ONLINE MUSIC PORTAL

A MINI PROJECT REPORT

Submitted by

NIRANJANA.T(510120104015)

SOWMIYA.S(510120104025)

in partial fulfilment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

ADHIPARASAKTHI ENGINEERING COLLEGE
G.B. NAGAR, KALAVAI

ANNA UNIVERSITY: CHENNAI 600 025

JUNE 2023

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report "**ONLINE MUSIC PORTAL**" is the bonafide work of **NIRANJANA.T (510120104015) and SOWMIYA.S(510120104025)** Who carried out the project work under my supervision.

SIGNATURE

Mr. B. SUKKRIVAN, M.Tech.,Ph.D
HEAD OF THE DEPARTMENT

Department of CSE
Adhiparasakthi College of Engineering
G.B. Nagar, Kalavai.

SIGNATURE

Mr. G. JAYACHANDRAN, M.E.,Ph.D
SUPERVISOR

Department of CSE
Adhiparasakthi College of Engineering
G.B. Nagar, Kalavai.

Submitted for the project and viva-voce held on _______________

INTERNAL EXAMINER

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

We sincerely thank to Anna University for providing us with this opportunity to improve our analysis, design and programming skills through this project work. We kindly show our beloved thanks for completion of our project with divine blessing of his Holiness **PADMASHRI ARULTHIRU BANGARU ADIGALAR**, **SAKTHI. THIRUMATHI**. **LAKSHMI BANGARU ADIGALAR**, vice president of **ACMEC** Trust, **SAKTHI**. **THIRUMATHI. Dr.B. UMA DEVI,** Correspondent and **SAKTHI.THIRU**. **R. KARUNANITHI, B.Sc., M.A.,** Secretary of our college. We tender our profound and heartfelt thanks to our respected and beloved principal **Prof. Dr. T. KARTHIKEYAN, M.E., Ph.D(Engg)., MISTE., MIAE.,** for having extended all possible help in the execution of this project. We also express our sincere thanks to

Mr. B. SUKKRIVAN, M.Tech.,Ph.D Head of the Department and project coordinator, Computer Science and Engineering for allowing me to undertake this project. We wish to express my deep sense of gratitude to our guide to **Mr.G.JAYACHANDRAN,M.E.,Ph.D** Assistant Professor, Computer Science and Engineering for his valuable guidance throughout the course of this project. We also wish to express our sincere thanks to the review committee members, all Department Teaching and non-teaching staff members for their valuable suggestions and providing the required infrastructure. Finally, we wish to express our indebtedness to our beloved Parents, the ultimate force behind our grand success.

ABSTRACT

In today's scenario computer is an important part in day-to-day life. Every individual right from school student to a business tycoon utilizes the computer browse the internet to send email, chat, buy products via net etc.

This study is aimed at developing a web based music portal which can be used to manage a musical library and a picture gallery. It is aimed to replace the manual system of getting musical CD's considering the technology advancement. The study discusses the World Wide Web(www) as an Internet service that allows the distribution of pages. Familiarity with web based application; web programming and web development as an industry are also discussed. It also consist the system analysis and design which include the several download and upload mechanisms. The system design, file and database design is based on the detailed of the proposed system. The implementation and maintenance of the system comprises the software development, software testing and debugging as well as software implementation.

A music website is basically a webpage where one can play/pause music. It has other options like the home section, music section, etc. In this project, we are going to make a website that will play/pause music using HTML,CSS and JavaScript. We will use HTML to give a basic layout and with CSS, we will give beautify our design by giving a ground and play pause button image. We will use basic JavaScript feature to play and pause our music.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

The World Wide Web(WWW) has witnessed explosive growth over the past few years after its invention by Tim Berners-Lee in 1980. This is largely due to the new ways information has been created and used on the Internet. Consequently, the World Wide Web has become an important tool which can be adopted for various purposes.

