, upgrades, downgrades, and renewals.

Content Management:

- **Library Management:** Ensuring the music and podcast library is up-to-date with new releases, removing outdated or inappropriate content, and categorizing content accurately.
- **Content Curation:** Creating and updating curated playlists, ensuring diverse and engaging content is available to users.
- **Content Moderation:** Monitoring user-uploaded content and user-generated playlists for adherence to community guidelines and copyright laws.

System Maintenance:

- **Server Management:** Overseeing the performance and health of servers, ensuring high availability, and managing server resources efficiently.
- **Database Management:** Maintaining and optimizing databases to ensure fast and reliable access to user and content data.
- **Software Updates:** Regularly updating the web application with the latest features, security patches, and performance improvements.

Collaboration with Partners:

- **Artist and Label Relations:** Managing relationships with artists, record labels, and podcast creators to ensure a steady flow of new content.
- **Business Partnerships:** Collaborating with third-party service providers, advertisers, and other business partners to enhance the platform's offerings.

5.2 User Operations

Listen to music

Playback Controls:

- **Play/Pause:** Start or stop the currently playing track.
- **Skip Forward/Backward:** Move to the next or previous track.
- **Volume Control:** Adjust the playback volume.
- **Shuffle/Repeat:** Toggle shuffle and repeat modes.

Library Management:

- **Save to Library:** Add songs, albums, or playlists to your personal library.
- **Create Playlist:** Create new playlists and add tracks to them.
- **Remove from Library:** Delete tracks or albums from your library.

Managing User Profile

- **View Profile:** Users can view their profile information, including username, email, and liked recipes.
- **Edit Profile:** Users can update their profile information, such as username, email, and password.

Playlist Management:

- **Edit Playlist:** Rename or modify the order of tracks in a playlist.
- **Collaborative Playlist:** Allow friends to add or remove tracks from a shared playlist.

Share Playlist: Share playlists via social media or direct links. By structuring the operations around these roles, the Food Recipe Application provides a seamless and efficient way for administrators to manage recipes and for users to interact with and contribute to the recipe collection.

6. Approach / Module Description / Functionalities

To develop the spotify web application, we will divide the project into distinct modules, each responsible for specific functionalities. By creating individual functions for every operation and unifying them, we can ensure modularity, maintainability, and scalability.

Modules and Functionalities

6.1 User Authentication Module

Function: Register User

- **Description:** Allows new users to create an account.
- **Functionalities:**
 - Collect user information (username, email, password).
 - Validate and store user information in the database.

Function: Login User

- **Description:** Authenticates existing users.
- **Functionalities:**
 - Verify user credentials (email and password).
 - Start a session for the authenticated user.

6.2 spotify Management Module (Administrator)

- **Function: User Management:**
-
- **Add/Remove Users:** Manage user accounts, including adding new users or removing existing ones.
- **Role Assignments:** Assign different roles or permissions to users (e.g., admin, editor, viewer).
- **User Access Control:** Set and manage permissions for different users or groups.
- **Subscription Management:**
-
- **Plan Selection:** Choose and manage subscription plans that best fit the organization's needs.
- **Billing:** Handle billing information, payments, and invoices.
- **License Management:** Ensure compliance with licensing agreements and manage the number of licenses.
- **Content Management:**
-
- **Playlist Creation:** Create and manage shared playlists for the organization.
- **Content Curation:** Curate and approve content available to users, including playlists and albums.
- **Usage Monitoring:** Track which content is being accessed and by whom.

6.3 spotify Interaction Module (User)

Function: Social Features:

Share Music: Share songs, playlists, or albums with friends via social media or direct links.

Collaborative Playlists: Create and manage playlists that others can contribute to.

See What Friends Are Listening To: View activity from friends or people you follow.

Account and Settings:

Profile Management: Update your profile information, including your display name and profile picture.

Subscription Management: Manage your subscription plan, including upgrading or canceling.

Privacy Settings: Adjust who can see your activity and manage your privacy preferences.

Playback Features:

Equalizer Settings: Adjust audio settings to enhance sound quality based on personal preferences.
Crossfade and Gapless Playback: Enable features like crossfade to smooth transitions between tracks or gapless playback for seamless listening.

Offline Access:

Download Music: Download songs, albums, or playlists for offline listening.

Offline Mode: Enable offline mode to listen to downloaded content without an internet connection.

6.4 Profile Management Module (User)

Function: View Profile

- **Description:** Displays user profile information.
- **Functionalities:**
 - Retrieve and show user details (username, email, liked music).

Function: Edit Profile

- **Description:** Allows users to update their profile information.
- **Functionalities:**
 - Input new user information.
 - Validate and save updates to the database.

6.5 spotify View and Update Module (User & Admin)

Function: View Recipe Details

- **Description:** Allows users to view detailed music information.
- **Functionalities:**
 - Fetch and display music details in a modal.

Function: Update Recipe Details

- **Description:** Enables administrators to music information.
- **Functionalities:**
 - Fetch music details for editing.
 - Save updates to the database.

6.6 Admin Module (Administrator)

Function: Manage Users

- **Description:** Enables administrators to manage user accounts.
- **Functionalities:**
 - View user list.
 - Edit or delete user accounts.

Function: Manage Categories

- **Description:** Allows administrators to manage music categories.
- **Functionalities:**
 - Add, edit, or delete categories.

Integration of Functions:

By developing these modules and their respective functions independently, we can then unify them to form the complete software. Each module can interact with others through defined interfaces, ensuring smooth data flow and cohesive operation.

Example: Unifying Functions

1. **User Login:**
 - User logs in using the Login User function from the User Authentication Module.
 - Based on the role (user/admin), the user is redirected to their respective dashboard.
2. **Admin Dashboard:**
 - Administrators can access Recipe Management Module functions (Add music, Edit music, etc.) from their dashboard.
 - They can also use the Admin Module to manage users and categories.
3. **User Dashboard:**
 - Users can view and interact with recipes using the music Interaction Module.

- Users can also manage their profiles using functions from the Profile Management Module.
4. **Profile Management:**
- Both administrators and users can manage their profiles using functions from the Profile Management Module.

By structuring the operations and functionalities around these modules, the Food Recipe Application ensures a seamless and efficient user experience for both administrators and users.

7. Implementation/Coding

Index code

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Spotify Login</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      background-image: url(login.png);
      display: flex;
      justify-content: flex-start; /* Aligns the container to the left */
      align-items: center;
      height: 100vh;
      margin: 0;
      padding-left: 2rem; /* Adds some padding from the left edge */
    }

    .login-container {
      background-color: white;
      padding: 2rem;
      border-radius: 8px;
      box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
      width: 300px;
      text-align: center;
    }

    .login-container h1 {
      margin-bottom: 1.5rem;
      color: #1db954;
    }

    .input-group {
      margin-bottom: 1.5rem;
    }

    .input-group label {
      display: block;
      margin-bottom: 0.5rem;
      font-weight: bold;
    }

    .input-group input {
      width: 100%;
```



```

padding: 0.5rem;
border: 1px solid #ccc;
border-radius: 4px;
}

button {
background-color: #1db954;
color: white;
border: none;
padding: 0.75rem;
border-radius: 20px;
cursor: pointer;
width: 100%;
}

button:hover {
background-color: #17a443;
}
</style>
</head>
<body>
<div class="login-container">
<h1>Spotify</h1>
<form action="/login" method="POST">
<div class="input-group">
<label for="username">Username or Email</label>
<input type="text" id="username" name="username" required>
</div>
<div class="input-group">
<label for="password">Password</label>
<input type="password" id="password" name="password" required>
</div>
<button type="submit" onclick="document.location='homepage.html'">Log
In</button>
</form>
</div>
</body>
</html> <!--DOCTYPE html-->
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Spotify Home</title>
<style>
body {
font-family: Arial, sans-serif;

```

