Khalsa College of Engineering & Technology

SOFTWARE ENGINEERING FILE

LABCODE:BTCS503-18

BACHELOR OF COMPUTER SCIENCE & ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Submitted By:- Nitish Kumar
Roll No:- 2304138

Submitted To:- Er.Parmjeet Kaur

Content

INTRODUCTION

In today’s digital world, multimedia applications have become an essential part of human life. Music, being one of the most universal forms of art and entertainment, plays a major role in web and mobile applications. From online streaming platforms to local players, music players are among the most widely used applications across all devices. As web technologies evolve, even simple browsers are capable of handling multimedia elements efficiently without the need for external plugins or software. The Nitonia Music Player is a web-based music player designed and developed using HTML, CSS, and JavaScript — the core building blocks of front-end web development. This project focuses on creating a functional, interactive, and user-friendly web application that allows users to play, pause, navigate, and control audio tracks seamlessly within a browser environment. The project demonstrates how front-end technologies can be used to implement multimedia functionality without relying on back-end frameworks or databases.

PROJECT OVERVIEW

The project titled “Creating a Nitonia Music Player using HTML, CSS, and JavaScript” is a web-based multimedia application that demonstrates how front-end web technologies can be integrated to create a fully functional and interactive music player. This project aims to provide an engaging platform where users can play, pause, and navigate through multiple audio tracks directly from a web browser, without the need for any server-side processing or third-party software.

The Nitonia Music Player showcases the power and flexibility of modern web standards and highlights how HTML5, Cascading Style Sheets (CSS3), and JavaScript (ES6) can work together to build rich, responsive, and dynamic web applications. It serves as an educational project for understanding user interface design, media control through scripting, and responsive web development practices.

Purpose of the Project

The main purpose of the Nitonia Music Player project is to design a simple, user-friendly, and visually appealing web music player that demonstrates the integration of HTML, CSS, and JavaScript to handle multimedia functionality.

PROBLEM STATEMENT

The Nitonia Music Player aims to solve this problem by using only front-end technologies — HTML, CSS, and JavaScript — to design and develop a fully functional web-based music player that runs locally on any device with a modern browser.

Problem Definition

To develop a web-based music player that:

- Operates entirely on the client-side using HTML, CSS, and JavaScript.
- Provides core playback functions such as play, pause, next, previous, seek, and volume control.
- Displays an interactive playlist that highlights the currently playing song.

Problem Statement Summary

The existing music player systems are either too complex or require backend infrastructure and installations. There is no simple, lightweight, front-end-only player suitable for demonstration, learning, and personal use. Hence, the Nitonia Music Player project aims to design and develop a fully functional, offline, browser-based audio player using only front-end technologies.

KEY FEATURES OF NITONIA MUSIC PLAYER:-

The Nitonia Music Player is a lightweight, browser-based web application designed to provide an interactive and enjoyable music-listening experience. It integrates the power of HTML5, CSS3, and JavaScript (ES6) to deliver a responsive, feature-rich, and user-friendly multimedia platform.

1. Play / Pause Controls

The most fundamental function of any music player is the ability to play and pause songs.
In the Nitonia Music Player, the play/pause button is controlled through JavaScript events bound to the HTML <audio> element.