The World Wide Web known as the web, for short is a network of computers that are able to exchange text, graphics and multimedia information via the Internet by sitting at a computer that is attached to the Web, using either a dial-up phone line or a much faster broadband(Ethernet, cable, or DSL connection), one can also visit Web-connected computers next door, at a nearby university, or halfway around the world. And one can take full advantage of the resources these computers make available, including text, graphics, videos, sounds and animation. Think of the Web as the multimedia version of the Internet, and one will be right on the mark.

To understand the concept of the WWW, one needs to consider the concept of the Web as a large collection of the documents. In its relatively short life, the Web has grown to host million of sites and billions of pages. For the moment, think of each of these pages as a document. Many documents on the Web bear a strong similarity to the documents one meets in everyday life, and all documents have a structure of some of the documents seen in everyday life.

MUSIC AND THE INTERNET

The advent of the Internet has transformed the experience of music partly through the increased ease of access to music and the increased choice. Chris Anderson, in his book The Long Tail: Why the Future of Business is Selling Less of More, suggests that while the economic model of supply and demand describes scarcity, the Internet retail model is based on abundance. Digital storage costs are low, so a company can afford to make its whole inventory available online, giving customers as much choice as possible. It has thus become economically viable to offer products that very few people are interested in. customers growing awareness of their increased choice result in a closer association between listening tastes and social identity, and the creation of Thousands of the niche markets.

Another effect of the Internet arises with the online communities like YouTube and MySpace. MySpace has made social networking with other musicians easier, and greatly facilitates the distribution of one’s music. YouTube also has a large community of both amateur and professional musicians who post videos and comments. Professional musicians also use YouTube as a free publisher of promotional material. YouTube users, for example, no longer only download and listen to MP3, but also actively create their own. Accordingly to Don Tapscott and Anthony D. Williams, in their book Wikinomics, there has been a shift from a traditional consumer role to what they call a “prosumer” role, a consumer who both creates and consumes. Music is made available on the internet using a technology call Web Development Application.

OBJECTIVES OF ONLINE MUSIC PORTAL

The main objective pof the Project on Online Music Portal is to manage the details of Performer, Album, AlbumType, Track, Music. It manages all the information about Performer. The project is totally built at administrative end and thus only the administrator is guaranteed the access. The purpose of the project id to build an application program progress to reduce the manual work for managing the Performer, Album, User, Album Type. It tracks all the details about the Album Type, Track, Music.

FUNCTIONALITIES

The functionalities provided by Online Music Portal are as follows:

- Provides the searching facilities based on various factors. Such as Performer, Ablum Type, Track, Music.
- Online Music Portal also manage the user details online for Track details, Music details, Performer.
- It tracks all the information and transactionns of Tracks,etc.
- Manage the information of Album.
- Shows the information and description of the Performer,Album Type.
- To increase efficiency of managing gthe Performer,Album.
- It deals with monitering the information and transactions of Track.
- Manage the information of Performer

- Editing, adding and updating of Records is improved which results in proper resource management of Performer data.
- Manage the information of Track.

SCOPE OF ONLINE MUSIC PORTAL

It may help collecting perfect management in details. In a very short time, the collection will be obivious, simple and sensible. It will help a person to know the management of passes year perfectly and vivdly. It also helps in current all works relative to Online Music Portal. It will be aslo reduced the cost of collecting the management & collection procedure will go on smoothly.

Our project aims at Business process automation , i.e. we have tried to computerize various processess of Online Music Portal.

- In computer system the person has to fill various forms & number of copies of the forms can be easily generated at a time.
- In computer system, it is not necessary to create the manifest but we can directly print it, which saves our time.
- To assist the staff in capturing the effort spent on their respective working areas.
- To utilize resources in an efficient manner by increasing their productivity through automation.
- The system generates types of information that can be used for various purposes.
- It satisfy the user requirement.

- Be easy to understand by the user and operator.
- Be easy to operate.

MODULES OF ONLINE MUSIC PORTAL

- Performer Management Module: Used for managing the Performance details.
- Music Module: Used for managing the details of music.
- User Module: Used for managing the details of User.
- Album Management Module: Used for managing the information and details of the Album.
- Album Type Module: Used for managing the Album Type details.
- Track Module: Used for managing the Track informations.
- Login Module: Used for managing the login details.
- Users Module: Used for managing the users of the system.