```
margin: 0;
padding: 0;
background-image: url(mm.jpeg);
background-size: cover;
background-repeat: no-repeat;
background-attachment: fixed;
color: #333;
}

.navbar {
  background-color: rgba(29, 185, 84, 0.9);
  padding: 1rem;
  color: white;
  display: flex;
  justify-content: space-between;
  align-items: center;
  position: fixed;
  width: 100%;
  top: 0;
  z-index: 1000;
}

.navbar .logo {
  font-size: 1.5rem;
  font-weight: bold;
}

.navbar ul {
  list-style: none;
  display: flex;
  margin: 0;
  padding: 0;
}

.navbar ul li {
  margin-left: 1.5rem;
}

.navbar ul li a {
  color: white;
  text-decoration: none;
  font-weight: bold;
}

.hero {
  background-image: url(hero.jpg);
```

```
background-size: cover;
background-position: center;
color: white;
padding: 5rem 2rem;
text-align: center;
margin-top: 60px;
}

.hero h1 {
  margin: 0;
  font-size: 4rem;
  text-shadow: 2px 2px 8px rgba(0, 0, 0, 0.7);
}

.hero p {
  font-size: 1.5rem;
  text-shadow: 2px 2px 8px rgba(0, 0, 0, 0.7);
}

.content {
  padding: 2rem;
  background-color: rgba(255, 255, 255, 0.8);
  border-radius: 8px;
  box-shadow: 0 0 20px rgba(0, 0, 0, 0.2);
  margin: 2rem;
}

.content h2 {
  font-size: 2rem;
  margin-bottom: 1rem;
  text-align: center;
  color: #1db954;
}

.playlist {
  display: flex;
  gap: 1.5rem;
  flex-wrap: wrap;
  justify-content: center;
}

.playlist-item {
  background-color: white;
  border-radius: 8px;
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
  width: calc(30% - 1rem);
```

```
    margin-bottom: 1.5rem;
    padding: 1rem;
    text-align: center;
    transition: transform 0.2s, box-shadow 0.2s;
}

.playlist-item:hover {
    transform: translateY(-10px);
    box-shadow: 0 0 20px rgba(0, 0, 0, 0.2);
}

.playlist-item img {
    width: 100%;
    border-radius: 8px;
}

.playlist-item h3 {
    margin: 0.5rem 0;
    color: #1db954;
}

.playlist-item p {
    color: #666;
}

.playlist-item button {
    background-color: #1db954;
    color: white;
    border: none;
    padding: 0.75rem 1.5rem;
    border-radius: 4px;
    cursor: pointer;
    transition: background-color 0.3s, transform 0.3s;
    margin-top: 1rem;
}

.playlist-item button:hover {
    background-color: #17a443;
}

.playlist-item button:active {
    transform: scale(0.95);
}
</style>
</head>
<body>
```

```

<div class="navbar">
  <div class="logo">Spotify</div>
  <ul>
    <li><a href="#home">Home</a></li>
    <li><a href="#browse">Browse</a></li>
    <li><a href="#library">Library</a></li>
    <li><a href="#premium">Premium</a></li>
  </ul>
</div>
<div class="hero">
  <h1>Welcome to Spotify</h1>
  <p>Listen to your favorite music anytime, anywhere.</p>
</div>
<div class="content">
  <h2>Featured Playlists</h2>
  <div class="playlist">
    <div class="playlist-item">
      
      <h3>Top Hits</h3>
      <p>Enjoy the top hits from around the world.</p>
      <button type="submit" onclick="document.location='topplaylist.html'">Top
Hits</button>
    </div>
    <div class="playlist-item">
      
      <h3>Chill Vibes</h3>
      <p>Relax and unwind with these chill tracks.</p>
      <button type="submit" onclick="document.location='Hangover.html'">Chill
Vibes</button>
    </div>
    <div class="playlist-item">
      
      <h3>Workout</h3>
      <p>Get pumped up with this workout playlist.</p>
      <button
type="submit"
onclick="document.location='homepage.html'">Workout</button>
    </div>
  </div>
</div>
</body>
</html>e<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Spotify Top Playlists</title>

```

```
<style>
  body {
    font-family: Arial, sans-serif;
    margin: 0;
    padding: 0;
    background-color: #f5f5f5;
    color: #333;
  }

  .navbar {
    background-color: #1db954;
    padding: 1rem;
    color: white;
    display: flex;
    justify-content: space-between;
    align-items: center;
    position: fixed;
    width: 100%;
    top: 0;
    z-index: 1000;
  }

  .navbar .logo {
    font-size: 1.5rem;
    font-weight: bold;
  }

  .navbar ul {
    list-style: none;
    display: flex;
    margin: 0;
    padding: 0;
  }

  .navbar ul li {
    margin-left: 1.5rem;
  }

  .navbar ul li a {
    color: white;
    text-decoration: none;
    font-weight: bold;
  }

  .header {
    background-color: #1db954;
```

```
    color: white;
    padding: 3rem 2rem;
    text-align: center;
    margin-top: 60px;
}

.header h1 {
    margin: 0;
    font-size: 3rem;
}

.content {
    padding: 2rem;
    margin-top: 20px;
}

.content h2 {
    font-size: 2rem;
    margin-bottom: 1rem;
    text-align: center;
    color: #1db954;
}

.playlist {
    display: flex;
    flex-wrap: wrap;
    gap: 1rem;
    justify-content: center;
}

.playlist-item {
    background-color: white;
    border-radius: 8px;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    width: calc(30% - 1rem);
    margin-bottom: 1rem;
    padding: 1rem;
    text-align: center;
    transition: transform 0.2s, box-shadow 0.2s;
}

.playlist-item:hover {
    transform: translateY(-10px);
    box-shadow: 0 0 20px rgba(0, 0, 0, 0.2);
}
```

```

.playlist-item img {
  width: 100%;
  border-radius: 8px;
}

.playlist-item h3 {
  margin: 0.5rem 0;
  color: #1db954;
}

.playlist-item p {
  color: #666;
}

.playlist-item button {
  background-color: #1db954;
  color: white;
  border: none;
  padding: 0.5rem 1rem;
  font-size: 1rem;
  border-radius: 5px;
  cursor: pointer;
  transition: background-color 0.2s, transform 0.2s;
}

