ABSTRACT

Music is one of the best ways to relieve pressure in today’s stressful lifestyle. Media player software is a type of application software for playing multimedia computer files like audio and video files. Music Player is a software that plays audio files encoded in MP3 and other audio formats. On the software side, applications that reside in the user's computer, such as iTunes, Windows Media Player and RealPlayer, are used to organize a music collection, play audio files and rip music from a CD.

Media players commonly display standard media control icons from physical devices such as tape recorders and CD players, such as play, pause, fast forward, rewind, and stop buttons. In addition, they generally have progress bars (or "playback bars"), which are sliders, to locate the current position in the duration of the media file.

The purpose of the project is to implement a music player app using linked list as a data structure. Here, the linked list data structure is used to store the audio file in the form of MP3 and .WAV format and navigate through different music files. Meanwhile, this software can play, pause and select songs with the use of play/pause, forward and previous button according to the songs, as well as set them up. We can increase or decrease volume using the volume slider. Overall, the use of linked list data structure makes it easy to store and navigate through different audio files, and is easy and simple to build the application.

The project is implemented using HTML, CSS and JavaScript. Here HTML and CSS are used to make the user interface using different slider bars and buttons. Here JavaScript is used to make Doubly Linked List, store the music files in each node, and navigate through them. In addition, actions for each buttons and slider bars are done using JavaScript.

TABLE OF CONTENT

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Overview

Music is one of the best ways to relieve pressure in today’s stressful lifestyle. Media player software is a type of application software for playing multimedia computer files like audio and video files. Music Player is a software that plays audio files encoded in MP3 and other audio formats. On the software side, applications that reside in the user's computer, such as iTunes, Windows Media Player and RealPlayer, are used to organize a music collection, play audio files and rip music from a CD.

Media players commonly display standard media control icons known from physical devices such as tape recorders and CD players, such as play, pause, fast forward, rewind, and stop buttons. In addition, they generally have progress bars (or "playback bars"), which are sliders to locate the current position in the duration of the media file.

The purpose of the project is to implement a music player app using linked list as a data structure. Here linked list data structure is used to store the music file in the form of MP3 and .WAV format and navigate through different music files. Meanwhile, this software can play, pause and select songs with the use of forward and previous button according to sets requirement as well as set up song. We can increase or decrease volume using the volume slider. Overall, the use of linked list data structure makes it easy to store and navigate through different audio files easy and simple to build the application.