INPUT DATA AND VALIDATION OF PROJECT ON ONLINE MUSIC PORTAL

- All the fields such as Performer, Ablum Type, Music are validated and does not take invalid values.
- Each form for performer, Album, User can not accept blank values fields.
- Avoiding error in data.
- Controlling amount of input.
- Integration of all the modules/forms in the system.
- Preparation to the test cases.
- Preparation of the possible test data with all the validation checks.

- Actual testing done manually.
- Recording of all the reproduced errors.
- Modification done for the errors found during testing.
- Prepared the test results scripts after rectification of the errors.

FEATURES OF THE ONLINE MUSIC PORTAL

- Product and component based
- Creating & changing issues at ease
- Query issue list to any depth
- Reporting &charting in more comprehensive way
- User accounts to control the access and maintain security
- Simple status & resolutions
- Multi-level priorities & severities.
- Targets & milestones fpr guiding the programmers
- Attachment &additional comments for more information
- Robust database back-end
- Various level of reports available with lot of filter criteria's
- It contain better storage capacity
- Accuracy in work
- Easy & fast retrieval of information
- Well designed reports
- Decrease the load of the person involve in existing manual system
- Access of any information individually
- Easy to update information

CHAPTER 2

SOFTWARE REQUIREMENT SPECIFICATION

The software requirements specification is produced at the culmination of the analysis task. The function and performance allocated to software as part of system engineering are refined by establishing a complete information description, a detailed functional and behavioral description, an indication of performance requirements and design constraints, appropriate validation criteria, and other data pertinent to requirements.

The proposed system has the following requirements

- System needs to store information about new entry of performer.
- System needs to help the internal staff to keep information of Album and find them as oer various queris.
- System need to maintain quantity record.
- System need to keep the record of Album Type.
- System need to update and delete the record
- System also needs a search area.
- It also needs a security system to prevent data.

Identification of need

The old manual system was suffering from a series of drawbacks. Since whole of the system was to be maintained with hands the process of keeping, maintaining and retrieving the information was very tedious and lengthy. The records were never used to be in a systematic order there used to be a lots of difficulties in associating any particular transaction with a particular context. If any information was to be found it was required to go through the different registers, documents there could never exist anything like report generation. There would always be unnecessary consumption of time while entering records and retrieving records.

The reason behind it is that there is lot of information to be maintained and have to be kept in mind while running the business. For this reason we have provided features present system is partially automated, actually existing system is quite laborious as one has to enter same information at three different places.

Following points should be well considered

- Documents and reports that must be provided by the new system: there can also be few reports , which can help management in decision-making and cost controlling, but since these reports do not required attention, such kind of reports and information were also identifies and given required attention, such kind of reports and information were also identifies and given required attention.
- The required frequency and distribution for each document.
- With the implementation of computerized system, the task of keeping records in an organized manner will be solved. The greatest of all is the retrival of information in saving the time in different operations and making information flow easy giving valuable reports.

Feasibility Study:

After doing the project Online Music Portal, study and analyzing all the existing or required functionalities of the system, the next task is to do the feasibility study for the project. All projects are feasible – given unlimited resources and infinite time.

Feasibility study includes consideration of all the possible ways to provide a solution to the given problem. The proposed solution should satisfy all the user requirements and should be flexible enough so that future changes can be easily done based on the future upcoming requirements.

A. Economical Feasibility

This is very important aspect to be considered while developing a project. We decided the technology based on minimum possible cost factor.

- All hardware and software can has to be borne by the organization
- Overall we have estimated that the benefits the organization is going to receive from the proposed system will surely overcome the initial costs and the later on running cost for system.

B. Technical Feasibility

This included the study of fun ction, performance and constraints that may affect the ability to achieve an acceptable system. For this feasibility study, we studied complete functionality to be provided in the syustem, as described in the system requirement system(SRS), and checked if everything was possible using different type of fronted and backend platformst.

