MINI PROJECT II

2020-21

"Song Playlist Web Application"

REPORT

Team Members:

Anushka Pandey

University Roll No.-181500125(G-16)

Naman Maheshwari

University Roll No.-181500411(H-14)

Saurabh Tripathi

University Roll No.-181500641(D-55)

Supervised By:

Ms. Priya Agarwal

Assistant Professor

Department of Computer Science Engineering & Applications

Acknowledgement

In the present world of competition there is a race of existence in which those are having will to come forward succeed. Project is like a bridge between theoretical and practical working. With this willing we joined this particular project. First of all, we would like to thank the supreme power the Almighty God who is obviously the one has always guided me to work on the right path of life. Without his grace this project could not become a reality. Next to him are my parents, whom we are greatly indebted for me brought up with love and encouragement to this stage. We are feeling oblige in taking the opportunity to sincerely thanks to **Ms. Priya Agarwal**. At last, but not the least We are thankful to all my teachers and friends who have been always helping and encouraging me throughout the year. We have no valuable words to express my thanks, but my heart is still full of the favours received from every person.

Contents

Abstract

As we know, the present world is using advanced 4G and 5G technology, so it is easy to spend our free time by watching movies, playing indoor and outdoor games expenses and we can spend time on our hobbies but listening music is different one.

Recent studies confirm that humans respond and react to music and that music has a high impact on person's brain activity. An average person listens up to four hours of music every day . People tend to listen to music based on their mood and interests. This project focuses on creating a web application to suggest songs for user based on their mood. Once the mood is recognized, the system suggests a play-list for that mood, saving a lot of time for a user over selecting and playing songs manually.

Introduction

Music is an important entertainment medium. With advancement of technology, the optimization of manual work has gained a lot of attention. Currently, there are many traditional music players that require songs to be manually selected and organized. User, have to create and update play-list for each mood, which is time consuming. Some of the music players have advanced features like providing lyrics and recommending similar songs based on the singer or genre. Although some of these features are enjoyable for user, there is room to improve in the field of automation when it comes to music players. Selecting songs automatically and organizing these based on the user's mood gives user's a better experience. This can be accomplished through the system reacting to the user's emotion, saving time that would have been spent entering information manually.

In this project, emotions can be expressed through selecting an emoji. For the system to understand a user's mood, we use emojis depicting different facial expressions and then music player plays its role.

Motivation

Nowadays, music is everywhere. From small television commercials, music videos and shopping centres to the more traditional music sell (albums, both on physical or digital format, and tickets), radio play or live events (concerts, gigs, festivals, etc), we cannot help consider now it as an industry. This can be explained through economic reasons but also through emotional ones: why some concerts get sold out in just a few minutes1 , even when the tickets price is high?1 Why are there true fan communities online which create means to bring some of their most beloved artists to their countries or cities (petitions, organizing gigs/concerts, etc)2? And why big part of humanitarian and charity events are based in (or incorporate) music Certainly not only because of temporary fashions or trends. Even some of the most popular social networks on the web are music-oriented: MySpace4 (with more than 110 million active users5) and Last.fm6 (with more than 30 million users). People need to relate to the way they feel (or want to feel), and music is one of the most utilized means to achieve it. This is done not only by relating to the instrumental part of the songs, but in the case of the existence of voice, the analysis of the lyrical content.

Objective

The main objective of this is, as the title suggests, the design and implementation of an application that provides the user with an automatically generated playlist of songs based on the mood of a song specified by the user and given as input. In another perspective, it can be used to generate a playlist to complement the mood of the user – e.g., the user introduces that his mood is anxious, it will be expected that the playlist will include some relaxing music. The song given as input will be an existing song in the application's database (songs are stored in different folders). The selection would be done based on a database consisting of music files/musical pieces. The pieces that would fit the mood specified by the user (or the pieces that would complement the specified mood) would be added to the final playlist presented to it.

Problem Statement

Music listeners have tough time creating and segregating the play-list manually when they have hundreds of songs. It is also difficult to keep track of all the songs: sometimes songs that are added and never used, wasting a lot of device memory and forcing the user to find and delete songs manually. Users also have difficulty to re-organize and playing music when play-style varies. Users have to manually change or update

each song in their play-list every time. The sequence of songs in a play-list might not be the same every time, and songs that a user wants to listen frequently might not be given priority or might be left out from the list. Currently, there are no applications that allows users to play songs on-the-go without selecting songs manually or from a play-list.

Software/Hardware Requirements

Hardware:

A PC or Laptop having Processor intel core 3 or above RAM 8.0 GB or above Hard Disk Drive 500 GB or above

Software :

- ✓ Window 7,8 or 10
- ✓ Visual Studio code
- ✓ Google Chrome
- ✓ GitHub

Technologies Used

HTML

The Hypertext Markup Language, or HTML is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.

Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, implements and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by *tags*, written using angle brackets

HTML can embed programs written in a scripting language such as JavaScript, which affects the behaviour and content of web pages. Inclusion of CSS defines the look and layout of content. The World Wide Web Consortium (W3C), former maintainer of the HTML and current maintainer of the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997.