2. Next / Previous Song Navigation

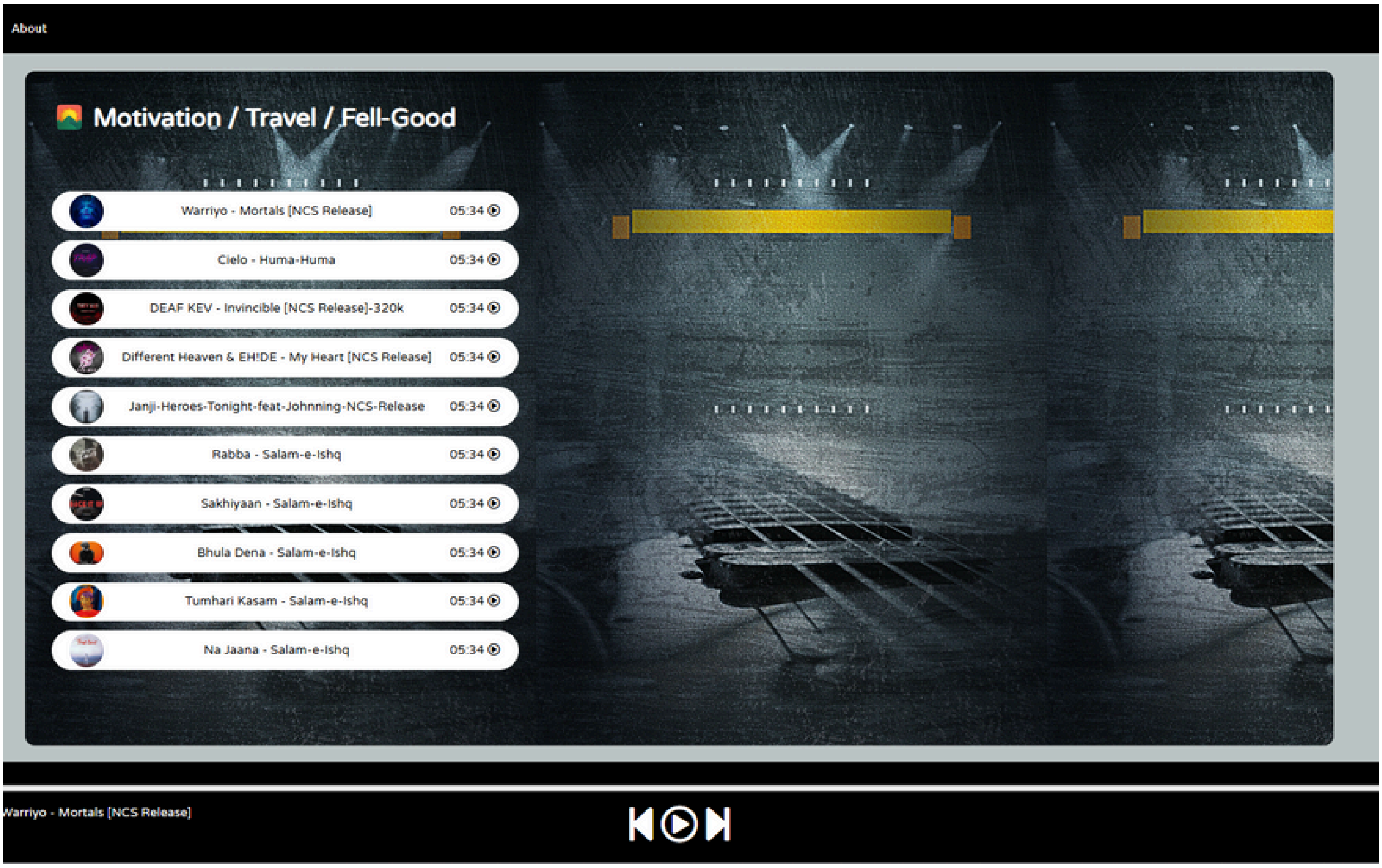
The player allows users to skip to the next or previous track in the playlist using navigation buttons.
This is implemented using an array of tracks in the JavaScript code. Each song is stored as an object with properties like title, source, and cover image. When the user clicks the “Next” or “Previous” button, the player updates the index of the current song and loads the corresponding file using the loadTrack() function.

3. Interactive Playlist Display

The Nitonia Music Player includes a visually appealing and interactive playlist that displays the list of available songs.

Each song entry contains:

- Song title
- Duration (optional)
- A play button beside it



Motivation / Travel / Fell-Good

Warriyo - Mortals [NCS Release]	05:34 ⏵
Cielo - Huma-Huma	05:34 ⏵
DEAF KEV - Invincible [NCS Release]-320k	05:34 ⏵
Different Heaven & EH!DE - My Heart [NCS Release]	05:34 ⏵
Janji-Heroes-Tonight-feat-Johnning-NCS-Release	05:34 ⏵
Rabba - Salam-e-Ishq	05:34 ⏵
Sakhiyaan - Salam-e-Ishq	05:34 ⏵
Bhula Dena - Salam-e-Ishq	05:34 ⏵
Tumhari Kasam - Salam-e-Ishq	05:34 ⏵
Na Jaana - Salam-e-Ishq	05:34 ⏵

Warriyo - Mortals [NCS Release]

Folder Structure of Nitonia Music Website:-

```
 Nitonia-Music/
  ├── index.html
    ├── css/  │
    └── styles.css
     ├──js/  │
     └── app.js
   ├── music/  │
├── track1.mp3  │
├── track2.mp3  │
└── ...  └── images/
   └── logo.png
```

Explanation:

The index.html file builds the structure of the webpage. The css/ folder controls the styling and layout through styles.css. The js/ folder manages functionality (Play/Pause, Next/Previous, Progress Bar, etc.) via app.js. The music/ folder stores all MP3 tracks used in the player. The images/ folder holds icons, artwork, and the logo for branding.