C. OperationalFeasibility

No doubt the proposed system is fully GUI based that is very user friendly and all inputs to be taken all self-explanatory even to a layman. Besides, a proper training has been conducted to let know the essence of the system to the users so that they feel comfortable with now system. As for our study is concerned the clients are comfortable and happy as the system has cut down their loads and doing.

CHAPTER 3

SYSTEM DESIGN OF ONLINE MUSIC PORTAL

In this phase, a logical system is built which fulfils the given requirements. Design phase of software development deals with transforming the client's requirements into a logically working system. Normally, design is performed in the following in the following two steps:

Primary Design Phase

In this phase, the system is designed at block level. The blocks are created on the basis of analysis done in the problem identification phase. Different blocks are created for different functions emphasis is put on minimizing the information flow between blocks. Thus, all activities which require more interaction are kept in one block.

Secondary Design Phase

In the second design phase the detailed design of every block is performed.

The general tasks involved in the design process are the following

1. Design various blocks for overall system processes.
2. Design smaller, compact and workable modules in each block.
3. Design various database structures.
4. Specify details of programs to achieve desired functionality.
5. Design the form of inputs, and outputs of the system.
6. Perform documentation of the design.
7. System reviews.

USER INTERFACE DESIGN

User Interface Design is concerned with the dialogue between a user and the computer. It is concerned with everything from starting the system or logging into the system to the eventually presentation of desired inputs and outputs. The overflow of screens and messages is called a dialogue.

<u>The following steps are various guidelines for User Interface Design:</u>

- The system user should always be aware of what to do next
- The screen should be formatted so that various types of information, instructions and messages always appear in the same general display area.
- Message, instruction or information should be displayed long enough to allow the system user to read them.
- Use display attributes sparingly
- Default values for fields and answers to be entered by the user should be specified.
- A user should not be allowed to proceed without correcting an error
- The system user should never get an operating system message or fatal error.

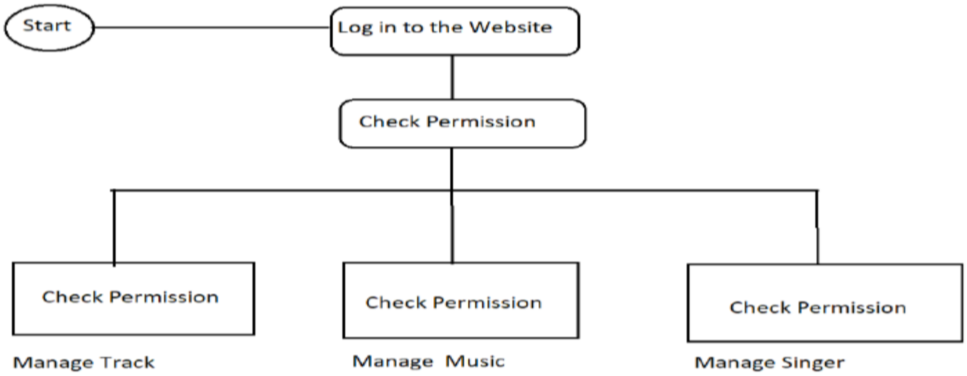
Approach

The analysis of the currently available tools to accomplish the task is one of the most important steps because the ground concepts of the application should never change, regardless of its future complexity. The several possible implementations of the web service together with the balancing of computing tasks between server and client are the first parameters that have to be defined for a solid base. Also the programming language plays a crucial part in the development process, as it is shown later. The amount of callbacks to the database in favor of less memory usage is also an important aspect that is difficult to estimate from the start. In order to allow a high flexibility while still maintaining a small dataflow, the implementation of the logic is mainly on the server. The UI responsibility is fully retained by the client side as well as servicing UI requests and only notify the server of such activity. In order to achieve the high goals that were set, the structure of the application is important to be highly modularized to allow interchanging the modules with better, more complex implementations.

