**ACKNOWLEDGEMENT**

With great pleasure I take this opportunity to express my heartfelt gratitude to all people who helped in making this seminar work a grand success.

First of all, I thank God Almighty for giving strength, courage and blessings to complete this work. I also thankful to my friends and relatives who helped me to complete this seminar work.

I express my sincere thanks to **Dr. Shaileshkumar M. Gheewala** (Lecturer EC in the Computer Engineering Department), for his guidance and suggestions during this project work. We would not forget to remember **Mr. M. D. Patel**, Head of Department, Computer Engineering, Dr. S. & S. S. Ghandhy College of Engineering & Technology, Surat for his encouragement and more over for his timely support and guidance till the completion of the internship work.

**Ariwa Patel**

**(226120307002)**

**DECLARATION OF ORIGINALITY**

I confirm that I am the only author of this internship and that neither any part nor the entire work has been submitted for a degree to any other University or Institution.

To the best of my knowledge, my work does not infringe upon anyone's copyright, nor violate any proprietary rights. Any ideas, techniques, quotations, or any other material from the work of other people included in our report or otherwise, are fully acknowledged by the standard referencing practices.

I affirm that this is an accurate copy of my report, including all final revisions, as approved by my supervisor.

**Date:**

**Place:** Surat



**Ariwa Patel**

**(226120307002)**

**Table of Contents**

|  |  |
| --- | --- |
| **Certificate** | 1 |
| **Acknowledgement** | 3 |
| **Declaration of Originality** | 4 |
| **Chapter-1 INTRODUCTION** | 6 |
| **Chapter-2 PROBLEM DEFINTION** | 7 |
| **Chapter-3 OVERVIEW** | 10 |
| **Homepage Code** | 11 |
| **JS Code Snippets** | 17 |
| **Results** | 19 |

**CHAPTER 1 INTRODUCTION**

**This project is a clone website of the popular music application of SPOTIFY. This project gives us a demo about the visuals, graphics and the system used by spotify. The main purpose behind developing this mini project was to understand the working of the existential systems and the loopholes.**

**Creating a Spotify clone involves developing a music streaming application that replicates some or all of the features of the popular music streaming service, Spotify. This can be a complex project, but it offers a great opportunity to learn various aspects of software development, including frontend and backend technologies, database management, user authentication, and integrating third-party APIs. Below is a general introduction to the key components and features you might consider when creating a Spotify clone**

Key Components:

**1. User Interface (UI):**

Homepage: Display featured playlists, new releases, and personalized recommendations.

**2. Search Functionality:**

Allow users to search for artists, albums, and tracks.

**3. User Profile:**

Provide a user profile page with playlists, saved tracks, and user information.

**4. Music Catalog:**

Database: Store information about artists, albums, and tracks in a structured database.

**5. Metadata:**

Include details like song duration, genre, release date, and artist information.

**6. Audio Streaming:**

Audio Player: Develop a player for streaming music with play, pause, skip, and volume controls.

**7. User Authentication:**

Sign Up/Log In: Allow users to create accounts and log in securely.

**8. Authorization:**

Implement user authentication to control access to certain features.

**9. Playlists and Library:**

Create Playlists: Enable users to create, edit, and delete playlists.

**CHAPTER 2 Problem Definition**

**The problem definition for creating a Spotify clone involves identifying the key challenges and requirements associated with developing a music streaming application that replicates the features and functionalities of Spotify. Here's a detailed problem definition:**

Problem Statement:

Overview:

**Develop a music streaming application, referred to as the "Spotify clone," that emulates the core features of Spotify. This includes creating a user-friendly interface, implementing audio streaming, managing a comprehensive music catalog, enabling user authentication, and incorporating social and recommendation features.**

Key Objectives:

**1. User Interface (UI):**

* **Challenge:** Design and implement an intuitive and responsive user interface that closely resembles the aesthetic and functionality of Spotify.
* **Requirements:** Create pages for homepage, search, user profile, playlists, and recommendations.

