**Ex.No.2 Music Streaming Web Application Date: 03-04-2025**

**Aim:**To develop a music streaming web application to provide users with a seamless and interactive music listening experience.

**Procedure:**

1 **User Authentication (Login Page)**

* Users enter a **username & password** to log in.
* Upon successful login, they are **redirected to the music player**.

2 **Music Player UI**

* Displays a **grid of songs** with title & artist name.
* Each song has **Play, Stop, and Like buttons**.

3 **Playback Functionality**

* Clicking **Play** starts the song.
* Clicking **Stop** stops the song.

4 **Like & Favorite Songs**

* Users can **"like" songs** by clicking the heart button.
* Liked songs change color to **red**.

5 **Attractive UI Design**

* **Gradient backgrounds** and **glassmorphism effects**.
* **Responsive design** for desktop & mobile.

6 **Modern Controls**

* Uses **React Icons** for buttons ( Play, Stop, Like).

7 **Navigation with React Router**

* Users can move between **Login** and **Music Player** pages.

.

**Program:**

***App.jsx:***

import React from "react";

import { BrowserRouter as Router, Routes, Route } from "react-router-dom";

import Login from "./Login";

import MusicPlayer from "./MusicPlayer";

const App = () => {

  return (

    <Router>

      <Routes>

        <Route path="/" element={<Login />} />

        <Route path="/music-player" element={<MusicPlayer />} />

      </Routes>

    </Router>

  );

};

export default App;

***Login.jsx:***

import React from "react";

import { useNavigate } from "react-router-dom";

import './Auth.css'

const Login = () => {

  const navigate = useNavigate();

  const handleLogin = (e) => {

    e.preventDefault(); // Prevent form refresh

    navigate("/music-player"); // Navigate to Music Player Page

  };

  return (

    <div className="login-container">

      <div className="login-box">

        <h2>Login</h2>

        <form onSubmit={handleLogin}>

          <input type="text" placeholder="Username" required />

          <input type="password" placeholder="Password" required />

          <button type="submit">Login</button>

        </form>

        <p>Forgot password?</p>

      </div>

    </div>

  );

};

export default Login;

***main.jsx:***

import { StrictMode } from 'react'

import { createRoot } from 'react-dom/client'

import App from './App.jsx'

createRoot(document.getElementById('root')).render(

  <StrictMode>

    <App />

  </StrictMode>,

)

***Music-Player.jsx:***

import { useState, useRef } from "react";

import { FaPlay, FaStop, FaHeart } from "react-icons/fa";

import './styles.css'

const songs = [

  { title: "Chances", artist: "Silent Partner", url: "https://www.soundhelix.com/examples/mp3/SoundHelix-Song-1.mp3" },

  { title: "Space Walk", artist: "Silent Partner", url: "https://www.soundhelix.com/examples/mp3/SoundHelix-Song-2.mp3" },

  { title: "Don't Hate Me", artist: "Jingle Punks", url: "https://www.soundhelix.com/examples/mp3/SoundHelix-Song-3.mp3" },

  { title: "Dreams", artist: "Bensound", url: "https://www.soundhelix.com/examples/mp3/SoundHelix-Song-4.mp3" },

  { title: "Eclipse", artist: "Bensound", url: "https://www.soundhelix.com/examples/mp3/SoundHelix-Song-5.mp3" },

  { title: "Epic Journey", artist: "Bensound", url: "https://www.soundhelix.com/examples/mp3/SoundHelix-Song-6.mp3" },

];

const MusicPlayer = () => {

  const [currentSong, setCurrentSong] = useState(null);

  const [likedSongs, setLikedSongs] = useState([]);

  const audioRef = useRef(null);

  const playSong = (song) => {

    if (audioRef.current) {

      audioRef.current.pause();

      audioRef.current.src = song.url;

      audioRef.current.play();

      setCurrentSong(song);

    }

  };

  const stopSong = () => {

    if (audioRef.current) {

      audioRef.current.pause();

      audioRef.current.currentTime = 0;

      setCurrentSong(null);

    }

  };

  const toggleLike = (song) => {

    setLikedSongs((prev) =>

      prev.includes(song) ? prev.filter((s) => s !== song) : [...prev, song]

    );

  };

  return (

    <div className="container">

      <h1 className="title">VibeTune 🎵</h1>

      <div className="songGrid">

        {songs.map((song, index) => (

          <div key={index} className="songCard">

            <h2 className="songTitle">{song.title}</h2>

            <p className="songArtist">{song.artist}</p>

            <div className="controls">

              <button className="playButton" onClick={() => playSong(song)}><FaPlay /></button>

              <button className="stopButton" onClick={stopSong}><FaStop /></button>

              <button

                className="likeButton"

                style={{ color: likedSongs.includes(song) ? "#ff4d4d" : "#bbb" }}

                onClick={() => toggleLike(song)}

              >

                <FaHeart />

              </button>

            </div>

          </div>

        ))}

      </div>

      <audio ref={audioRef} />

      {currentSong && (

        <div className="player">

          <h2 className="playingText">Now Playing:</h2>

          <p className="playingSong">{currentSong.title} - {currentSong.artist}</p>

        </div>

      )}

    </div>

  );

};

export default MusicPlayer;