System Analysis

After carefully analyzing the requirements and functionality of the web application, I had one important diagrams by the end of the analysis phase. This is the ER diagram and data flow diagram which were the basis for finding out entities and relationships between them, the flow of information.

Following are the Diagrams –



Languages Used

The languages used to create the online music portal are:

- HTML5
- CSS3.
- JavaScript.

.

CHAPTER 4

HTML

An HTML document is a file containing hypertext markup language. HTML code is based on tags, or hidden keywords, which provide instructions for formatting the document. A tag starts with an angle bracket and the 'less than' sign: '<'. The tag ends with an angle bracket and the 'greater than' sign '>'. Tags tell the processing program, often the web browser, what to do with the text. For example, to make the word 'Hello' bold, you would use the opening bold tag <b> and then the closing bold tag </b>, like this:

```
<b>Hello</b>
```

HTML is defined by the World Wide Web Consortium, an organization that regulates standards for the Internet. Each version of HTML has a set of definitions. Note that HTML is not a programming language. While we often refer to HTML markup as HTML code, programming languages require the processing of logical statements and math. HTML allows the developer to make text documents look engaging and pleasant. In most cases, programming on an HTML document is done with JavaScript. Types of HTML- The newest version of HTML, which is entering the industry, is HTML 5. HTML can be divided into three categories: transitional, strict, and frameset. These types apply to how HTML is used, not necessarily to the selection of tags.

Transitional- Transitional is the most common type of HTML. It has a flexible syntax, or grammar and spelling component. Over the years, transitional HTML has been used without syntax restrictions, and browsers support a 'best effort' approach to reading the tags. If tags are misspelled, the browsers do not correct web developers' errors, and they display the content anyway. Browsers do not report HTML errors - they simply display what they can. This is the 'best effort' concept.

Strict- The strict type of HTML is meant to return rules into HTML and make it more reliable. For example, the strict type requires closing all tags for all opened tags. This style of HTML is important on phones, where the processing power may be limited. A clean and error-free code helps to load pages faster.

Frame set- Finally, a frameset allows web developers to create a mosaic of HTML documents where multiple documents can be connected into a single screen. This technique is often used to create a menu system. You click on a menu item on the left side of the screen, and only the right side of the screen re-loads. The menu stays in place.

HTML Examples- Let's take a look at a few examples. To execute the examples, simply create a text document on your desktop called 'test.html'. If you are using a Notepad program, please make sure the extension on the file is .html, not 'test.html.txt'. To do this, use 'Save As' in Notepad and then include in a double-quote the name of the file 'test.html' and then just click on that file it will automatically open in the default browser.

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, colours, 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; and enable 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 reader), 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.

Syntax- CSS has a simple syntax and uses a number of English keywords to specify the names of various style properties.

A style sheet consists of a list of rules. Each rule or rule-set consists of one or more selectors, and a declaration block.

Selector:

In CSS, selectors declare which part of the markup a style applies to by matching tags and attributes in the markup itself.

Selectors may apply to the following:- □

- All elements of a specific type, e.g. the second-level headers h2.

- Elements specified by attribute, in particular: o id: an identifier unique within the document, identified with a hash prefix e.g. #id o class: an identifier that can annotate multiple elements in a document, identified with a period prefix e.g. .classname
- Elements depending on how they are placed relative to others in the document tree..

Classes and IDs are case-sensitive, start with letters, and can include alphanumeric characters, hyphens, and underscores. A class may apply to any number of instances of any elements. An ID may only be applied to a single element.

Pseudo-classes are used in CSS selectors to permit formatting based on information that is not contained in the document tree. One example of a widely used pseudo-class is :hover, which identifies content only when the user "points to" the visible element, usually by holding the mouse cursor over it. It is appended to a selector as in a:hover or #elementid:hover. A pseudo-class classifies document elements, such as :link or :visited, whereas a pseudo-element makes a selection that may consist of partial elements, such as ::first-line or ::first-letter.