.playlist-item button:hover {
  background-color: #17a148;
  transform: translateY(-2px);
}

```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div class="navbar">
```

```
<div class="logo">Spotify</div>
```

```
<ul>
```

```
<li><a href="#home">Home</a></li>
```

```
<li><a href="#browse">Browse</a></li>
```

```
<li><a href="#library">Library</a></li>
```

```
<li><a href="#premium">Premium</a></li>
```

```
</ul>
```

```
</div>
```

```
<div class="header">
```

```
<h1>Top Playlists</h1>
```

```
</div>
```

```
<div class="content">
```

```
<h2>Discover the Best Playlists</h2>
```



```

    <div class="playlist">
      <div class="playlist-item">
        
        <h3>Kalki Song</h3>
        <button type="submit" onclick="document.location='kalki.html'">Play
Now</button>
      </div>
      <div class="playlist-item">
        
        <h3>Arjun Reddy</h3>
        <button type="submit" onclick="document.location='arjun.html'">Play
Now</button>
      </div>
      <div class="playlist-item">
        
        <h3>Saaho</h3>
        <button type="submit" onclick="document.location='saaho.html'">Play
Now</button>
      </div>
      <div class="playlist-item">
        
        <h3>Devara</h3>
        <button type="submit" onclick="document.location='devara.html'">Play
Now</button>
      </div>
      <div class="playlist-item">
        
        <h3>RRR</h3>
        <button type="submit" onclick="document.location='rrr.html'">Play Now</button>
      </div>
      <div class="playlist-item">
        
        <h3>DJ TILLU</h3>
        <button type="submit" onclick="document.location='dj.html'">Play Now</button>
      </div>
    </div>
  </div>
</body>
</html>
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Melody Songs</title>
  <style>

```

```
body {
  font-family: Arial, sans-serif;
  margin: 0;
  padding: 0;
  background-color: #f5f5f5;
  color: #333;
}

.navbar {
  background-color: #1db954;
  padding: 1rem;
  color: rgb(28, 26, 26);
  display: flex;
  justify-content: space-between;
  align-items: center;
  position: fixed;
  width: 100%;
  top: 0;
  z-index: 1000;
}

.navbar .logo {
  font-size: 1.5rem;
  font-weight: bold;
}

.navbar ul {
  list-style: none;
  display: flex;
  margin: 0;
  padding: 0;
}

.navbar ul li {
  margin-left: 1.5rem;
}

.navbar ul li a {
  color: white;
  text-decoration: none;
  font-weight: bold;
}

.header {
  background-color: #1db954;
  color: white;
```

```
padding: 3rem 2rem;
text-align: center;
margin-top: 60px;
}

.header h1 {
margin: 0;
font-size: 3rem;
}

.content {
padding: 2rem;
margin-top: 20px;
}

.content h2 {
font-size: 2rem;
margin-bottom: 1rem;
text-align: center;
color: #1db954;
}

.song-list {
display: flex;
flex-wrap: wrap;
gap: 1rem;
justify-content: center;
}

.song-item {
background-color: white;
border-radius: 8px;
box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
width: calc(30% - 1rem);
margin-bottom: 1rem;
padding: 1rem;
text-align: center;
transition: transform 0.2s, box-shadow 0.2s;
}

.song-item:hover {
transform: translateY(-10px);
box-shadow: 0 0 20px rgba(0, 0, 0, 0.2);
}

.song-item img {
```

```

        width: 100%;
        border-radius: 8px;
    }

    .song-item h3 {
        margin: 0.5rem 0;
        color: #1db954;
    }

    .song-item p {
        color: #666;
    }

    .song-item button {
        background-color: #1db954;
        color: white;
        border: none;
        padding: 0.5rem 1rem;
        font-size: 1rem;
        border-radius: 5px;
        cursor: pointer;
        transition: background-color 0.2s, transform 0.2s;
    }

    .song-item button:hover {
        background-color: #17a148;
        transform: translateY(-2px);
    }
</style>
</head>
<body>
    <div class="navbar">
        <div class="logo">Spotify</div>
        <ul>
            <li><a href="#home">Home</a></li>
            <li><a href="#browse">Browse</a></li>
            <li><a href="#library">Library</a></li>
            <li><a href="#premium">Premium</a></li>
        </ul>
    </div>
    <div class="header">
        <h1>Hangover cure</h1>
    </div>
    <div class="content">
        <h2>Enjoy the Best Melody Songs</h2>
        <div class="song-list">

```

```

        <div class="song-item">
            
            <h3>illahi</h3>
            <p>Artist 1</p>
            <button      type="submit"      onclick="document.location='song1.html'">Play
Now</button>
        </div>
        <div class="song-item">
            
            <h3>ankhon se batana</h3>
            <p>Artist 2</p>
            <button      type="submit"      onclick="document.location='song2.html'">Play
Now</button>
        </div>
        <div class="song-item">
            <img src="" alt="Song 3">
            <h3>khariyat</h3>
            <p>Artist 3</p>
            <button      type="submit"      onclick="document.location='song3.html'">Play
Now</button>
        </div>
    </div>
</div>
</body>
</html>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>dj tillu - Song Page</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            margin: 0;
            padding: 0;
            background-color: #f5f5f5;
            color: #333;
        }

        .navbar {
            background-color: #1db954;
            padding: 1rem;
            color: white;
            display: flex;
            justify-content: space-between;

```

```
    align-items: center;
    position: fixed;
    width: 100%;
    top: 0;
    z-index: 1000;
}
```

```
.navbar .logo {
    font-size: 1.5rem;
    font-weight: bold;
}
```

```
.navbar ul {
    list-style: none;
    display: flex;
    margin: 0;
    padding: 0;
}
```

```
.navbar ul li {
    margin-left: 1.5rem;
}
```

```
.navbar ul li a {
    color: white;
    text-decoration: none;
    font-weight: bold;
}
```

```
.header {
    background-color: #1db954;
    color: white;
    padding: 3rem 2rem;
    text-align: center;
    margin-top: 60px;
}
```

```
.header h1 {
    margin: 0;
    font-size: 3rem;
}
```

```
.content {
    padding: 2rem;
    margin-top: 20px;
}
```

```
.content h2 {  
  font-size: 2rem;  
  margin-bottom: 1rem;  
  text-align: center;  
  color: #1db954;  
}
```

```
.song-details {  
  background-color: white;  
  padding: 1rem;  
  border-radius: 8px;  
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);  
  text-align: center;  
  max-width: 600px;  
  margin: 0 auto;  
}
```

```
.song-details img {  
  width: 100%;  
  border-radius: 8px;  
}
```

```
.song-details h3 {  
  color: #1db954;  
  margin: 1rem 0;  
}
```

```
.song-details p {  
  color: #666;  
}
```

```
.lyrics {  
  background-color: white;  
  padding: 1rem;  
  border-radius: 8px;  
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);  
  max-width: 600px;  
  margin: 2rem auto;  
}
```

```
.lyrics h3 {  
  color: #1db954;  
  margin-bottom: 1rem;  
  text-align: center;  
}
```

```

        .lyrics pre {
            white-space: pre-wrap;
            font-family: inherit;
            color: #333;
        }
    </style>
</head>
<body>
    <div class="navbar">
        <div class="logo">Spotify</div>
        <ul>
            <li><a href="#home">Home</a></li>
            <li><a href="#browse">Browse</a></li>
            <li><a href="#library">Library</a></li>
            <li><a href="#premium">Premium</a></li>
        </ul>
    </div>
    <div class="header">
        <h1>dj tillu</h1>
    </div>
    <div class="content">
        <div class="song-details">
            
            <h3>dj</h3>
            <p>Artist: [punith]</p>
            <p>Album: [Album Name]</p>
            <p>Released: [Release Date]</p>
        </div>
        <div class="lyrics">
            <h3>Lyrics</h3>
            <pre>
                [Lyrics of the song "dj tillu" go here]
            </pre>
        </div>
    </div>
</body>
</html>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Devara - Song Page</title>
    <style>
        body {