**Music Catalog Management:**

* **Challenge:** Develop a database system to store and manage extensive information about artists, albums, and tracks.
* **Requirements:** Ensure efficient retrieval and update of music metadata.

**Audio Streaming:**

* **Challenge:** Implement a reliable and smooth audio streaming system for playing music**.**
* **Requirements:** Include features like play, pause, skip, volume control, and seamless buffering.

**User Authentication and Authorization:**

* **Challenge:** Establish a secure user authentication and authorization system.
* **Requirements:** Enable users to sign up, log in securely, and control access to premium features based on subscription status.

**Playlists and Library:**

* **Challenge:** Create a system for users to create, edit, and delete playlists, as well as save favorite tracks.
* **Requirements:** Allow users to organize their music library effectively.

**Recommendation Engine:**

* **Challenge:** Implement algorithms to provide personalized music recommendations based on user preferences and listening history.
* **Requirements:** Enhance user engagement by suggesting relevant content.

**Social Features:**

* **Challenge:** Develop social functionalities such as following friends, sharing playlists, and collaborating on playlists.
* **Requirements:** Foster a sense of community among users through social interactions.

**Payment Integration:**

* **Challenge:** Integrate a secure payment system for subscription-based premium features.
* **Requirements:** Implement subscription models and ensure smooth payment processing.

**Notifications:**

* **Challenge:** Provide timely and relevant notifications for events such as new releases, friend activity, and playlist updates.
* **Requirements:** Enhance user engagement by keeping users informed and involved.

**Offline Mode:**

* **Challenge:** Implement a feature that allows users to download songs for offline listening.
* **Requirements:** Provide a seamless offline experience without compromising on audio quality.

**Technologies and Tools:**

* **Frontend:** Choose a suitable frontend framework (React, Angular, Vue) for building a responsive and visually appealing user interface.
* **Backend:** Select a backend technology (Node.js, Django, Flask) for server-side development, ensuring efficient data management and processing.
* **Database:** Use a reliable database system (MySQL, PostgreSQL, MongoDB) for storing and retrieving music metadata and user information.
* **Audio Streaming:** Implement WebSockets for real-time communication and a dedicated media server for efficient audio streaming.
* **Authentication:** Employ OAuth 2.0 for secure user authentication and authorization.
* **APIs:** Utilize the Spotify Web API for fetching music metadata and integrating Spotify-like features**.**
* **Cloud Services:** Explore options such as AWS, Google Cloud, or Azure for hosting and storage**.**
* **Version Control:** Use Git and GitHub for collaborative development and version control.
* **Containerization:** Consider Docker for packaging and deploying the application in a consistent environment.

**Legal Considerations:**

**Ensure compliance with copyright laws, licensing agreements, and intellectual property rights when replicating features and using music data.**

**By addressing these challenges and requirements, the development team can create a functional and feature-rich Spotify clone that provides users with a seamless and enjoyable music streaming experience.**

**CHAPTER 3 Overview**

**FEASIBILITY STUDY:**

**i.) Technical Feasibility:**

* The languages used here are HTML, CSS & JavaScript. -Demo dataset of songs is taken from Internet Sources.

**ii.) Economical Feasibility:**

* It is cost effective.
* Cost/Benefit Analysis

**TECHNICAL REQUIREMENT:**

**i.) Hardware:**

* RAM: Min 4GB
* Processor: Ryzen 4000

**ii.) Coding style:**

* HTML
* CSS
* JAVASCRIPT

**Gathering Data Requirement & Functionality Requirement:-**

**i.) Project manager: -**

* Customer
* User

**ii.) Input data of the system: -**

* Music

**iii.) Output information of the system:-**

* plays the songs

**iv.) Functional / Processing of the system:**

* Processing Requirement:
* Requires the internet connection in the system. Needs the browser to start the music player.