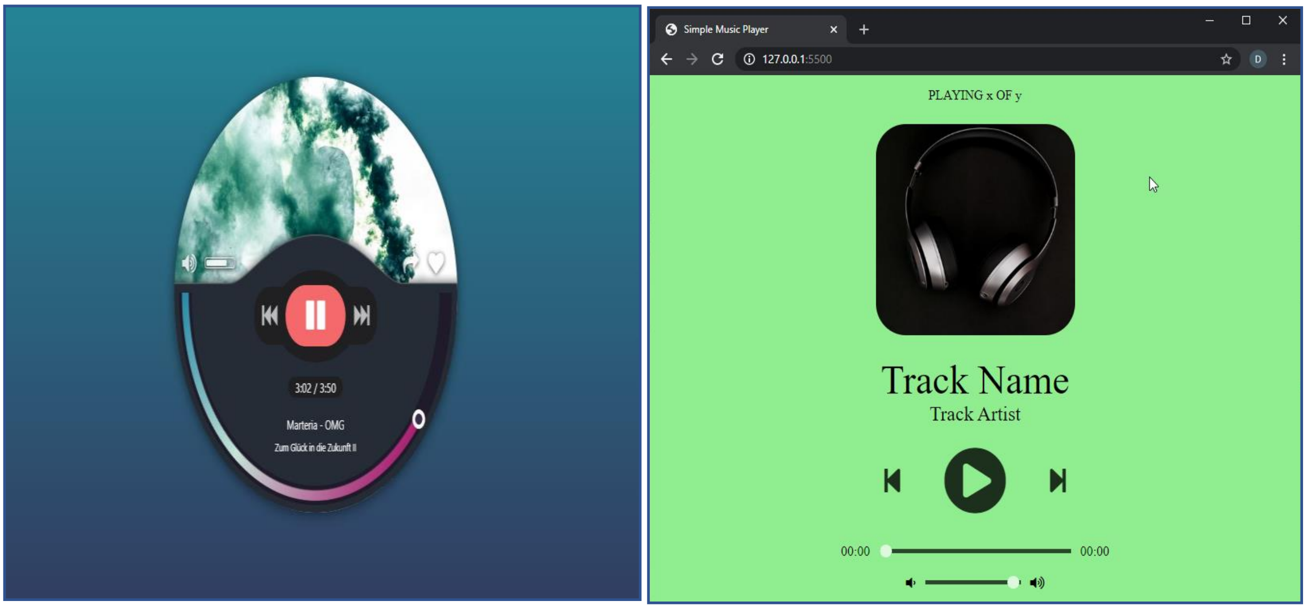

Figure 1.1: Examples of Music Player applications

1.2 Concepts used for the Application

Linked List is the data structures that is used in the application, due to its versatility and universality. Linked List can be defined as a collection of objects called nodes that are randomly stored in the memory. More specifically Doubly Linked List is used in the project in order to store the music files and navigate to different nodes thus accessing the music file. Doubly linked list is a complex type of linked list in which a node contains a pointer to the previous as well as the next node in the sequence. Therefore, in a doubly linked list, a node consists of three parts: node data, pointer to the next node in sequence (next pointer), pointer to the previous node (previous pointer).