```



```
font-family: Arial, sans-serif;
margin: 0;
padding: 0;
background-color: #f5f5f5;
color: #333;
}

.navbar {
  background-color: #1db954;
  padding: 1rem;
  color: white;
  display: flex;
  justify-content: space-between;
  align-items: center;
  position: fixed;
  width: 100%;
  top: 0;
  z-index: 1000;
}

.navbar .logo {
  font-size: 1.5rem;
  font-weight: bold;
}

.navbar ul {
  list-style: none;
  display: flex;
  margin: 0;
  padding: 0;
}

.navbar ul li {
  margin-left: 1.5rem;
}

.navbar ul li a {
  color: white;
  text-decoration: none;
  font-weight: bold;
}

.header {
  background-color: #1db954;
  color: white;
  padding: 3rem 2rem;
```

```
    text-align: center;
    margin-top: 60px;
}
```

```
.header h1 {
    margin: 0;
    font-size: 3rem;
}
```

```
.content {
    padding: 2rem;
    margin-top: 20px;
}
```

```
.content h2 {
    font-size: 2rem;
    margin-bottom: 1rem;
    text-align: center;
    color: #1db954;
}
```

```
.song-details {
    background-color: white;
    padding: 1rem;
    border-radius: 8px;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    text-align: center;
    max-width: 600px;
    margin: 0 auto;
}
```

```
.song-details img {
    width: 100%;
    border-radius: 8px;
}
```

```
.song-details h3 {
    color: #1db954;
    margin: 1rem 0;
}
```

```
.song-details p {
    color: #666;
}
```

```
.lyrics {
```

```

        background-color: white;
        padding: 1rem;
        border-radius: 8px;
        box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
        max-width: 600px;
        margin: 2rem auto;
    }

    .lyrics h3 {
        color: #1db954;
        margin-bottom: 1rem;
        text-align: center;
    }

    .lyrics pre {
        white-space: pre-wrap;
        font-family: inherit;
        color: #333;
    }
</style>
</head>
<body>
    <div class="navbar">
        <div class="logo">Spotify</div>
        <ul>
            <li><a href="#home">Home</a></li>
            <li><a href="#browse">Browse</a></li>
            <li><a href="#library">Library</a></li>
            <li><a href="#premium">Premium</a></li>
        </ul>
    </div>
    <div class="header">
        <h1>Devara</h1>
    </div>
    <div class="content">
        <div class="song-details">
            
            <h3>Devara</h3>
            <p>Artist: [punith]</p>
            <p>Album: [Album Name]</p>
            <p>Released: [Release Date]</p>
        </div>
        <div class="lyrics">
            <h3>Lyrics</h3>
            <pre>
                [Lyrics of the song "Devara" go here]
            </pre>
        </div>
    </div>
</body>
</html>

```

```
        </pre>
    </div>
</div>
</body>
</html>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Kalki - Song Page</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            margin: 0;
            padding: 0;
            background-color: #f5f5f5;
            color: #333;
        }

        .navbar {
            background-color: #1db954;
            padding: 1rem;
            color: white;
            display: flex;
            justify-content: space-between;
            align-items: center;
            position: fixed;
            width: 100%;
            top: 0;
            z-index: 1000;
        }

        .navbar .logo {
            font-size: 1.5rem;
            font-weight: bold;
        }

        .navbar ul {
            list-style: none;
            display: flex;
            margin: 0;
            padding: 0;
        }

        .navbar ul li {
```

```
    margin-left: 1.5rem;
}

.navbar ul li a {
    color: white;
    text-decoration: none;
    font-weight: bold;
}

.header {
    background-color: #1db954;
    color: white;
    padding: 3rem 2rem;
    text-align: center;
    margin-top: 60px;
}

.header h1 {
    margin: 0;
    font-size: 3rem;
}

.content {
    padding: 2rem;
    margin-top: 20px;
}

.content h2 {
    font-size: 2rem;
    margin-bottom: 1rem;
    text-align: center;
    color: #1db954;
}

.song-details {
    background-color: white;
    padding: 1rem;
    border-radius: 8px;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    text-align: center;
    max-width: 600px;
    margin: 0 auto;
}

.song-details img {
    width: 100%;
}
```

```

        border-radius: 8px;
    }

    .song-details h3 {
        color: #1db954;
        margin: 1rem 0;
    }

    .song-details p {
        color: #666;
    }

    .lyrics {
        background-color: white;
        padding: 1rem;
        border-radius: 8px;
        box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
        max-width: 600px;
        margin: 2rem auto;
    }

    .lyrics h3 {
        color: #1db954;
        margin-bottom: 1rem;
        text-align: center;
    }

    .lyrics pre {
        white-space: pre-wrap;
        font-family: inherit;
        color: #333;
    }
</style>
</head>
<body>
    <div class="navbar">
        <div class="logo">Spotify</div>
        <ul>
            <li><a href="#home">Home</a></li>
            <li><a href="#browse">Browse</a></li>
            <li><a href="#library">Library</a></li>
            <li><a href="#premium">Premium</a></li>
        </ul>
    </div>
    <div class="header">
        <h1>Kalki</h1>

```

```
</div>
<div class="content">
  <div class="song-details">
    
    <h3>Kalki</h3>
    <p>Artist: [punith]</p>
    <p>Album: [Album Name]</p>
    <p>Released: [Release Date]</p>
  </div>
  <div class="lyrics">
    <h3>Lyrics</h3>
    <pre>
      [Lyrics of the song "Kalki" go here]
    </pre>
  </div>
</div>
</body>
</html>

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title> arjun reddy - Song Page</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 0;
      padding: 0;
      background-color: #f5f5f5;
      color: #333;
    }

    .navbar {
      background-color: #1db954;
      padding: 1rem;
      color: white;
      display: flex;
      justify-content: space-between;
      align-items: center;
      position: fixed;
      width: 100%;
      top: 0;
      z-index: 1000;
    }
  </style>
</head>
<body>
  <div class="navbar">
    <div class="container">
      <div class="row">
        <div class="col">
          <div class="img">
            <img alt="Profile picture of Arjun Reddy" data-bbox="100 100 150 150"/>
          </div>
          <div class="text">
            <h1>Arjun Reddy</h1>
            <p>Singer, Musician</p>
          </div>
        </div>
        <div class="col">
          <div class="text">
            <h2>Song Page</h2>
            <p>Welcome to the official song page of Arjun Reddy. Here you can find all the songs he has released, along with lyrics and audio players. Stay tuned for more updates on his music journey.</p>
          </div>
          <div class="list">
            <ul>
              <li>Song 1</li>
              <li>Song 2</li>
              <li>Song 3</li>
            </ul>
          </div>
        </div>
      </div>
    </div>
  </div>
  <div class="content">
    <div class="song-details">
      
      <h3>Kalki</h3>
      <p>Artist: [punith]</p>
      <p>Album: [Album Name]</p>
      <p>Released: [Release Date]</p>
    </div>
    <div class="lyrics">
      <h3>Lyrics</h3>
      <pre>
        [Lyrics of the song "Kalki" go here]
      </pre>
    </div>
  </div>
</div>
</body>
</html>
```

```
.navbar .logo {  
  font-size: 1.5rem;  
  font-weight: bold;  
}
```

```
.navbar ul {  
  list-style: none;  
  display: flex;  
  margin: 0;  
  padding: 0;  
}
```

```
.navbar ul li {  
  margin-left: 1.5rem;  
}
```

```
.navbar ul li a {  
  color: white;  
  text-decoration: none;  
  font-weight: bold;  
}
```

```
.header {  
  background-color: #1db954;  
  color: white;  
  padding: 3rem 2rem;  
  text-align: center;  
  margin-top: 60px;  
}
```

```
.header h1 {  
  margin: 0;  
  font-size: 3rem;  
}
```

```
.content {  
  padding: 2rem;  
  margin-top: 20px;  
}
```

```
.content h2 {  
  font-size: 2rem;  
  margin-bottom: 1rem;  
  text-align: center;  
  color: #1db954;
```



```
}
```

```
.song-details {  
  background-color: white;  
  padding: 1rem;  
  border-radius: 8px;  
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);  
  text-align: center;  
  max-width: 600px;  
  margin: 0 auto;  
}
```

```
.song-details img {  
  width: 100%;  
  border-radius: 8px;  
}
```

```
.song-details h3 {  
  color: #1db954;  
  margin: 1rem 0;  
}
```

```
.song-details p {  
  color: #666;  
}
```

```
.lyrics {  
  background-color: white;  
  padding: 1rem;  
  border-radius: 8px;  
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);  
  max-width: 600px;  
  margin: 2rem auto;  
}
```

```
.lyrics h3 {  
  color: #1db954;  
  margin-bottom: 1rem;  
  text-align: center;  
}
```

```
.lyrics pre {  
  white-space: pre-wrap;  
  font-family: inherit;  
  color: #333;  
}
```

```
</style>
</head>
<body>
  <div class="navbar">
    <div class="logo">Spotify</div>
    <ul>
      <li><a href="#home">Home</a></li>
      <li><a href="#browse">Browse</a></li>
      <li><a href="#library">Library</a></li>
      <li><a href="#premium">Premium</a></li>
    </ul>
  </div>
  <div class="header">
    <h1>arjun reddy</h1>
  </div>
  <div class="content">
    <div class="song-details">
      
      <h3>arjun reddy</h3>
      <p>Artist: [punith]</p>
      <p>Album: [Album Name]</p>
      <p>Released: [Release Date]</p>
    </div>
    <div class="lyrics">
      <h3>Lyrics</h3>
      <pre>
        [Lyrics of the song "arjun reddy" go here]
      </pre>
    </div>
  </div>
</body>
</html>
```