HOMEPAGE PROGRAM CODE:

<!DOCTYPE html>

<head>

    <title>Spotify</title>

    <link rel="stylesheet" href="index.css">

    <link rel="ion" type="image/x-ion" href="C:\Users\Ariwa\OneDrive\Desktop\project'\spotify clone\favion.png">

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1">

    <link href='https://fonts.googleapis.com/css?family=Roboto:400,100,100italic,300,300italic,400italic,500,500italic,700,700italic,900italic,900' rel='stylesheet' type='text/css'>

    <script src="index.js"></script>

    <link rel="stylesheet" href="https://code.ionicframework.com/ionicons/2.0.1/css/ionicons.min.css">

</head>

<body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/noUiSlider/8.3.0/nouislider.min.js"></script>

<section class="header">

    <div class="window\_\_actions">

      <i class="ion-record red"></i>

      <i class="ion-record yellow"></i>

      <i class="ion-record green"></i>

    </div>

    <div class="page-flows">

        <span class="flow">

            <i class="ion-chevron-left"></i>

        </span>

        <span class="flow">

            <i class="ion-chevron-right disabled"></i>

        </span>

    </div>

    <div class="search">

        <input type="text" placeholder="Search" />

    </div>

    <div class="user">

        <div class="user\_\_notifications">

            <i class="ion-android-notifications"></i>

        </div>

        <div class="user\_\_inbox">

            <i class="ion-archive"></i>

        </div>

        <div class="user\_\_info">

            <span class="user\_\_info\_\_img">

                <img src="pfp.jpg" alt="Profile Picture" class="img-responsive"  width="25px" height="25px"/>

            </span>

            <span class="user\_\_info\_\_name">

                <span class="first">Ariwa</span>

                <span class="last">Patel</span>

            </span>

        </div>

        <div class="user\_\_actions">

           <label for="Activity" style="background-color: #181818;">

           <select name="Activity" id="Activity"  >

            <option value="1">private session</option>

            <option value="1">account</option>

            <option value="1">setting</option>

            <option value="1">logout</option>

           </select>

        </label>

        </div>

        </div>

    </div>

</section>

We have then created another html file for login page

PROGRAM CODE:

<!DOCTYPE html>

<head>

  <title>login</title>

  <link rel="ion" type="image/x-ion" href="C:\Users\Ariwa\OneDrive\Desktop\project'\spotify clone\favicon.png">

    <meta charset="UTF-8">

    <meta name="description" content="Free Web tutorials">

    <meta name="keywords" content="HTML, CSS, JavaScript">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet" href="login.css">

    <script src="index.js"></script>

</head>

<body background="C:\Users\Ariwa\OneDrive\Desktop\project'\spotify clone\bg.png">

  <div id="id01" class="modal"></div>

    <span onclick="document.getElementById('id01').style.display='none'"

  class="close" title="Close Modal">&times;</span>

    <!-- Modal Content -->

    <form class="modal-content animate" action="/action\_page.php">

      <div class="imgcontainer">

        <img src="login.png" alt="Avatar" class="avatar">

      </div>

      <div class="container"><center>

        <label for="uname" id="uname" ></label><br>

        <input type="text" placeholder="Username" name="uname" required style="background-color: #3e4240; border-radius: 25px;">

      <br>

      <br>

        <label for="psw" style="border-radius: 25px;"></label><br>

        <input type="password" placeholder="Password" name="psw" required style="background-color: #3e4240; border-radius: 25px;">

        <br>

        <br>

        <br>

        <button type="submit" style="background-color: #1ed760; border-radius: 25px;"><a href="index.html" style="color: black; text-decoration: none; ">LOG IN</a></button>

        <label style="color:silver; font-family:'Courier New', Courier, monospace; font-display: inherit;">