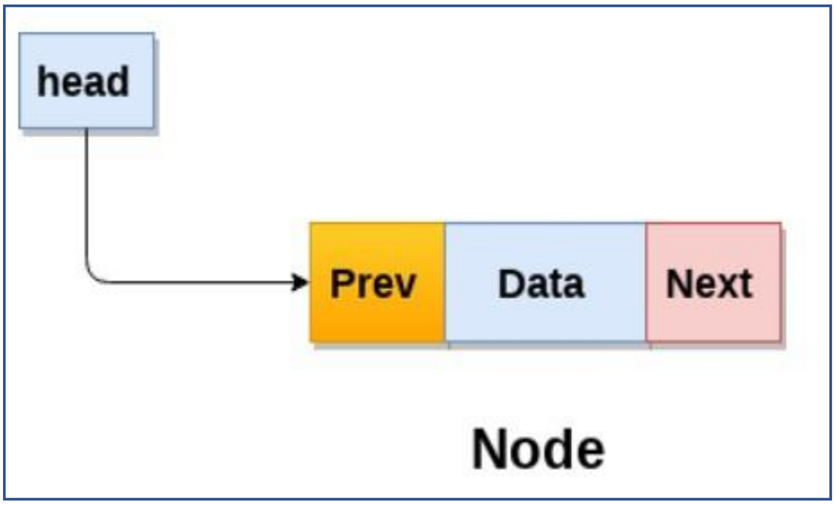

Figure 1.2: Representation of Node in Doubly Linked List

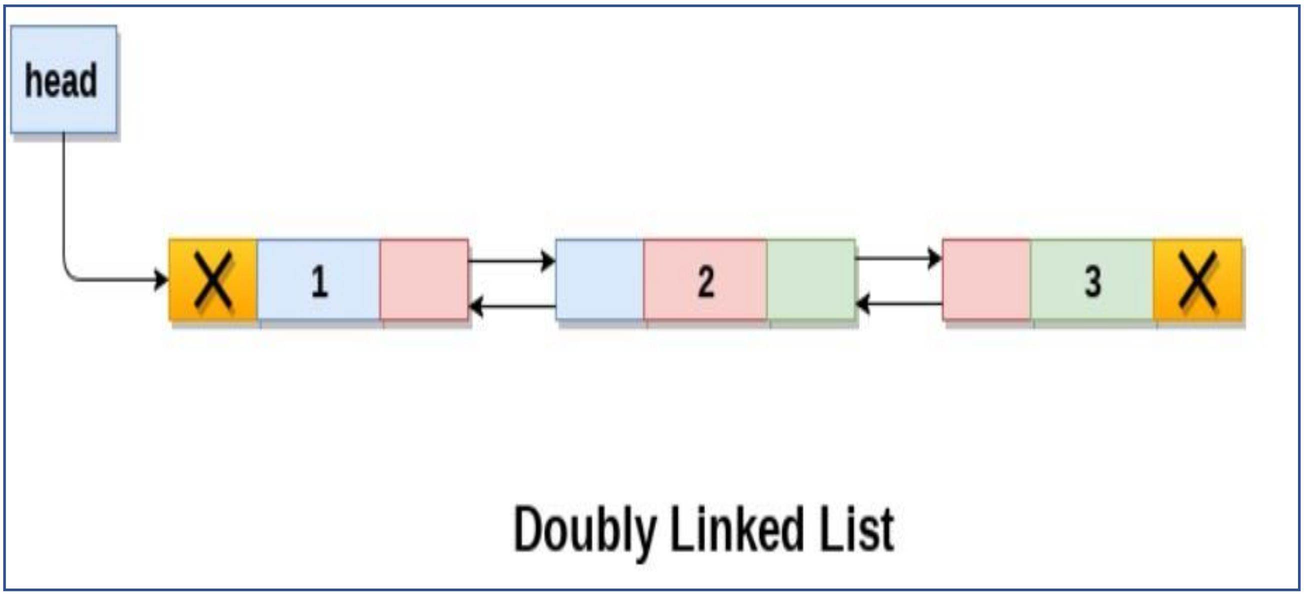

Figure 1.3: Representation of Doubly Linked List

1.3 Problem Statement

The problem we opted was to develop a Music Player using linked list. Here linked list data structure is used to store the music file in the form of MP3 and .WAV format and navigate through different music files. Meanwhile, this software can play, pause and select songs with the use of forward and previous button according to sets requirement as well as set up song. Additionally, HTML and CSS is used to create the Basic user interface, linked list and button functionalities using JavaScript.

1.4 Objectives

The main objective of this project is to entirely design a music player application that is user friendly and is easy to navigate through songs using doubly linked list as the data structure and at the same time, learning different data structures and algorithms, tools and disciplines required for the development of the music player application. In order to be able to achieve this main objective, the project has been divided into multiple targets. The general targets include:

- Create a UI for the music player application using HTML and CSS
- Implement Doubly Linked list to store the different music files in each node
- Create suitable buttons and slider bars for navigation of different music
- Store the MP3 and .WAV format music files in linked list using JavaScript and also implement actions to the buttons used and enhance the UI

CHAPTER 2

METHODOLOGY

2.1 Analysis

To move forward with the project, multiple concepts from HTML, CSS and JavaScript were searched that could be helpful in the development of the music player application. The main part of the project was to implement one of the data structures. In this project, Doubly Linked List is used as one of the Data Structure to store the music files and navigate through different nodes to access the music files. The basic UI design consists of a background image, different buttons and slide bars for the navigation of songs and for the adjustment of the volume. A basic flowchart of the, music player is shown below.