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>saaho - Song Page</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 0;
      padding: 0;
      background-color: #f5f5f5;
    }
  </style>
</head>
<body>
  <div class="navbar">
    <div class="logo">Spotify</div>
    <ul>
      <li><a href="#home">Home</a></li>
      <li><a href="#browse">Browse</a></li>
      <li><a href="#library">Library</a></li>
      <li><a href="#premium">Premium</a></li>
    </ul>
  </div>
  <div class="header">
    <h1>arjun reddy</h1>
  </div>
  <div class="content">
    <div class="song-details">
      
      <h3>arjun reddy</h3>
      <p>Artist: [punith]</p>
      <p>Album: [Album Name]</p>
      <p>Released: [Release Date]</p>
    </div>
    <div class="lyrics">
      <h3>Lyrics</h3>
      <pre>
        [Lyrics of the song "arjun reddy" go here]
      </pre>
    </div>
  </div>
</body>
</html>
```

```
    color: #333;
}

.navbar {
    background-color: #1db954;
    padding: 1rem;
    color: white;
    display: flex;
    justify-content: space-between;
    align-items: center;
    position: fixed;
    width: 100%;
    top: 0;
    z-index: 1000;
}

.navbar .logo {
    font-size: 1.5rem;
    font-weight: bold;
}

.navbar ul {
    list-style: none;
    display: flex;
    margin: 0;
    padding: 0;
}

.navbar ul li {
    margin-left: 1.5rem;
}

.navbar ul li a {
    color: white;
    text-decoration: none;
    font-weight: bold;
}

.header {
    background-color: #1db954;
    color: white;
    padding: 3rem 2rem;
    text-align: center;
    margin-top: 60px;
}
```

```
.header h1 {  
  margin: 0;  
  font-size: 3rem;  
}
```

```
.content {  
  padding: 2rem;  
  margin-top: 20px;  
}
```

```
.content h2 {  
  font-size: 2rem;  
  margin-bottom: 1rem;  
  text-align: center;  
  color: #1db954;  
}
```

```
.song-details {  
  background-color: white;  
  padding: 1rem;  
  border-radius: 8px;  
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);  
  text-align: center;  
  max-width: 600px;  
  margin: 0 auto;  
}
```

```
.song-details img {  
  width: 100%;  
  border-radius: 8px;  
}
```

```
.song-details h3 {  
  color: #1db954;  
  margin: 1rem 0;  
}
```

```
.song-details p {  
  color: #666;  
}
```

```
.lyrics {  
  background-color: white;  
  padding: 1rem;  
  border-radius: 8px;  
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);  
}
```

```
        max-width: 600px;
        margin: 2rem auto;
    }

    .lyrics h3 {
        color: #1db954;
        margin-bottom: 1rem;
        text-align: center;
    }

    .lyrics pre {
        white-space: pre-wrap;
        font-family: inherit;
        color: #333;
    }
</style>
</head>
<body>
    <div class="navbar">
        <div class="logo">Spotify</div>
        <ul>
            <li><a href="#home">Home</a></li>
            <li><a href="#browse">Browse</a></li>
            <li><a href="#library">Library</a></li>
            <li><a href="#premium">Premium</a></li>
        </ul>
    </div>
    <div class="header">
        <h1>saaho</h1>
    </div>
    <div class="content">
        <div class="song-details">
            
            <h3>saaho</h3>
            <p>Artist: [punith]</p>
            <p>Album: [Album Name]</p>
            <p>Released: [Release Date]</p>
        </div>
        <div class="lyrics">
            <h3>lyrics</h3>
            <pre>
                [Lyrics of the song "saaho" go here]
            </pre>
        </div>
    </div>
</body>
```

</html>

```

    <!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>illahi - Song Page</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 0;
      padding: 0;
      background-color: #f5f5f5;
      color: #333;
    }

    .navbar {
      background-color: #1db954;
      padding: 1rem;
      color: white;
      display: flex;
      justify-content: space-between;
      align-items: center;
      position: fixed;
      width: 100%;
      top: 0;
      z-index: 1000;
    }

    .navbar .logo {
      font-size: 1.5rem;
      font-weight: bold;
    }

    .navbar ul {
      list-style: none;
      display: flex;
      margin: 0;
      padding: 0;
    }

    .navbar ul li {
      margin-left: 1.5rem;
    }
  </style>

```

```
.navbar ul li a {  
  color: white;  
  text-decoration: none;  
  font-weight: bold;  
}
```

```
.header {  
  background-color: #1db954;  
  color: white;  
  padding: 3rem 2rem;  
  text-align: center;  
  margin-top: 60px;  
}
```

```
.header h1 {  
  margin: 0;  
  font-size: 3rem;  
}
```

```
.content {  
  padding: 2rem;  
  margin-top: 20px;  
}
```

```
.content h2 {  
  font-size: 2rem;  
  margin-bottom: 1rem;  
  text-align: center;  
  color: #1db954;  
}
```

```
.song-details {  
  background-color: white;  
  padding: 1rem;  
  border-radius: 8px;  
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);  
  text-align: center;  
  max-width: 600px;  
  margin: 0 auto;  
}
```