HTML is a markup language that web browsers use to interpret and compose text, images, and other material into visual or audible web pages. Default characteristics for every item of HTML markup are defined in the browser, and these characteristics can be altered or enhanced by the web page designer's additional use of CSS. Many of the text elements are found in the 1988 ISO technical report TR 9537 *Techniques for using SGML*, which in turn covers the features of early text formatting languages such as that used by the RUNOFF Command developed in the early 1960s for the CTSS (Compatible Time-Sharing System) operating system: these formatting commands were derived from the commands used by typesetters to manually format documents. However, the SGML concept of generalized markup is based on elements (nested annotated ranges with attributes) rather than merely print effects, with also the

separation of structure and markup; HTML has been progressively moved in this direction with CSS.

CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file which reduces complexity and repetition in the structural content as well as enabling the .css file to be cached to improve the page load speed between the pages that share the file and its formatting.

Separation of formatting and content also makes it feasible to present the same markup page in different styles for different rendering methods,

such as on-screen, in print, by voice (via speech-based browser or screen render), and on Braille-based tactile devices. CSS also has rules for alternate formatting if the content is accessed on a mobile device.

The name *cascading* comes from the specified priority scheme to determine which style rule applies if more than one rule matches a particular element. This cascading priority scheme is predictable.

The CSS specifications are maintained by the World Wide Web Consortium (W3C). Internet media type (MIME type) text/css is registered for use with CSS by RFC 2318 (March 1998). The W3C operates a free CSS validation service for CSS documents.

In addition to HTML, other markup languages support the use of CSS including XHTML, plain XML, SVG, and XUL.

JavaScript

JavaScript, often abbreviated as JS, is a programming language that conforms to the ECMAScript specification. JavaScript is high-level, often just-in-time compiled, and multi-paradigm. It has curly-bracket syntax, dynamic typing, prototype - based object orientation, and first-class functions.

Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web. Over 97% of websites use it client side for web page behaviour, often incorporating third-party libraries. All major web browsers have a dedicated JavaScript engine to execute the code on the user's device.

As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative programming styles. It has Application

Programming Interfaces (APIs) for working with text, dates, regular expressions, standard data structures, and the Document Object Mode (DOM).

The ECMAScript standard does not include any input/output (I/O), such as networking, storage, or graphics facilities. In practice, the web browser or other runtime system provides JavaScript APIs for I/O.

JavaScript engines were originally used only in web browsers, but they are now core components of other software systems, most notably servers and a variety of applications.

JavaScript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities.

JavaScript was first known as Live Script**,** but Netscape changed its name to JavaScript, possibly because of the excitement being generated by Java. JavaScript made its first appearance in Netscape 2.0 in 1995 with the name Live Script. The general-purpose core of the language has been embedded in Netscape, Internet Explorer, and other web browsers.

Client-side JavaScript is the most common form of the language. The script should be included in or referenced by an HTML document for the code to be interpreted by the browser.

It means that a web page need not be a static HTML, but can include programs that interact with the user, control the browser, and dynamically create HTML content.

The JavaScript client-side mechanism provides many advantages over traditional CGI server-side scripts. For example, you might use JavaScript to check if the user has entered a valid e-mail address in a form field.

The JavaScript code is executed when the user submits the form, and only if all the entries are valid, they would be submitted to the Web Server.

JavaScript can be used to trap user-initiated events such as button clicks, link navigation, and other actions that the user initiates explicitly or implicitly.

Visual Studio Code

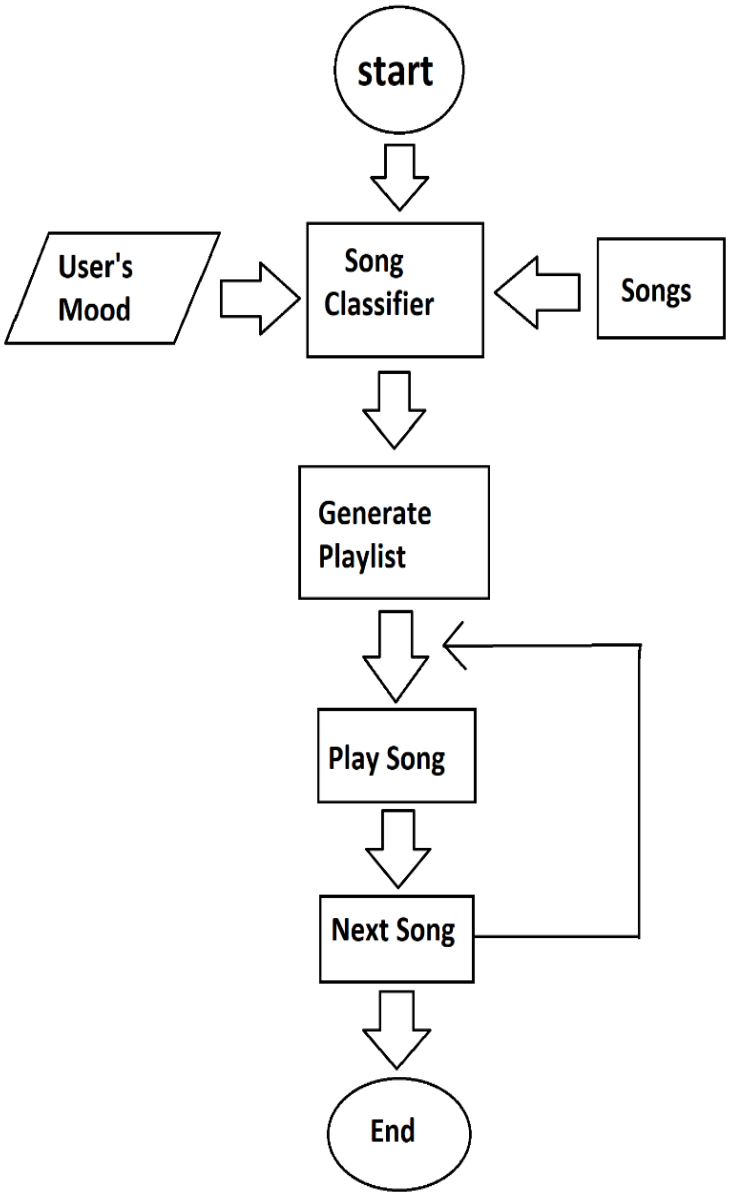
Microsoft has released Visual Studio Code's source code on the Microsoft/VScode (Code - OSS) repository of GitHub, under the permissive MIT License, while the releases by Microsoft are freeware. In the Stack Overflow 2019 Developer Survey, Visual Studio Code was ranked the most popular developer environment tool, with 50.7% of 87,317 respondents reporting that they use it. Visual Studio Code was first announced on April 29, 2015, by Microsoft at the 2015 Build conference. A Preview build was released shortly thereafter. On November 18, 2015, Visual Studio Code was released under the MIT License, having its source code available on GitHub. Extension support was also