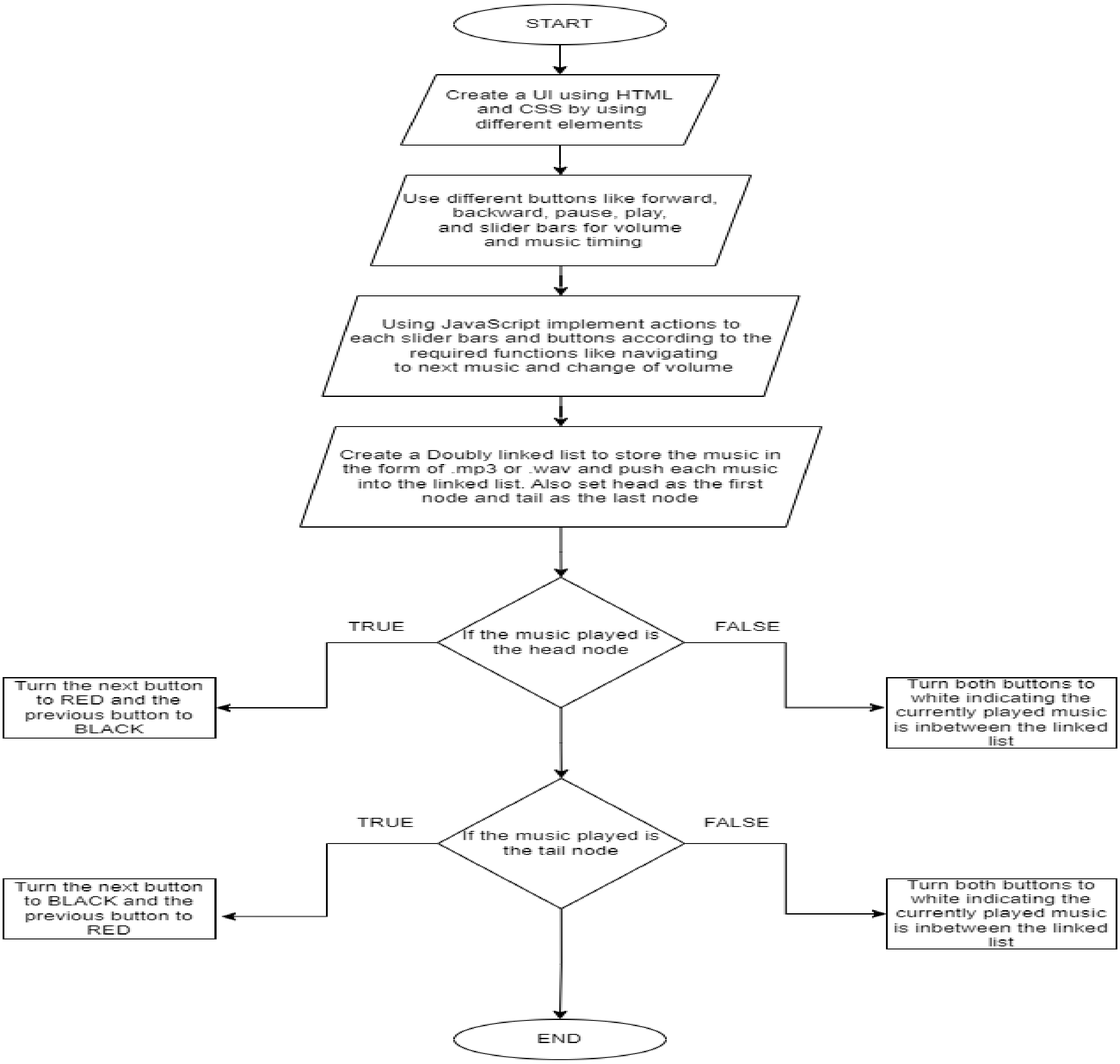
START
Create a UI using HTML and CSS by using different elements
Use different buttons like forward, backward, pause, play, and slider bars for volume and music timing
Using JavaScript implement actions to each slider bars and buttons according to the required functions like navigating to next music and change of volume
Create a Doubly linked list to store the music in the form of .mp3 or .wav and push each music into the linked list. Also set head as the first node and tail as the last node
If the music played is the head node
TRUE
FALSE
Turn the next button to RED and the previous button to BLACK
Turn both buttons to white indicating the currently played music is inbetween the linked list
If the music played is the tail node
TRUE
FALSE
Turn the next button to BLACK and the previous button to RED
Turn both buttons to white indicating the currently played music is inbetween the linked list
END

Figure 2.1: Basic Flowchart of the Design

2.2 Design

The basic structure and functionality of the project is based on doubly linked list where the music file is stored and is navigated using next and previous buttons. The basic logic of the application is as follows:

- After launching the landing page, the music player user interface is displayed and all the necessary buttons and slider bars are displayed.

- All the necessary actions to the buttons are provided using JavaScript and the music files are stored in a linked list and is navigated using next and previous button.

- Here the music file address is pushed into the node and when the specific music is navigated the music file is played.

- Volume can be adjusted through a slider bar and the music time line track is also viewed through a slider bar.

- In order to indicate the head node and tail node, special button features are implemented.

The basic elements considered for designing of the music player application are as follows:

- Development of music player UI using HTML and CSS using different elements such as buttons, slider bars and making the page interactive and styling it through different elements of CSS.