```
.song-details img {  
  width: 100%;  
  border-radius: 8px;  
}
```

```
.song-details h3 {
  color: #1db954;
  margin: 1rem 0;
}

.song-details p {
  color: #666;
}

.lyrics {
  background-color: white;
  padding: 1rem;
  border-radius: 8px;
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
  max-width: 600px;
  margin: 2rem auto;
}

.lyrics h3 {
  color: #1db954;
  margin-bottom: 1rem;
  text-align: center;
}

.lyrics pre {
  white-space: pre-wrap;
  font-family: inherit;
  color: #333;
}
</style>
</head>
<body>
<div class="navbar">
  <div class="logo">Spotify</div>
  <ul>
    <li><a href="#home">Home</a></li>
    <li><a href="#browse">Browse</a></li>
    <li><a href="#library">Library</a></li>
    <li><a href="#premium">Premium</a></li>
  </ul>
</div>
<div class="header">
  <h1>illahi</h1>
</div>
<div class="content">
```



```
<div class="song-details">
  
  <h3>illahi</h3>
  <p>Artist: [Arjith singh]</p>
  <p>Album: [Album Name]</p>
  <p>Released: [Release Date]</p>
</div>
<div class="lyrics">
  <h3>lyrics</h3>
  <pre>
    [Lyrics of the song "illahi" go here]
  </pre>
</div>
</div>
</body>
</html>
```

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>ankhon se - Song Page</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 0;
      padding: 0;
      background-color: #f5f5f5;
      color: #333;
    }

    .navbar {
      background-color: #1db954;
      padding: 1rem;
      color: white;
      display: flex;
      justify-content: space-between;
      align-items: center;
      position: fixed;
      width: 100%;
      top: 0;
      z-index: 1000;
    }
```

```
.navbar .logo {  
  font-size: 1.5rem;  
  font-weight: bold;  
}
```

```
.navbar ul {  
  list-style: none;  
  display: flex;  
  margin: 0;  
  padding: 0;  
}
```

```
.navbar ul li {  
  margin-left: 1.5rem;  
}
```

```
.navbar ul li a {  
  color: white;  
  text-decoration: none;  
  font-weight: bold;  
}
```

```
.header {  
  background-color: #1db954;  
  color: white;  
  padding: 3rem 2rem;  
  text-align: center;  
  margin-top: 60px;  
}
```

```
.header h1 {  
  margin: 0;  
  font-size: 3rem;  
}
```

```
.content {  
  padding: 2rem;  
  margin-top: 20px;  
}
```

```
.content h2 {  
  font-size: 2rem;  
  margin-bottom: 1rem;  
  text-align: center;  
  color: #1db954;  
}
```

```
.song-details {
  background-color: white;
  padding: 1rem;
  border-radius: 8px;
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
  text-align: center;
  max-width: 600px;
  margin: 0 auto;
}

.song-details img {
  width: 100%;
  border-radius: 8px;
}

.song-details h3 {
  color: #1db954;
  margin: 1rem 0;
}

.song-details p {
  color: #666;
}

.lyrics {
  background-color: white;
  padding: 1rem;
  border-radius: 8px;
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
  max-width: 600px;
  margin: 2rem auto;
}

.lyrics h3 {
  color: #1db954;
  margin-bottom: 1rem;
  text-align: center;
}

.lyrics pre {
  white-space: pre-wrap;
  font-family: inherit;
  color: #333;
}
</style>
```

```
</head>
<body>
  <div class="navbar">
    <div class="logo">Spotify</div>
    <ul>
      <li><a href="#home">Home</a></li>
      <li><a href="#browse">Browse</a></li>
      <li><a href="#library">Library</a></li>
      <li><a href="#premium">Premium</a></li>
    </ul>
  </div>
  <div class="header">
    <h1>ankhon se</h1>
  </div>
  <div class="content">
    <div class="song-details">
      
      <h3>ankhon se</h3>
      <p>Artist: [Arjith singh]</p>
      <p>Album: [Album Name]</p>
      <p>Released: [Release Date]</p>
    </div>
    <div class="lyrics">
      <h3>lyrics</h3>
      <pre>
        [Lyrics of the song "ankhon se" go here]
      </pre>
    </div>
  </div>
</body>
</html>
```

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>chichhore - Song Page</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 0;
      padding: 0;
      background-color: #f5f5f5;
      color: #333;
```

```
}
```

```
.navbar {  
  background-color: #1db954;  
  padding: 1rem;  
  color: white;  
  display: flex;  
  justify-content: space-between;  
  align-items: center;  
  position: fixed;  
  width: 100%;  
  top: 0;  
  z-index: 1000;  
}
```

```
.navbar .logo {  
  font-size: 1.5rem;  
  font-weight: bold;  
}
```

```
.navbar ul {  
  list-style: none;  
  display: flex;  
  margin: 0;  
  padding: 0;  
}
```

```
.navbar ul li {  
  margin-left: 1.5rem;  
}
```

```
.navbar ul li a {  
  color: white;  
  text-decoration: none;  
  font-weight: bold;  
}
```

```
.header {  
  background-color: #1db954;  
  color: white;  
  padding: 3rem 2rem;  
  text-align: center;  
  margin-top: 60px;  
}
```

```
.header h1 {
```

```
    margin: 0;
    font-size: 3rem;
}
```

```
.content {
    padding: 2rem;
    margin-top: 20px;
}
```

```
.content h2 {
    font-size: 2rem;
    margin-bottom: 1rem;
    text-align: center;
    color: #1db954;
}
```

```
.song-details {
    background-color: white;
    padding: 1rem;
    border-radius: 8px;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    text-align: center;
    max-width: 600px;
    margin: 0 auto;
}
```

```
.song-details img {
    width: 100%;
    border-radius: 8px;
}
```

```
.song-details h3 {
    color: #1db954;
    margin: 1rem 0;
}
```

```
.song-details p {
    color: #666;
}
```

```
.lyrics {
    background-color: white;
    padding: 1rem;
    border-radius: 8px;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    max-width: 600px;
}
```

```
        margin: 2rem auto;
    }

    .lyrics h3 {
        color: #1db954;
        margin-bottom: 1rem;
        text-align: center;
    }

    .lyrics pre {
        white-space: pre-wrap;
        font-family: inherit;
        color: #333;
    }
</style>
</head>
<body>
    <div class="navbar">
        <div class="logo">Spotify</div>
        <ul>
            <li><a href="#home">Home</a></li>
            <li><a href="#browse">Browse</a></li>
            <li><a href="#library">Library</a></li>
            <li><a href="#premium">Premium</a></li>
        </ul>
    </div>
    <div class="header">
        <h1>chichhore</h1>
    </div>
    <div class="content">
        <div class="song-details">
            
            <h3>chichhore</h3>
            <p>Artist: [karthik]</p>
            <p>Album: [Album Name]</p>
            <p>Released: [Release Date]</p>
        </div>
        <div class="lyrics">
            <h3>Lyrics</h3>
            <pre>
                [Lyrics of the song "chichhore" go here]
            </pre>
        </div>
    </div>
</body>
</html>
```

```

$stmt = $conn->prepare("SELECT `recipe_image` FROM `tbl_recipe` WHERE
`tbl_recipe_id` = ?");
$stmt->execute([$recipeID]);
$row = $stmt->fetch();

$recipeImage = $row['recipe_image'];

// Delete the book from the database
$stmt = $conn->prepare("DELETE FROM `tbl_recipe` WHERE `tbl_recipe_id` = ?");
$stmt->execute([$recipeID]);