FOLDER STRUCTURE OF NITONIA MUSIC WEBSITE

A well-organized folder structure is the foundation of every successful web development project. It helps developers maintain clarity, improve scalability, and simplify debugging or future updates. In the Nitonia Music Player, the folder structure is designed following standard web development conventions to keep all resources — such as HTML files, stylesheets, JavaScript code, media files, and images — properly separated and easy to manage.

Purpose of Folder Organization

- To improve readability and maintainability of the project.
- To separate design, logic, and data elements.
- To ensure reusability of code components.
- To help in quick debugging and version control.

Project Folder Hierarchy

```
Nitonia-Music/
│
├─── index.html
│
├─── css/
│    └─── styles.css
│
├─── js/
│    └─── app.js
│
├─── music/
│    ├─── track1.mp3
│    ├─── track2.mp3
│    ├─── track3.mp3
│    └─── ...
│
├─── images/
│    ├─── logo.png
│    ├─── background.jpg
│    └─── album-art.png
│
└─── README.txt
```

JAVASCRIPT LOGIC (STEP BY STEP)

JavaScript acts as the controller/brain of the Nitonia Music Player. It:

- loads and manages the playlist,
- controls the HTML5 <audio> element,
- updates UI elements (play/pause button, progress bar, current time),
- handles user input (clicks, keyboard shortcuts), and
- provides optional features (shuffle, repeat, save preferences).