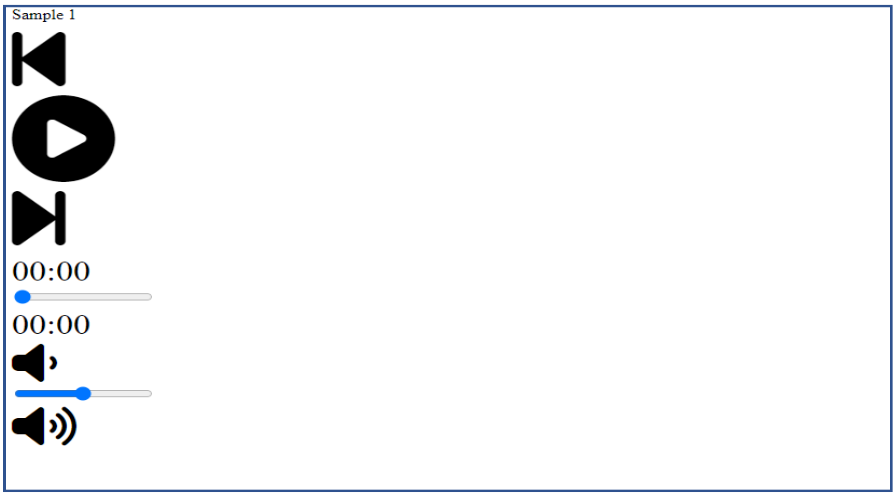

Figure 2.2: Basic Outline of the music Player

- Implement the controls for the next, previous, pause, play buttons and music timeline slider bar and volume slider bar using JavaScript.
- Implement Doubly Linked List to store the music file in each node. Use the Head to point the first node and tail to point the last node, and implement button action indicating the beginning and end of the playlist.
- If the music playing is the head node make the previous button to BLACK and next button to RED indicating to move forward to access the next music file. Also if the music playing is the tail node then make the next button BLACK and previous button RED indicating to move backward to replay the song again.