***App.css:***

***/\* Centering the login container \*/***

***body {***

***display: flex;***

***justify-content: center;***

***align-items: center;***

***height: 100vh;***

***background: linear-gradient(135deg, #1e3c72, #2a5298);***

***margin: 0;***

***font-family: 'Poppins', sans-serif;***

***}***

***/\* Login box styling \*/***

***.login-container {***

***display: flex;***

***justify-content: center;***

***align-items: center;***

***height: 100vh;***

***width: 100%;***

***}***

***.login-box {***

***background: rgba(255, 255, 255, 0.15);***

***backdrop-filter: blur(15px);***

***padding: 2.5rem;***

***border-radius: 20px;***

***box-shadow: 0px 5px 15px rgba(0, 0, 0, 0.3);***

***text-align: center;***

***width: 350px;***

***border: 1px solid rgba(255, 255, 255, 0.3);***

***}***

***/\* Heading \*/***

***h2 {***

***color: #ffffff;***

***margin-bottom: 20px;***

***font-size: 24px;***

***font-weight: 600;***

***}***

***/\* Input fields \*/***

***input {***

***width: 100%;***

***padding: 14px;***

***margin: 12px 0;***

***border: none;***

***border-radius: 8px;***

***outline: none;***

***font-size: 16px;***

***background: rgba(255, 255, 255, 0.2);***

***color: white;***

***text-align: center;***

***transition: 0.3s ease;***

***}***

***input::placeholder {***

***color: rgba(255, 255, 255, 0.8);***

***}***

***input:focus {***

***background: rgba(255, 255, 255, 0.3);***

***}***

***/\* Login button \*/***

***button {***

***width: 100%;***

***padding: 14px;***

***border: none;***

***border-radius: 8px;***

***background: linear-gradient(90deg, #ff8c00, #ff007a);***

***color: white;***

***font-size: 16px;***

***font-weight: bold;***

***cursor: pointer;***

***transition: 0.3s ease;***

***margin-top: 15px;***

***}***

***button:hover {***

***background: linear-gradient(90deg, #ff007a, #ff8c00);***

***transform: scale(1.05);***

***}***

***/\* Forgot password text \*/***

***p {***

***color: white;***

***font-size: 14px;***

***margin-top: 15px;***

***cursor: pointer;***

***transition: 0.3s;***

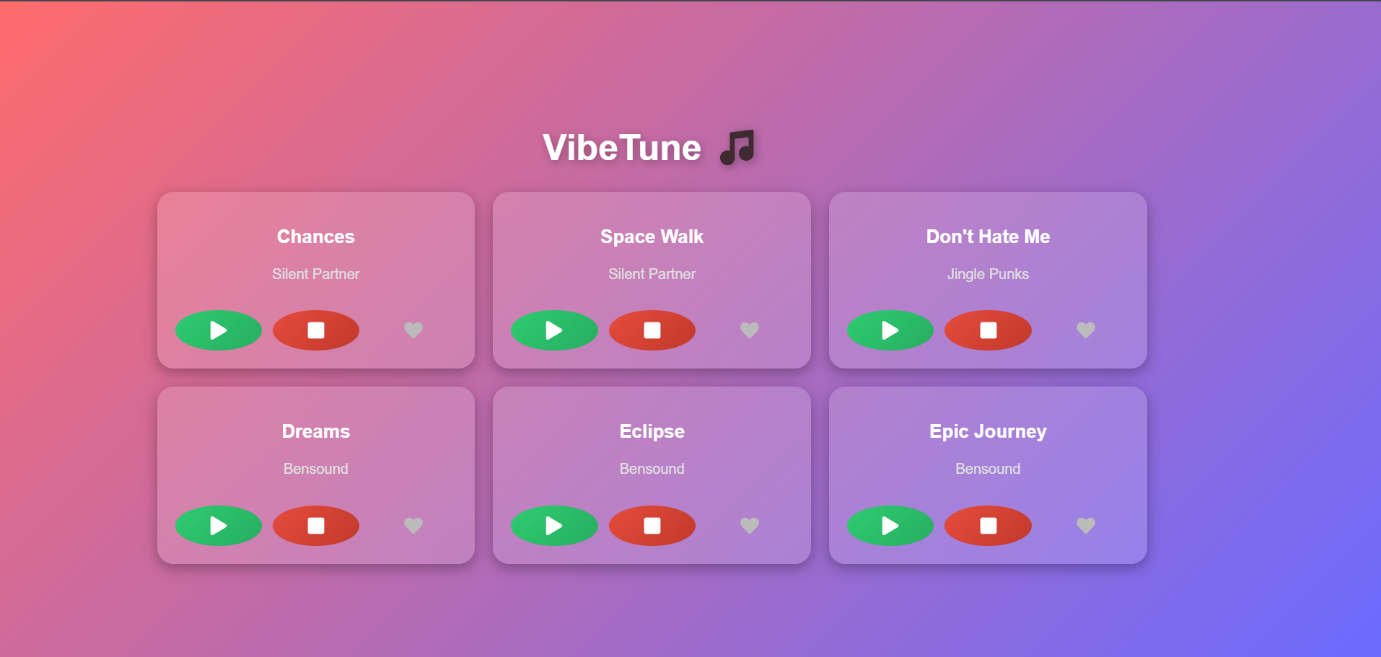
***}***

***p:hover {***

***text-decoration: underline;***

***color: #ffcc00;***

**Output:**

****

**Result:**

Thus, the music streaming web application to provide users with a seamless and interactive music listening experience was developed and the output was verified.