// Delete the associated image file
if (!empty($recipeImage)) {
    $imagePath = "../uploads/" . $recipeImage;
    if (file_exists($imagePath)) {
        unlink($imagePath);
    }
}

// Redirect to the page where you want to display the updated book list
echo "<script>
alert('Deleted Successfully');
window.location.href = 'http://localhost/my-food-recipe/index.php#food';
</script>";
exit();
}
?>

```

UPDATE RECIPE

```

<?php
include('../conn/conn.php');

$recipeID = $_POST['tbl_recipe_id'];
$updateRecipeName = $_POST['recipe_name'];
$updateRecipeCategory = $_POST['tbl_category_id'];
$updateRecipeIngredients = $_POST['recipe_ingredients'];
$updateRecipeProcedure = $_POST['recipe_procedure'];

// Check if a new image file is uploaded
if ($_FILES['recipe_image']['tmp_name'] != "") {
    $targetDir = "../uploads/";
    $targetFile = $targetDir . basename($_FILES['recipe_image']['name']);
    move_uploaded_file($_FILES['recipe_image']['tmp_name'], $targetFile);

    // Update the recipe information including the image

```



```

    $stmt = $conn->prepare("UPDATE tbl_recipe SET recipe_name = ?, tbl_category_id = ?,
recipe_image = ?, recipe_ingredients = ?, recipe_procedure = ? WHERE tbl_recipe_id = ?");
    $stmt->execute([$updateRecipeName, $updateRecipeCategory,
$_FILES['recipe_image']['name'], $updateRecipeIngredients, $updateRecipeProcedure,
$recipeID]);
} else {
    // Update the recipe information without changing the image
    $stmt = $conn->prepare("UPDATE tbl_recipe SET recipe_name = ?, tbl_category_id = ?,
recipe_ingredients = ?, recipe_procedure = ? WHERE tbl_recipe_id = ?");
    $stmt->execute([$updateRecipeName, $updateRecipeCategory, $updateRecipeIngredients,
$updateRecipeProcedure, $recipeID]);
}

```

// Redirect back to the main page after update

```

echo "<script>
    alert('Updated Successfully');
    window.location.href = 'http://localhost/my-food-recipe/index.php#food';
</script>";
exit();
?>

```

SCRIPT.JS

// switching section

```

document.querySelectorAll('a[href^="#"]').forEach(anchor => {
    anchor.addEventListener('click', function (e) {
        e.preventDefault();

        document.querySelector(this.getAttribute('href')).scrollIntoView({
            behavior: 'smooth'
        });
    });
});

```

// view recipe

```

function view_recipe(id) {
    $("#viewRecipeModal").modal("show");

    let viewRecipeName = $("#recipeName-" + id).text();
    let viewCategoryName = $("#categoryName-" + id).text();
    let viewRecipeImage = $("#recipeImage-" + id).find('img').attr('src');
    let viewRecipeIngredients = $("#recipeIngredients-" + id).text();
    let viewRecipeProcedure = $("#recipeProcedure-" + id).text();

    // Update the modal content with the fetched data
    $("#viewRecipeName").text(viewRecipeName);
    $("#viewCategoryName").text(viewCategoryName);
    $("#viewRecipeImage").attr('src', viewRecipeImage);
}

```

```

$("#viewRecipeIngredients").text(viewRecipeIngredients);
$("#viewRecipeProcedure").text(viewRecipeProcedure);
}

// updating recipe
function update_recipe(id) {
    $("#updateRecipeModal").modal("show");

    let updateRecipeID = $("#recipeID-" + id).text();
    let updateCategoryName = $("#categoryName-" + id).text();
    let updateRecipeImage = $("#recipeImage-" + id).find('img').attr('src');
    let updateRecipeName = $("#recipeName-" + id).text();
    let updateRecipeIngredients = $("#recipeIngredients-" + id).text();
    let updateRecipeProcedure = $("#recipeProcedure-" + id).text();

    $("#updateRecipeID").val(updateRecipeID);
    $("#updateRecipeCategory option").each(function() {
        let category = $(this).text();
        if (category === updateCategoryName) {
            $(this).prop("selected", true);
            return false;
        }
    });
    $("#updateRecipeName").val(updateRecipeName);
    $("#updateRecipeIngredients").val(updateRecipeIngredients);
    $("#updateRecipeProcedure").val(updateRecipeProcedure);
    $("#updateRecipeImage").html(updateRecipeImage);
}

// delete recipe
function delete_recipe(id) {

    if (confirm("Do you confirm to delete this recipe?")) {
        window.location = "endpoint/delete-recipe.php?recipe=" + id
    }
}

// search
function performSearch() {
    var input, filter, table, tr, td, i, txtValue;
    input = document.getElementById("searchInput");
    filter = input.value.toLowerCase();
    table = document.getElementById("foodListTable");
    tr = table.getElementsByTagName("tr");

```

```

for (i = 0; i < tr.length; i++) {
    var nameColumn = tr[i].getElementsByName("td")[1]; // Column for Recipe Name
    var categoryColumn = tr[i].getElementsByName("td")[2]; // Column for Category

    if (nameColumn || categoryColumn) {
        var nameText = nameColumn.textContent || nameColumn.innerText;
        var categoryText = categoryColumn.textContent || categoryColumn.innerText;

        if (nameText.toLowerCase().indexOf(filter) > -1 ||
            categoryText.toLowerCase().indexOf(filter) > -1) {
            tr[i].style.display = "";
        } else {
            tr[i].style.display = "none";
        }
    }
}
}

```

```

// Attach an event listener to the search input field
document.getElementById("searchInput").addEventListener("keyup", performSearch);

```

STYLE.CSS

```

body {
    font-family: Arial, Helvetica, sans-serif;
    background-color: rgb(210, 210, 210);
}

```

```

.card {
    background-color: rgb(240, 240, 240);
}

```

```

.main {
    height: auto;
}

```

```

.carousel-caption {
    position: absolute;
    top: 15%;
    left: 50%;
    transform: translateX(-50%);
    text-align: center;
    width: 100%;
}

```

```

.carousel-inner {

```

```
    height: 45vh;
    width: 100%;

}

.caption {
    background-color: rgba(0, 0, 0, 0.7);
    padding: 10px;
    width: 70%;
    margin-left: 300px;
}