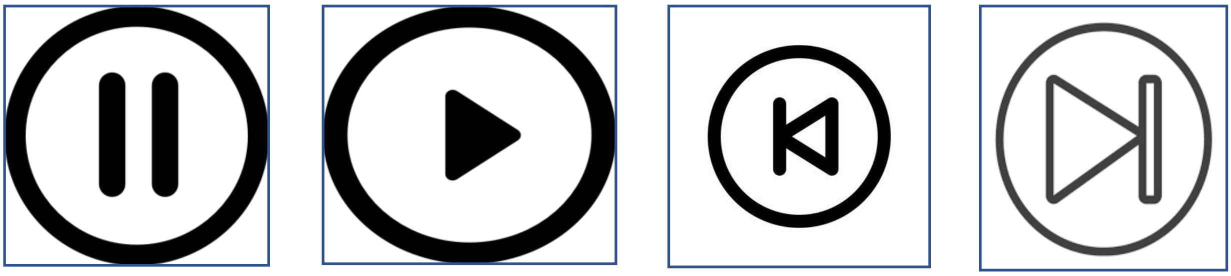

Figure 2.3: Different Buttons used in the music player

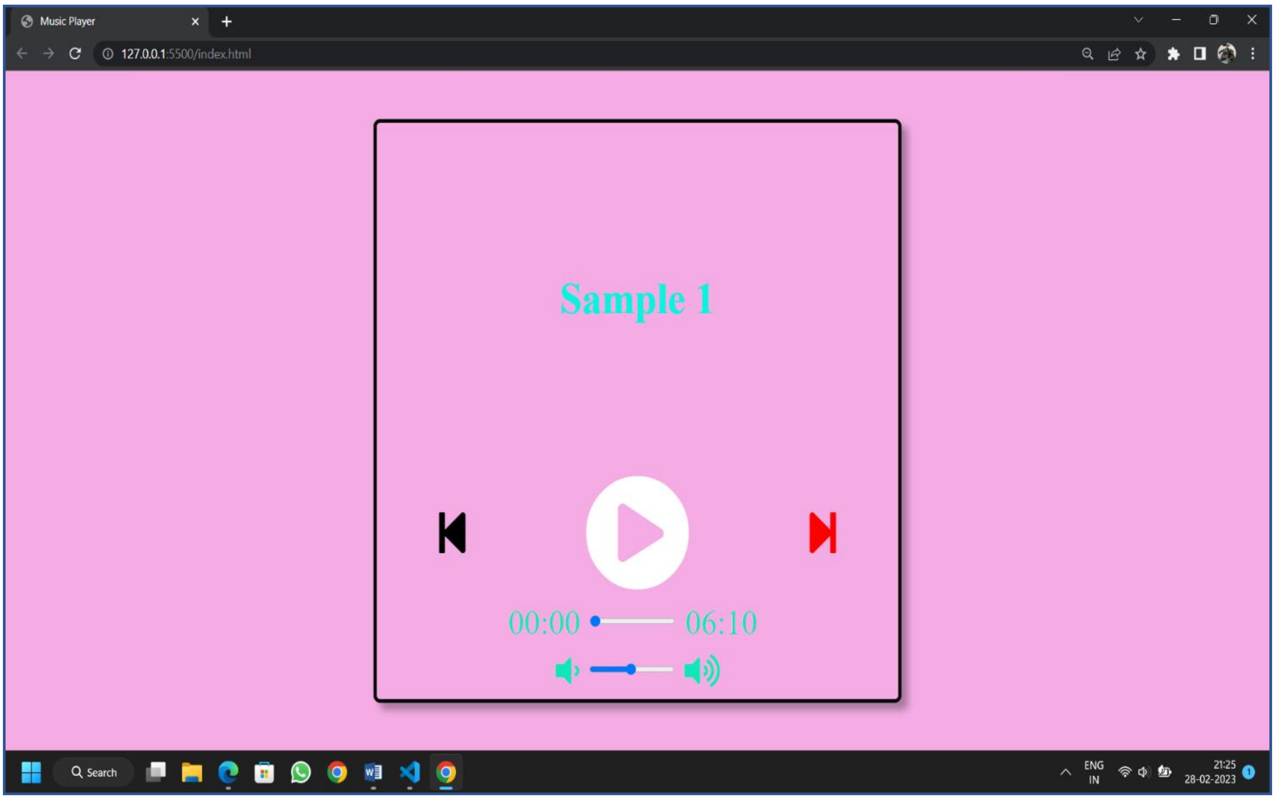

Figure 2.4: Styling using CSS and implementing actions to the buttons and slider bars

CHAPTER 3

EXPERIMENTAL RESULTS

This chapter presents the results of different tasks made to complete the music player application using linked list. The results include tests made during the implementation of the project.

All the objectives of the project were met satisfactorily. The UI design and controls for the music player worked hand-in-hand to provide the required outcome. The interface provides the user to listen and navigate through different music using different buttons and slider bars. The storage of music using Doubly Linked list makes it easy to add more music files into the doubly linked list through addition of nodes.

- When the application is started, it is guided to a UI landing page made using HTML and CSS. The UI of the music player has a background image and different buttons and slider bars that has specific control.