          <input type="checkbox" checked="checked" name="remember">Remember me

        </label>

        <br><br>

      </div>

      <div class="container" style="background-color:#181818"><center>

        <span class="psw" ><a href="https://community.spotify.com/t5/FAQs/How-do-I-reset-my-password/ta-p/1300394?gad=1&gclid=Cj0KCQjwusunBhCYARIsAFBsUP8ttJfYHRwgLBAZ6lgA\_a-6OHJMc7hqtGer6RrCuvUb1gxxaJ4BPiUaAmpVEALw\_wcB&gclsrc=aw.ds" style="color:silver; text-decoration: none;">Forgot password?</a></span>

      </div>

    </form>

  </div>

</center>

<script>

  var modal = document.getElementById('id01');

window.onclick = function(event) {

  if (event.target == modal) {

    modal.style.display = "none";

  }

}

</script>

</body>

  </html>

MUSIC PLAYER PROGRAM CODE:

<!DOCTYPE html>

<head>

    <title>EFW spot</title>

    <link href="main.css" rel="stylesheet">

    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">

    <link rel="icon" type="image/x-icon" href="favicon.png">

</head>

<div>

   >

</div>

<body class="screen">

   <input type="checkbox" value="None" id="magicButton" name="check" />

    <label class="main" for="magicButton"></label>

    <img class="coverImage" id="cover" src="https://c.saavncdn.com/142/Good-Life-English-2017-500x500.jpg"> bla</img>

    <div class="bodyPlayer"></div>

    <table class="list">

        <tr class="song" id="song1">

            <td class="nr"><h5>1<h5></td>

            <td class="title"><button id="b1" onclick="btn1()">Good Life</button></td>

            <td class="length"><h5>3:50<h5></td>

                <td><input type="checkbox" id="heart"/>

                    <label class="zmr" for="heart"></label></td>

        </tr>

                    <tr class="song" id="song2">

                          <td class="nr"><h5>2<h5></td>

                          <td class="title"><button id="b1" onclick="btn2()">Raataan Lambiyan</button></td>

                          <td class="length"><h5>3:50<h5></td>

                            <td><input type="checkbox" id="heart1" checked /><label class="zmr" for="heart1"></label></td>

                         </tr>

                      <tr class="song" id="song3">

                          <td class="nr"><h5>2<h5></td>

                          <td class="title"><button id="b1" onclick="btn2()">Kesariya</button></td>

                          <td class="length"><h5>4:28<h5></td>

                          <td><input type="checkbox" id="heart2" checked /><label class="zmr" for="heart2"></label></td>

                      </tr>

                      <tr class="song" id="song4">

                          <td class="nr"><h5>3<h5></td>

                          <td class="title"><button id="b1" onclick="btn4()">Mehbooba</button></td>

                          <td class="length"><h5>4:08<h5></td>

                          <td><input type="checkbox" id="heart3"/><label class="zmr" for="heart3"></label></td>

                      </tr>

                      <tr class="song" id="song5">

                          <td class="nr"><h5>5<h5></td>

                          <td class="title"><button id="b1" onclick="btn5()">Apna Bana Le</button></td>

                          <td class="length"><h5>4:21<h5></td>

                          <td><input type="checkbox" id="heart4" checked /><label class="zmr" for="heart4"></label></td>

                      </tr>

                      <tr class="song" id="song6">

                          <td class="nr"><h5>6<h5></td>

                          <td class="title"><button id="b1" onclick="btn6()">Maan Meri Jaan</button></td>

                          <td class="length"><h5>3:14<h5></td>

                          <td><input type="checkbox" id="heart5"/><label class="zmr" for="heart5"></label></td>

                      </tr>

                      <tr class="song" id="song7">

                          <td class="nr"><h5>8<h5></td>

                          <td class="title"><button id="b1" onclick="btn7()">Deva Deva</button></td>

                          <td class="length"><h5>4:39<h5></td>

                          <td><input type="checkbox" id="heart6"/><label class="zmr" for="heart6"></label></td>

                      </tr>

                      <tr class="song" id="song8">

                          <td class="nr"><h5>8<h5></td>

                          <td class="title"><button id="b1" onclick="btn8()">Tum Mere</button></td>

                          <td class="length"><h5>3:48<h5></td>

                          <td><input type="checkbox" id="heart7"/><label class="zmr" for="heart7"></label></td>