announced. On April 14, 2016, Visual Studio Code graduated from the public preview stage and was released to the web.

Instead of a project system, it allows users to open one or more directories, which can then be saved in workspaces for future reuse. This allows it to operate as a language-agnostic code editor for any language. It supports a number of programming languages and a set of features that differs per language. Unwanted files and folders can be excluded from the project tree via the settings. Many Visual Studio Code features are not exposed through menus or the user interface but can be accessed via the command palette.

Visual Studio Code can be extended via extensions, available through a central repository. This includes additions to the editor and language support. A notable feature is the ability to create extensions that add support for new languages, themes, and debuggers, perform static code analysis, and add code linters using the Language Server Protocol.

Data Flow Diagram



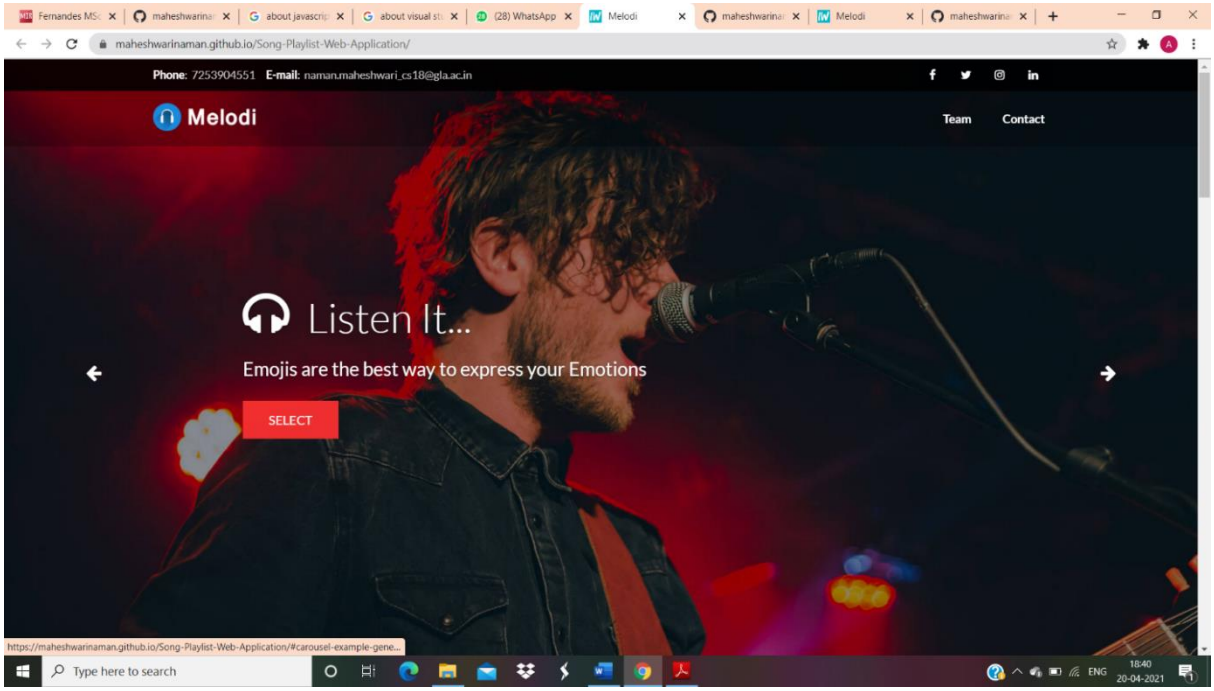
Implementation Details

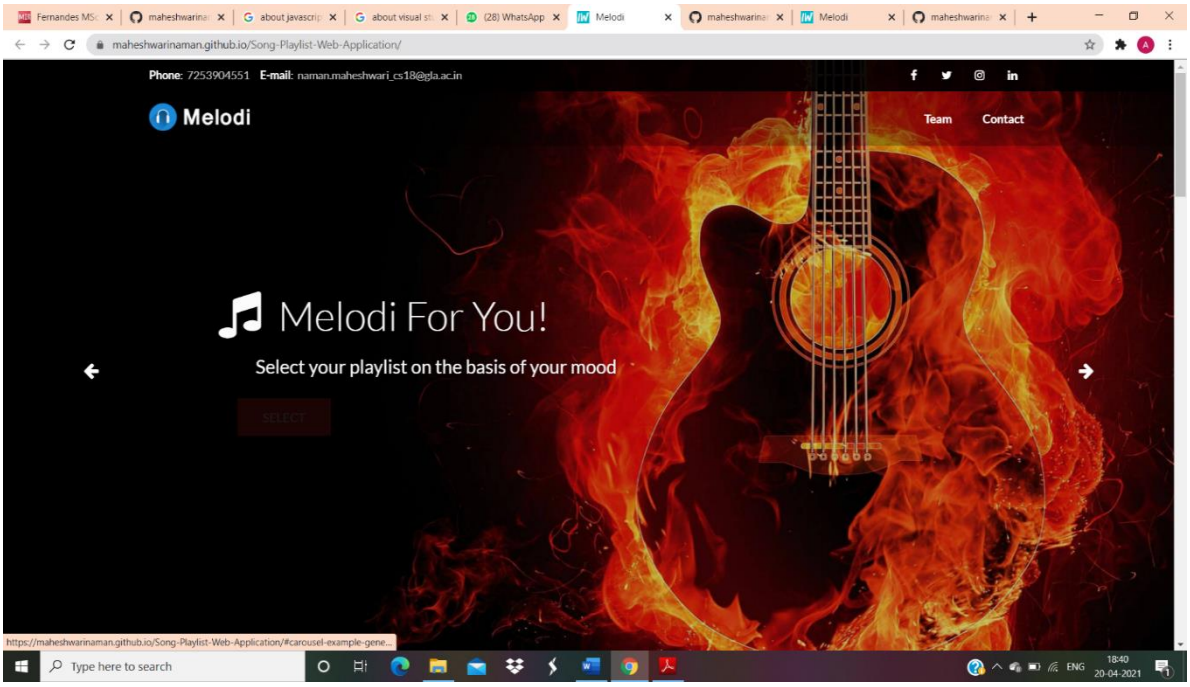
We have designed a website that plays song according to the emoji (representing mood) selected by the user. First there is a homepage, in which we provide the name of our team members and social links and contact information and required links for the mood predictor page.

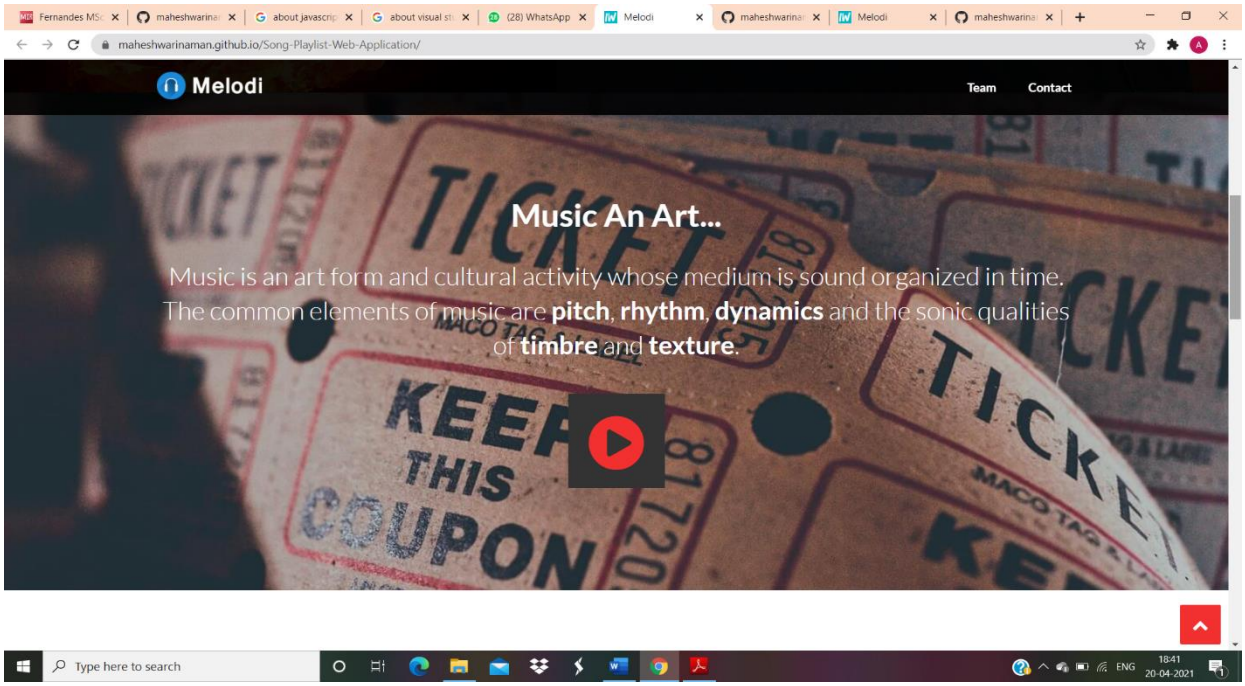
In mood predictor page, we have made a emoji slider in which there are four emotions, angry, sad, happy and love. When the user clicks on emoji based on his/her emotion, he/she will be directed to the music player page, in which we have insert five song playlists for each emotion.