Selectors may be combined in many ways to achieve great specificity and flexibility.[7] Multiple selectors may be joined in a spaced list to specify elements by location, element type, id, class, or any combination thereof. The order of the selectors is important. For example, div .myClass {color: red;} applies to all elements of class myClass that are inside div elements, whereas .myClass div {color: red;} applies to all div elements that are inside elements of class myClass.

This is not to be confused with concatenated identifiers such as div.myClass {color: red;} which applies to div elements of class myClass.

The following table provides a summary of selector syntax indicating usage and the version of CSS that introduced it.

JavaScript

JavaScript often abbreviated JS, is a programming language that is one of the core technologies of the World Wide Web, alongside HTML and CSS Over 97% of websites use JavaScript on the 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 users' devices.

JavaScript is a high-level, often just-in-time compiled language that conforms to the ECMAScript standard. It has dynamic typing, prototype-based object-orientation, and first-class functions. It is multi-paradigm, supporting 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 Model (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 are now core components of some servers and a variety of applications. The most popular runtime system for this usage is Node.js.

Although Java and JavaScript are similar in name, syntax, and respective standard libraries, the two languages are distinct and differ greatly in design.

Object-orientation (prototype-based)

Prototypal inheritance in JavaScript is described by Douglas Crockford as:

You make prototype objects, and then ... make new instances. Objects are mutable in JavaScript, so we can augment the new instances, giving them new fields and methods.

These can then act as prototypes for even newer objects. We don't need classes to make lots of similar objects... Objects inherit from objects. What could be more object oriented than that?

In JavaScript, an object is an associative array, augmented with a prototype (see below); each key provides the name for an object property, and there are two syntactical ways to specify such a name: dot notation (obj.x = 10) and bracket notation (obj['x'] = 10).

A property may be added, rebound, or deleted at run-time. Most properties of an object (and any property that belongs to an object's prototype inheritance chain) can be enumerated using a for...in loop.

Security

JavaScript and the DOM provide the potential for malicious authors to deliver scripts to run on a client computer via the Web. Browser authors minimize this risk using two restrictions. First, scripts run in a sandbox in which they can only perform Web-related actions, not general-purpose programming tasks like creating files. Second, scripts are constrained by the same-origin policy: scripts from one Web site do not have access to information such as usernames, passwords, or cookies sent to another site. Most JavaScript-related security bugs are breaches of either the same origin policy or the sandbox.

There are subsets of general JavaScript—ADsafe, Secure ECMAScript (SES)—that provide greater levels of security, especially on code created by third parties (such as advertisements). Closure Toolkit is another project for safe embedding and isolation of third-party JavaScript and HTML.

Content Security Policy is the main intended method of ensuring that only trusted code is executed on a Web page.

Proposed Model

- The application is a simple HTML file that you open in your browser
- You only need to download our zip file from the button near the beginning of the article, and unzip it somewhere on your computer.
- Unfortunately, due to security restrictions in modern browsers it won't work if you just double click the index.html file.
- You will have to open it through a locally running web server like Apache or Nginx and access it through local host. Or you can just use our demo, nothing is uploaded so your music is safe.
- The app listens for JavaScript drag and drop events.
- When you drop a mp3 file, it extracts information like song and artist name, if they are available, from the file's ID3 tags.
- Each song is placed in an array, which represents our playlist.
- The application then initializes the Wavesurfer.js audio player, which generates the awesome wave visualization for every song and plays it.
- From there on we can do everything you would expect from a native audio player - play next/previous, pause, pick songs and so on.
- Our playlist section also gives users the option to remove songs from the player or search for a particular track, album or artist

Motivation

Several solutions already use intelligent playlists embedded in music players installed on computers. There are also online solutions, the most popular of which islast.fm, which acts as a personalized radio station that plays preferred music.