                      </tr>

                      <tr class="song" id="song9">

                          <td class="nr"><h5>9<h5></td>

                          <td class="title"><button id="b1" onclick="btn9()">O Bedardeya</button></td>

                          <td class="length"><h5>2:51<h5></td>

                          <td><input type="checkbox" id="heart8"/><label class="zmr" for="heart8"></label></td>

                      </tr>

                  </table>

                  <div class="shadow"></div>

                  <div class="bar"></div>

                  <div class="info" id="info">

                      <h4 style="justify-self: center;">Good Life</h4>

                      <h3 style="justify-self: center;margin-left: 20px;">G-easy Fast and Furious 8</h3>

                  </div>

                  <audio preload="auto" id="audio" controls>

                    <source id="source" src="Kesariya.mp3">

                      </audio>

                  <table class="player">

                      <td><button style="height:100%; width:80%;" id="backward" onclick="backward()"><img src="back.png" height="100%" width="100%"></button>

                      </td>

                      <td><button style="height:100%; width:80%;" id="play" title="Play" onclick="togglePlayPause()">

                        <img id="image" src="play (2).png" height="100%" width="100%"></button>

                    </td>

                      <td><button style="height:100%; width:80%;" id="forward" onclick="forward()"><img src="next.png" height="100%" width="100%"></button>

                    </td>

                  </table>

                  <table class="footer">

                      <td><input type="checkbox" id="love" checked /><label class="love" for="love"></label></td>

                      <td><input type="checkbox" id="shuffle"/><label class="shuffle" for="shuffle"></label></td>

                      <td><input type="checkbox" id="repeat" checked /><label class="repeat" for="repeat"></label></td>

                      <td><input type="checkbox" id="options"/><label class="options" for="options"></label></td>

                  </table>

                  <script src="main.js"></script>

                </body>

                </html>

JS CODE SNIPPETS:

var audio = new Audio();

var playpause = document.getElementById("play");

var img = document.getElementById("image");

var forbtn = document.getElementById("forward");

var backbtn = document.getElementById("backward");

var cover = document.getElementById("cover");

var info = document.getElementById("info");

a function is created for audio pause play and default song is good life. Js has inbuilt audio functions for example, audio.pause(), audio.play(), audio.end().

audio = new Audio("good life.mp3");

let count=1;

function togglePlayPause() {

   if (audio.paused || audio.ended) {

      img.src = "pause.png";

      audio.play();

   } else {

      img.src = "play (2).png";

      audio.pause();

   }

}

For forwarding the music switch case has been used. Count variable increments by one whenever forward button is used and once its greater than 8 that is number of songs, count variable is again assigned 1 creating loop from end to first.

function forward(){

      count++;

      if(count>8){

         count=1;

      }

      switch(count){

         case 2:

            audio.pause();

            audio = new Audio("Raatan Lambiyan.mp3");

            info.children[0].textContent="Raatan Lambiyan";

            info.children[1].textContent="Jubin Nautiyal - Shershah";

            cover.src="https://c.saavncdn.com/222/Raataan-Lambiyan-From-Shershaah--Hindi-2021-20210729181107-500x500.jpg";

            audio.play();

         break;

backward method is used to play previous song in our list.

function backward(){

   count--;

   if(count<1){

      count=8;

   }

      switch(count){

         case 1:

            audio.pause();

            audio = new Audio("good life.mp3");

            info.children[0].textContent="Good Life";

            info.children[1].textContent="G-easy - Fast Furious 8";

            cover.src="https://c.saavncdn.com/142/Good-Life-English-2017-500x500.jpg";

            audio.play();

EXTERNAL CSS LINKS:

<https://code.ionicframework.com/ionicons/2.0.1/css/ionicons.min.css>

<https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css>

Results:-