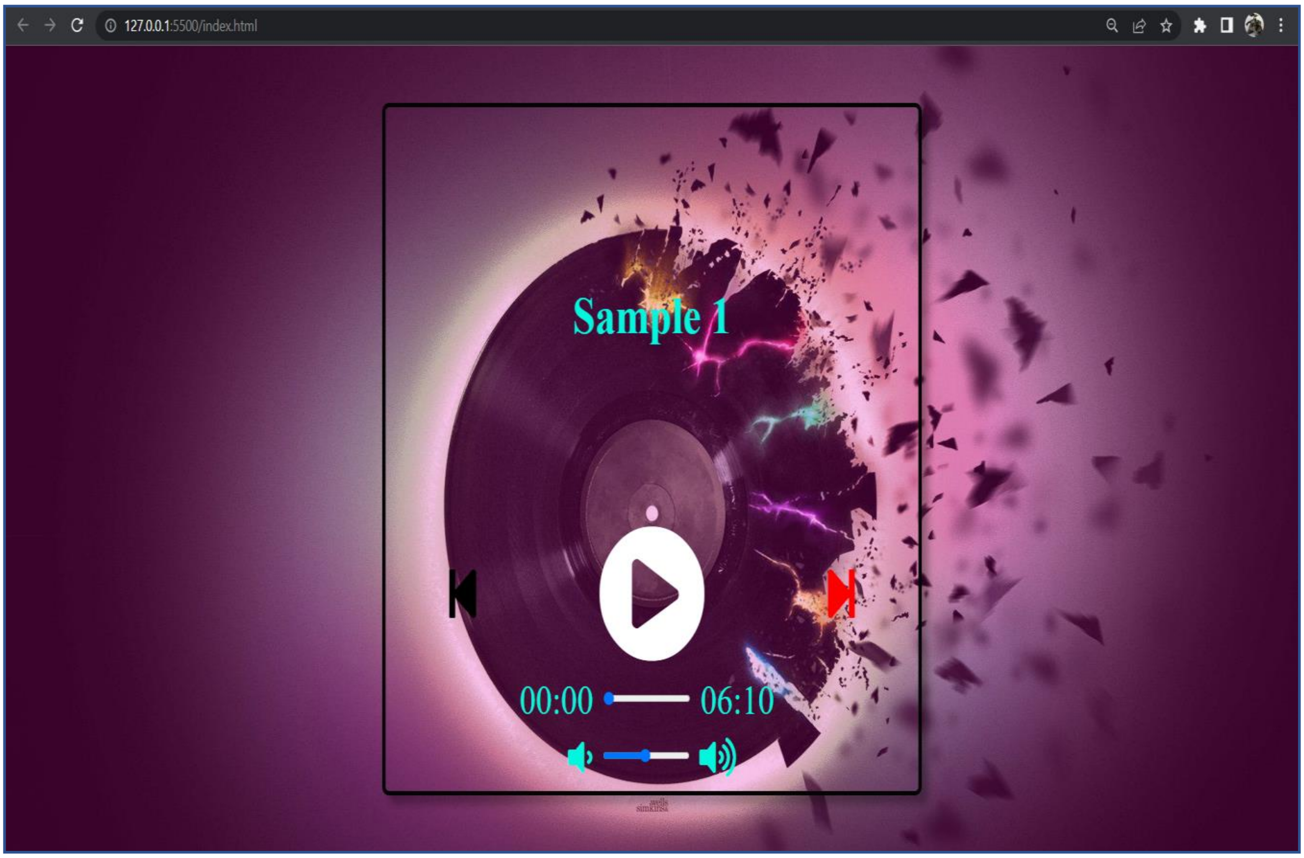

Figure 3.1: Landing page of the music player application

- The music files are stored using doubly linked list and each music file is stored in a node and is used for accessing music files to play. The Buttons are given specific action that can be performed in order to navigate through different music and also pause and play the music. The music timeline is shown through a slider bar and the volume can be adjusted using a slider bar.

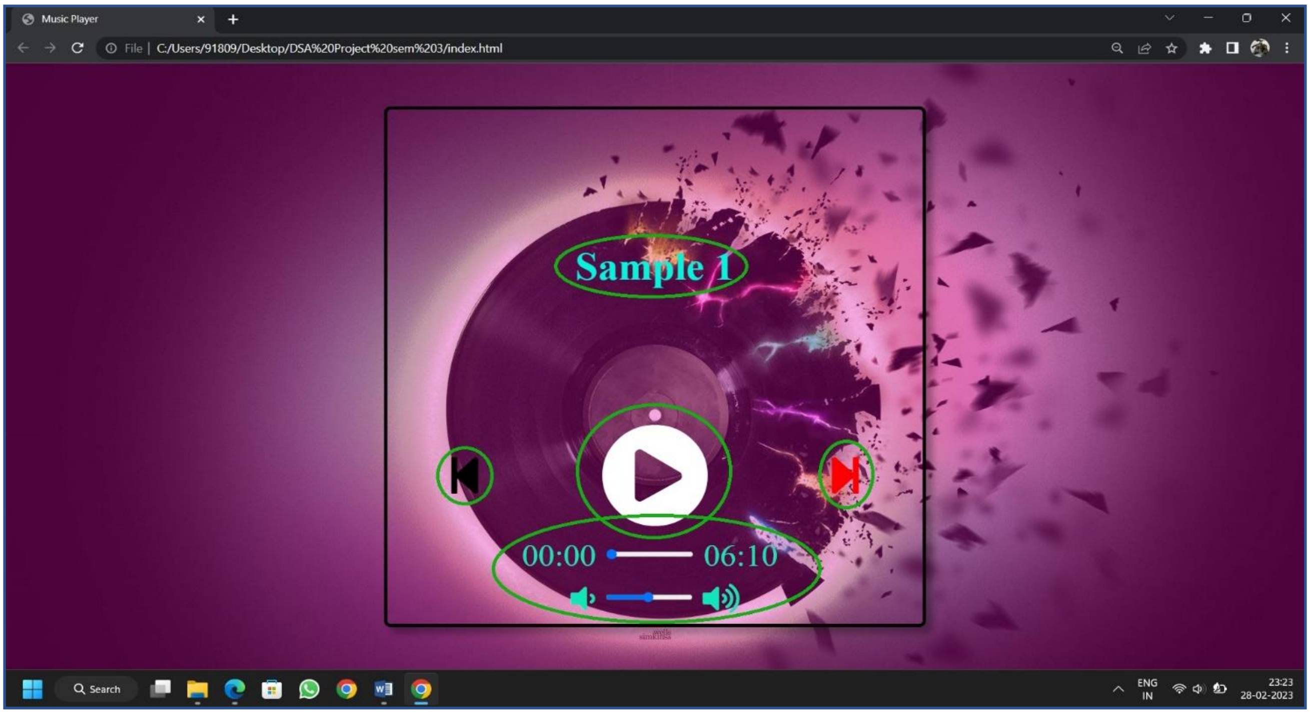
Music Player
File | C:/Users/91809/Desktop/DSA%20Project%20sem%203/index.html
Sample 1
00:00
06:10
Search
ENG
IN
23:23
28-02-2023

Figure 3.2: Representation of different buttons and slider bars for different functions

RED indicating to move forward to access the next music file. Also if the music playing is the tail node then make the next button BLACK and previous button RED indicating to move backward to replay the song again.