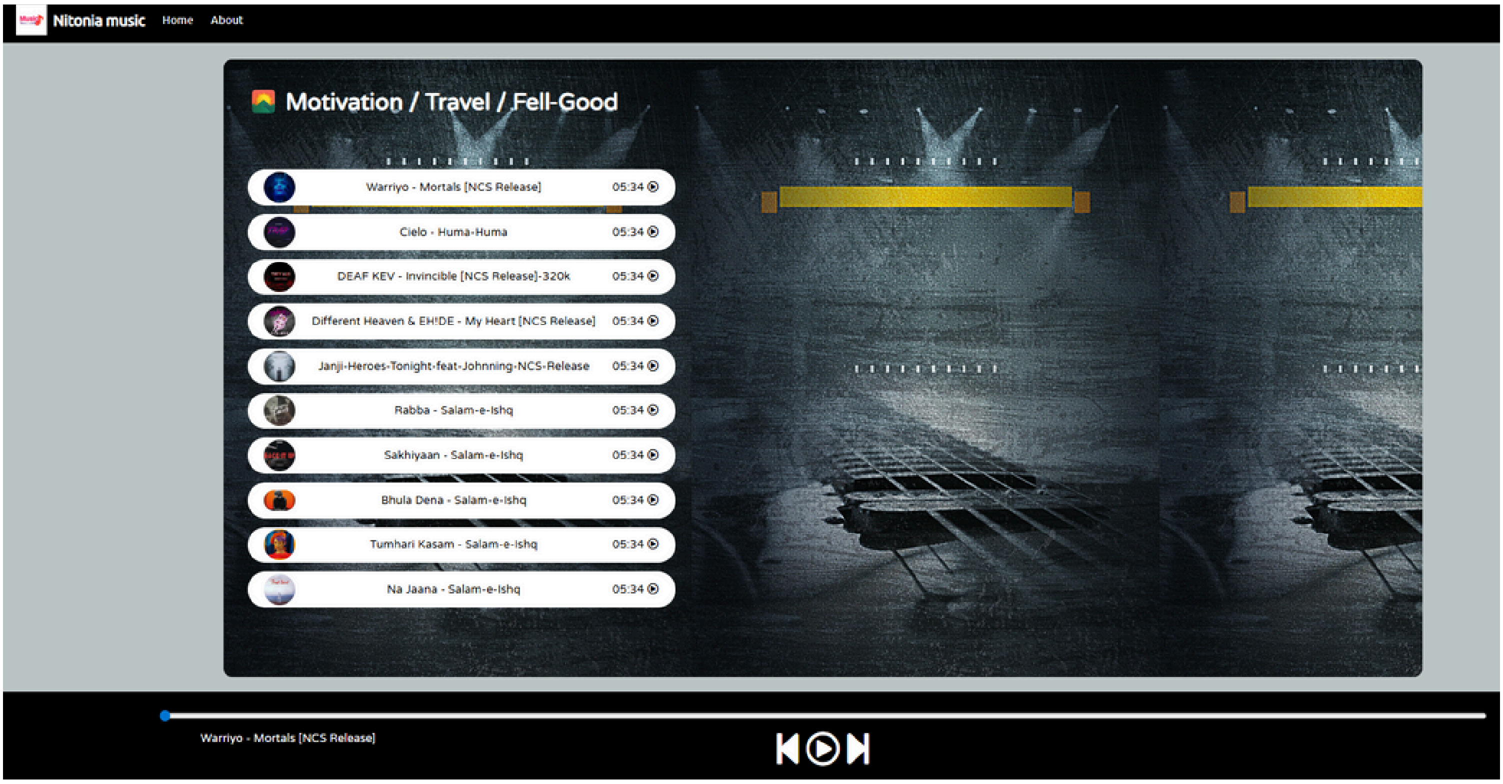
Main Data Structures

{ id: 0, title: "Song Title", src: "music/song.mp3", duration: "03:20", cover: "images/cover.png" }

- currentIndex — index of the currently loaded/playing track.
- audio — reference to an HTMLAudioElement (new Audio() or document.querySelector('audio')).

LIVE DEMO / HOW TO RUN THE PROJECT

The Nitonia Music Player is a front-end web application that runs entirely in a web browser without requiring any special server setup or installation. Since it is built using HTML, CSS, and JavaScript, it can be executed on any device that supports a modern browser such as Google Chrome, Mozilla Firefox, Microsoft Edge, or Safari.



Motivation / Travel / Fell-Good

Warriyo - Mortals [NCS Release]	05:34
Cielo - Huma-Huma	05:34
DEAF KEV - Invincible [NCS Release]-320k	05:34
Different Heaven & EH!DE - My Heart [NCS Release]	05:34
Janji-Heroes-Tonight-feat-Johnning-NCS-Release	05:34
Rabba - Salam-e-Ishq	05:34
Sakhiyaan - Salam-e-Ishq	05:34
Bhula Dena - Salam-e-Ishq	05:34
Tumhari Kasam - Salam-e-Ishq	05:34
Na Jaana - Salam-e-Ishq	05:34

Warriyo - Mortals [NCS Release]

Interaction with the Player

Once the website is open, you can interact with the player using the following steps:

- Play / Pause Music
- Click on the play button to start a song.
- Click again to pause.
- Next / Previous Track
- Use the next and previous buttons to navigate between songs in the playlist.
- Progress Bar Interaction
- The progress bar shows the current time of the song
- Click anywhere on it to jump to a specific point.
- Volume Control
- Adjust volume using the slider at the bottom.
- Use the mute button to temporarily disable sound.
- Playlist Navigation
- Click any song name in the playlist to start playing it instantly.
- The active song is automatically highlighted.
- Keyboard Shortcuts (Optional)
- Spacebar: Play/Pause
- Right Arrow: Next Song
- Left Arrow: Previous Song
- Up/Down Arrows: Volume Control

Folder Setup

```
Nitonia-Music/
│
├── index.html
├── css/
│   └── styles.css
├── js/
│   └── app.js
├── music/
│   ├── track1.mp3
│   ├── track2.mp3
│   └── ...
└── images/
    └── logo.png
```

CONCLUSION

The project titled “Creating a Nitonia Music Player using HTML, CSS, and JavaScript” was undertaken to design and develop a fully functional, responsive, and interactive web-based music player using only front-end technologies. Through this project, the integration of multimedia elements with web development skills has been successfully demonstrated, showing how a simple idea can evolve into a complete and engaging web application

The project “Creating a Nitonia Music Player using HTML, CSS, and JavaScript” was successfully designed and developed as a simple and functional web-based music player. It allows users to play, pause, change songs, control volume, and view an interactive playlist through a clean and responsive interface.

This project helped in understanding how HTML, CSS, and JavaScript work together to create interactive web applications. It also improved practical knowledge of DOM manipulation, event handling, and responsive design.

References

1. W3Schools – HTML, CSS, and JavaScript Tutorials
2. 🔗 https://www.w3schools.com/
3. MDN Web Docs – Web Development Guides and Documentation
4. 🔗 https://developer.mozilla.org/
5. Free Music Archive – For copyright-free and open-source music tracks
6. 🔗 https://freemusicarchive.org/
7. YouTube Tutorials – Front-end web development and JavaScript-based music player projects