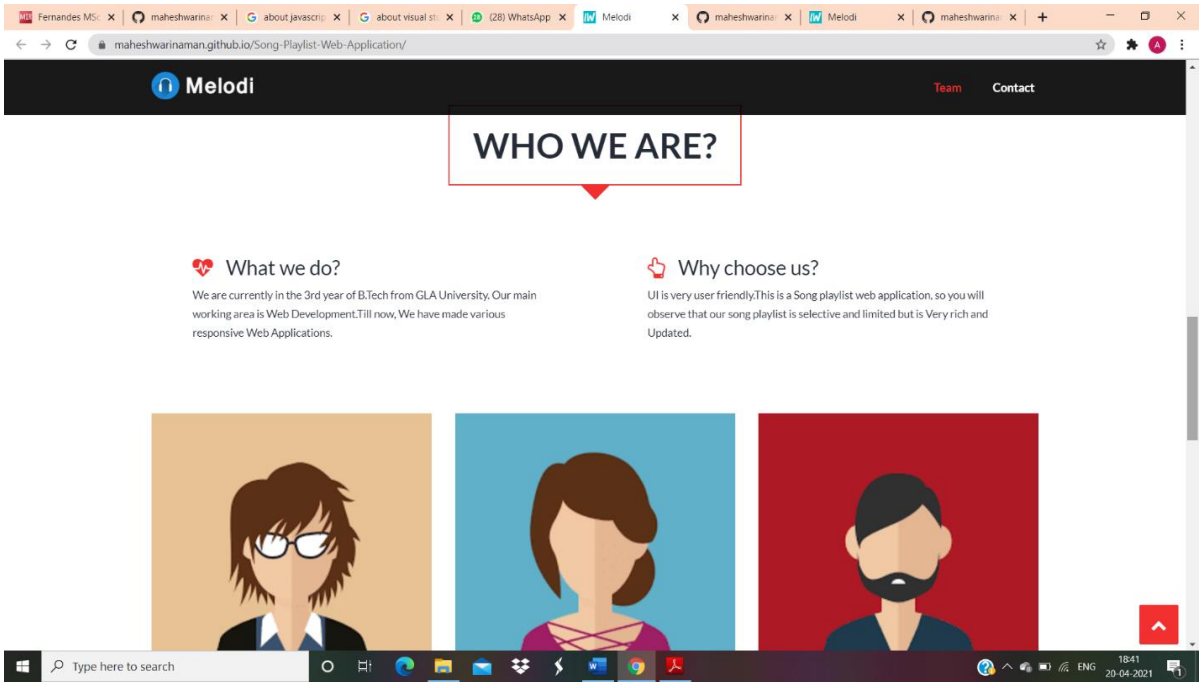
You are also allowed to change the song. You can control volume also. We have also mentioned the name of the singer and an Auto play button is also there