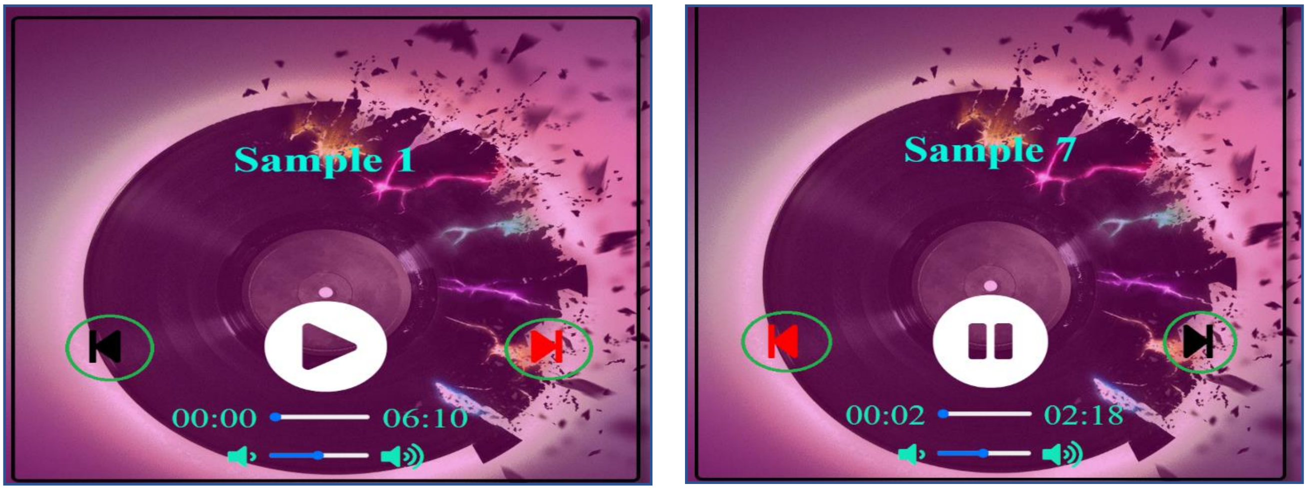

Figure 3.3: Special actions of the next and previous button

CHAPTER 4

CONCLUSION

The main objective of this project is to entirely design a music player application that is user-friendly and is easy to navigate through songs using doubly linked list as the data structure, and at the same time, learning different data structures and algorithms, tools and disciplines required for the development of the music player application. In order to implement all the features HTML, CSS and JavaScript were used and the required results were achieved. The implementation of the correct data structure, which is the Doubly Linked List and the action for the buttons, was the main part in the project.

As a result, the Music Player application made using Linked List was designed and also the user interface was designed letting the user listen to the music and also made the application user friendly and simple in order to navigate through different songs using different buttons. Special features such as making the previous button BLACK in colour when the music playing is the head node and making the next button to RED indicating to move forward to access the next music file. Also, making the next button BLACK if the music playing is the tail node and the previous button RED indicating to move backward to replay the song again was also implemented, thus, indicating the first and the last nodes of the linked list.

The achievements during the process were:

- Create an UI for the music player application using HTML and CSS
- Implement Doubly Linked list to store the different music files in each node
- Create suitable buttons and slider bars for navigation of different music
- Store the MP3 and .WAV format music files in linked list using JavaScript and also implement actions to the buttons used and enhance the UI

CHAPTER 5

FUTURE SCOPE

The current music player application can have many features implemented in the future. The list of features and improvisations that can be done are as follows:

- Allowing users to make a custom playlist of songs based on their desire.
- Allow the users to sort the songs based on the order in which they want to listen to them.
- Let the app be connected to the internet and display recommended or similar songs based on the users playlist
- Get the lyrics of the songs and display them while the song is being played.
- Allow users to categorise songs into different genres like party songs, romantic songs, inspirational songs, etc.
- Develop a better and more Interactive user interface for the music application.

References

1. https://en.wikipedia.org/wiki/Media_player_software
2. https://www.javatpoint.com/singly-linked-list
3. https://www.javatpoint.com/doubly-linked-list
4. https://www.w3schools.com/html/
5. https://www.w3schools.com/css/
6. https://www.w3schools.com/js/
7. https://www.geeksforgeeks.org/implementation-linkedlist-javascript/
8. https://www.freecodecamp.org/news/html-button-onclick-javascript-click-event-tutorial/