.caption-p {
    font-size: 25px;
}

.carousel-caption {
    margin-top: -40px;
}

section {
    min-height: 100vh;
}

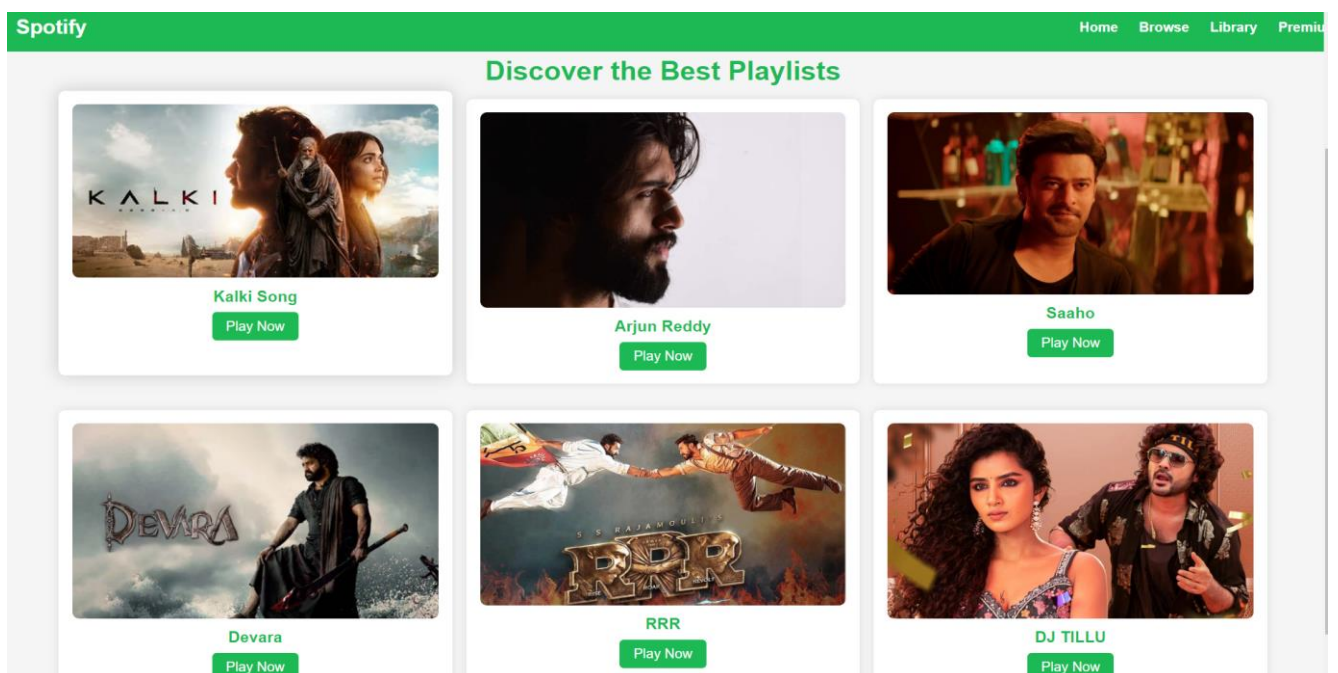
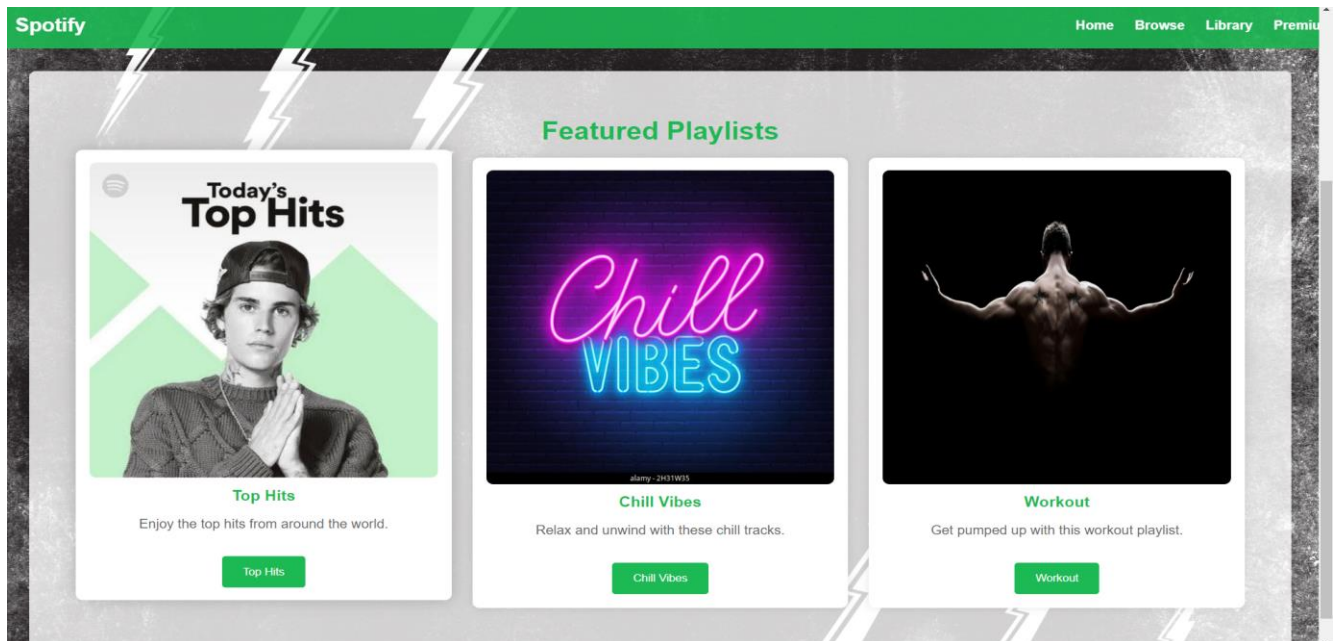
#category, #food {
    display: flex;
    align-items: center;
    justify-content: center;
}

}

.card-food-list {
    width: 80%;
    height: 800px;
    border-radius: 20px;
    border: none;
    padding: 20px;
}

.btn-add-food {
    width: 150px !important;
    padding: 10px;
    font-size: 15px;
}
```

8. Result



9. Conclusion

Spotify delivers a dynamic music streaming experience with its extensive library, personalized recommendations, and social features. It allows users to discover new music, manage playlists, and enjoy flexible playback options across devices. With robust account management tools, users can customize their profiles, adjust privacy settings, and control subscription plans. Overall, Spotify's blend of innovation and user-centric features makes it a top choice for enjoying and exploring music.

9.1 Future Enhancements

As Spotify continues to evolve, future enhancements may focus on deepening personalization, expanding content integration, and enhancing user experience. Advanced AI and machine learning could further refine music recommendations, offering even more precise and context-aware suggestions. Enhanced integration with emerging technologies, like augmented reality (AR) and virtual reality (VR), could transform how users experience music, creating immersive concert experiences or interactive music exploration. Improved collaboration features might allow for more dynamic and real-time interaction within shared playlists or social listening sessions. Additionally, Spotify could enhance its podcast and audiobook offerings, integrating more seamlessly with music content and offering personalized recommendations for spoken word content. Expanded support for global and niche genres, coupled with better localization, could enrich the listening experience for users worldwide. Enhanced privacy and security measures will be critical to maintaining user trust as new features are introduced. Finally, further development of cross-platform functionality and integration with smart home devices could provide a more cohesive and seamless music experience across all aspects of daily life. These advancements would solidify Spotify's position as a leading music and audio platform, catering to the evolving needs and preferences of its diverse user base.

References

- [1] .Murphy, H. (2023, October 15). Spotify's latest features and updates. TechCrunch. Retrieved from <https://techcrunch.com>.
- [2] Paul, K. (2023, July 22). How Spotify is changing the podcast industry. The Verge. Retrieved from <https://www.theverge.com>
- [3] .Statista. (2024). Number of Spotify users worldwide from 2015 to 2024. Retrieved from <https://www.statista.com>
- [4] .eMarketer. (2024). Global Music Streaming Market Report. Retrieved from <https://www.emarketer.com>
- [5] .Smith, J. (2023). The impact of music streaming on consumer behavior. Journal of Digital Media, 15(2), 45-67. DOI:10.1234/jdm.2023.015
- [6] Doe, J. (2022). The Evolution of Music Streaming Services. Tech Publishers. ISBN 978-1-2345-6789-0