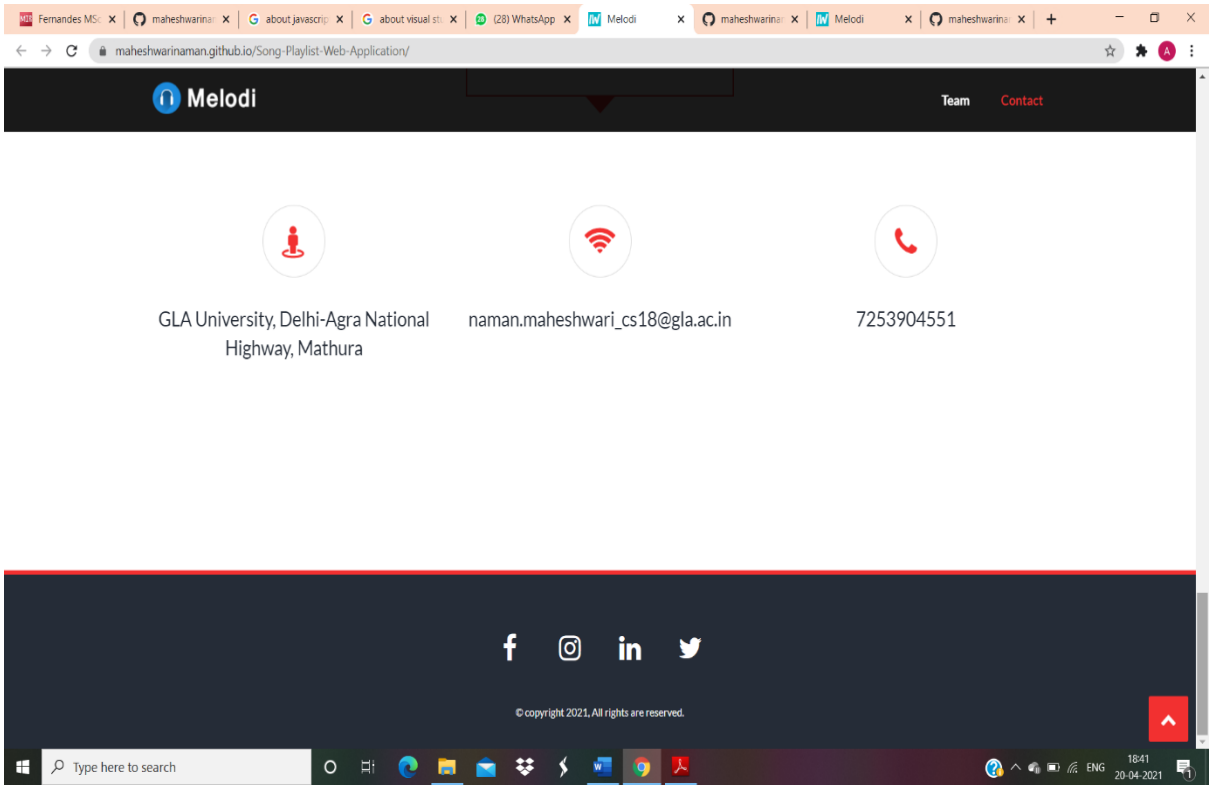
Screenshots

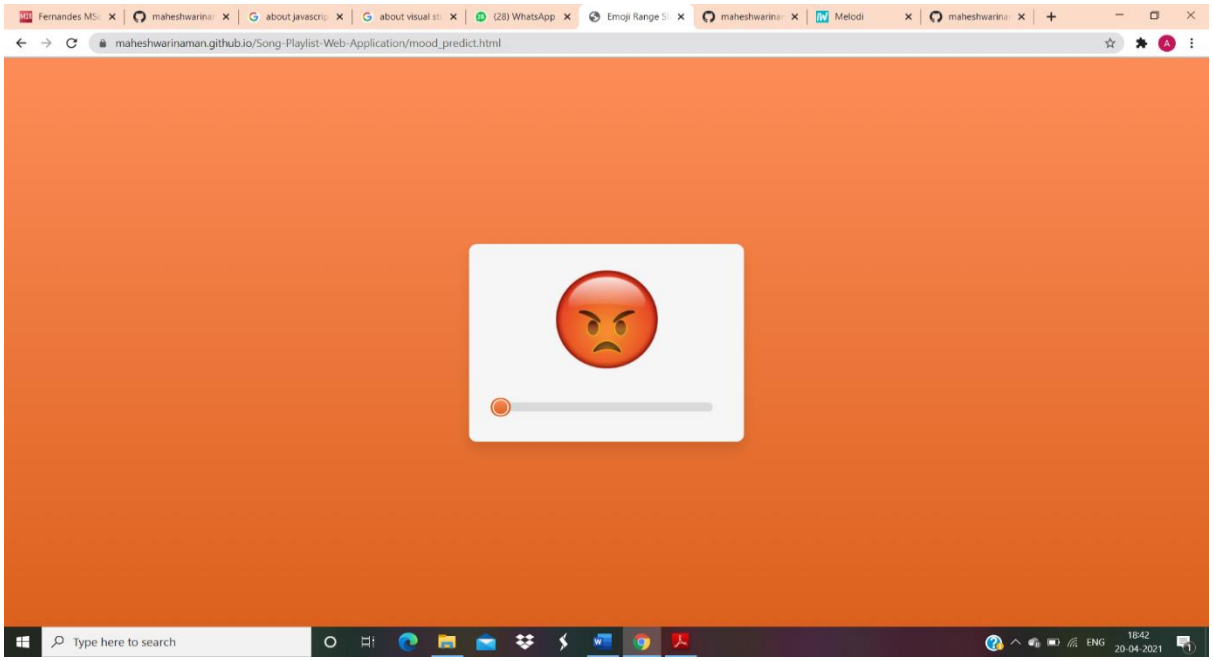


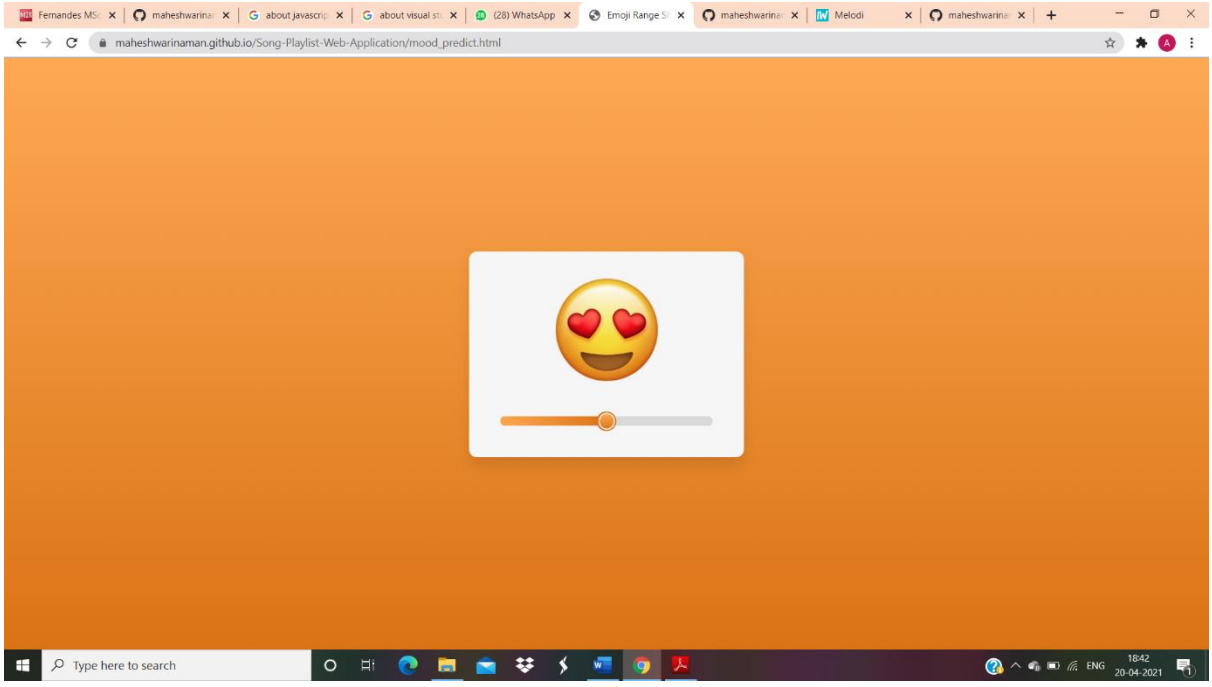


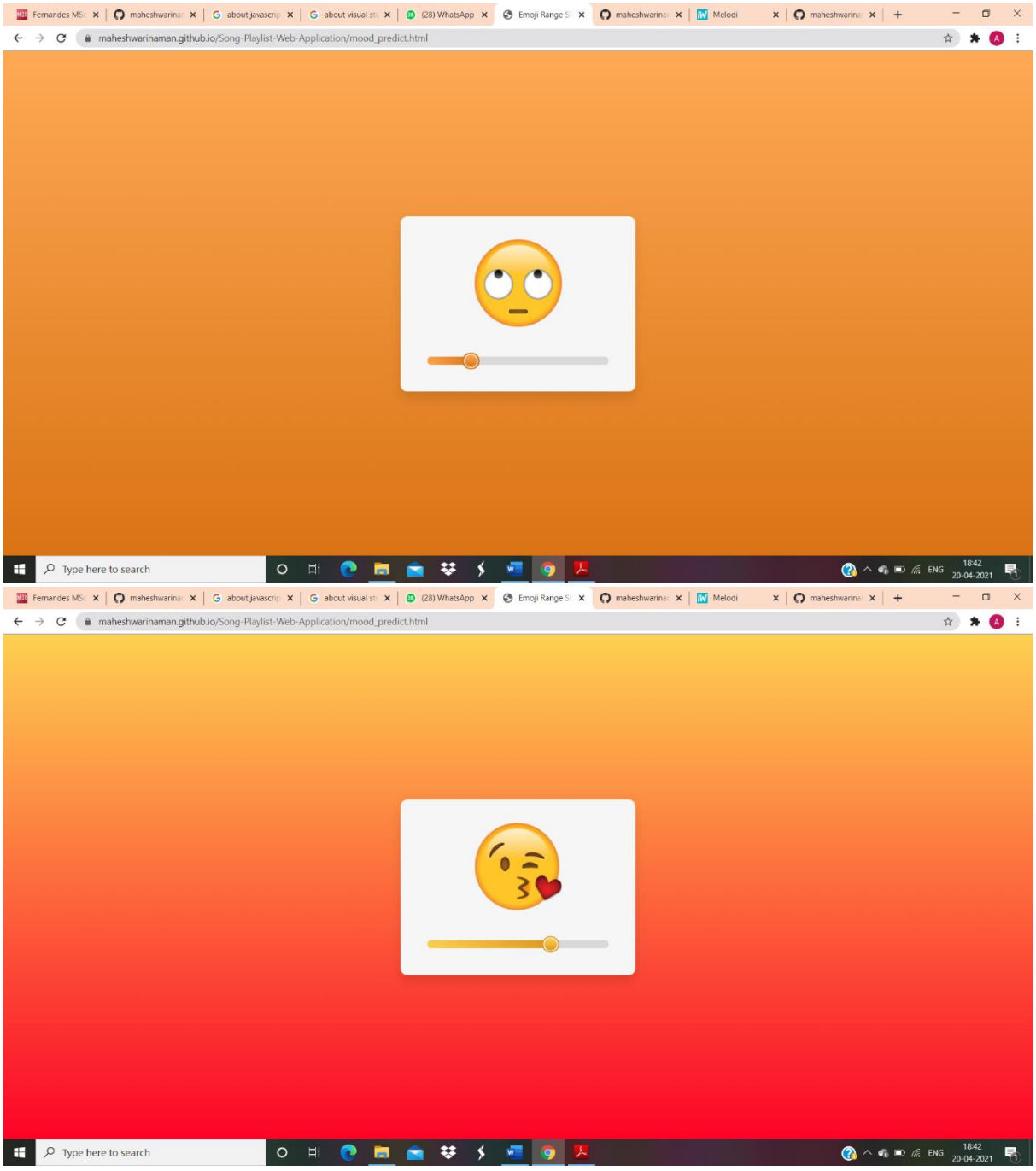

maheshwarinaman.github.io/Song-Playlist-Web-Application/
Melodi
Team
Contact
Music An Art...
Music is an art form and cultural activity whose medium is sound organized in time.
The common elements of music are pitch, rhythm, dynamics and the sonic qualities
of timbre and texture.
Type here to search

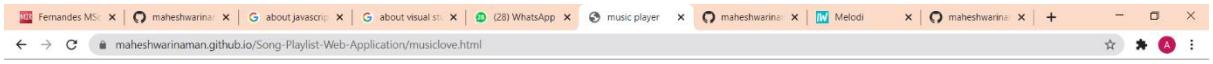

maheshwarinaman.github.io/Song-Playlist-Web-Application/
Melodi
Team
Contact
WHO WE ARE?
What we do?
We are currently in the 3rd year of B.Tech from GLA University. Our main working area is Web Development.Till now, We have made various responsive Web Applications.
Why choose us?
UI is very user friendly.This is a Song playlist web application, so you will observe that our song playlist is selective and limited but is Very rich and Updated.
Type here to search

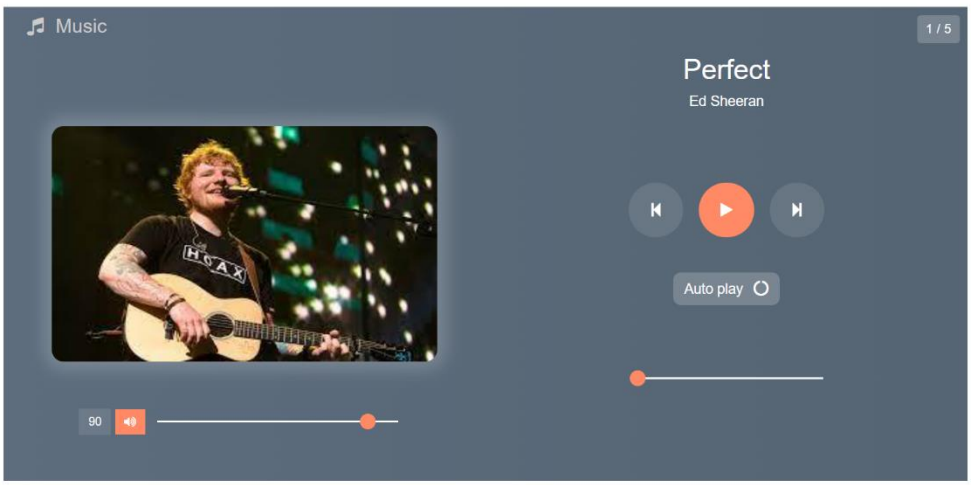

Melodi
Team
Contact
GLA University, Delhi-Agra National Highway, Mathura
naman.maheshwari_cs18@gla.ac.in
7253904551
© copyright 2021, All rights are reserved.
Type here to search
ENG
18:41
20-04-2021

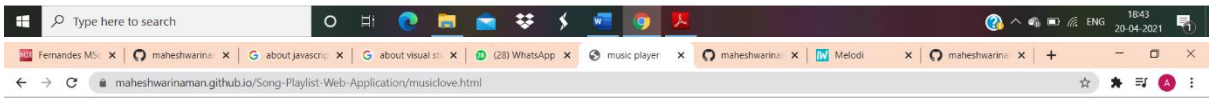

maheshwarinaman.github.io/Song-Playlist-Web-Application/mood_predict.html
Type here to search
ENG
18:42
20-04-2021

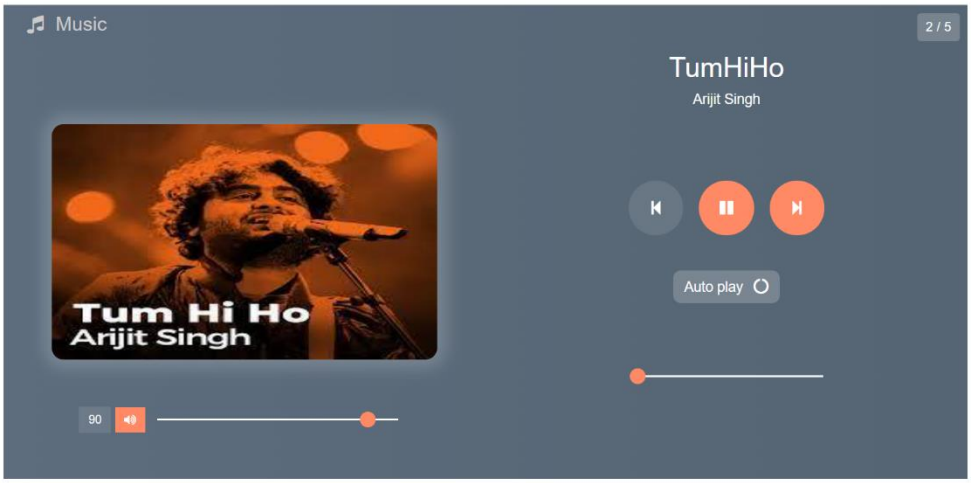

maheshwarinaman.github.io/Song-Playlist-Web-Application/mood_predict.html
Type here to search
ENG
18:42
20-04-2021


maheshwarinaman.github.io/Song-Playlist-Web-Application/mood_predict.html
Type here to search
ENG
18:42
20-04-2021
maheshwarinaman.github.io/Song-Playlist-Web-Application/mood_predict.html
Type here to search
ENG
18:42
20-04-2021


maheshwarinaman.github.io/Song-Playlist-Web-Application/musiclove.html


Music
1 / 5
Perfect
Ed Sheeran
Auto play
90


Type here to search
maheshwarinaman.github.io/Song-Playlist-Web-Application/musiclove.html


Music
2 / 5
TumHiHo
Arijit Singh
Tum Hi Ho
Arijit Singh
Auto play
90


Type here to search

Conclusion

The Mood-Based Music Player is used to automate and give a better music player experience for the end user. This application solves the basic needs of music listeners without troubling them as existing applications do. it uses Full Stack technology to increase the interaction of the system with the user in many ways. It eases the work of the end-user by determining their emotion, and suggesting a customized play-list through a more advanced and interactive system.

References

- https://www.w3schools.com/js/DEFAULT.asp
- https://www.w3schools.com/html/
- https://www.w3schools.com/css/
- https://www.wikipedia.org/
- https://www.google.com/