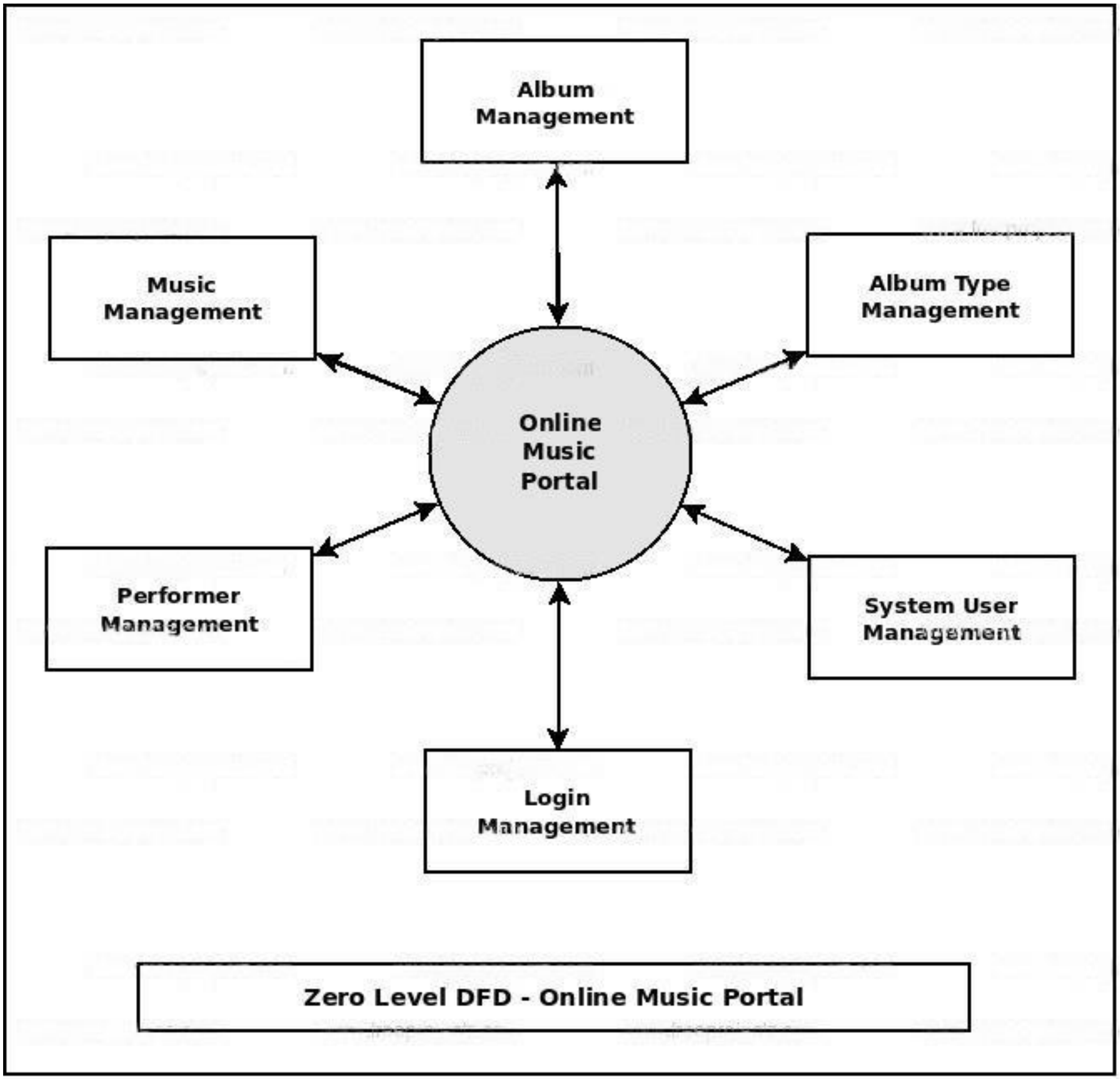
On the other hand it does not allow playback of a certain track. There are also other solutions, like the genius function of iTunes or the Music Explorer; both use the user's music collection to generate playlists. The biggest disadvantage of the latter solution is that the user can use only tracks that he/she already has on his/her PC to generate playlists.

Of course this limits the power or the algorithm very much. There are already services that provide the music content (like last.fm or YouTube to name a few) so it's a natural conclusion to try to use these service s in connection with the playlist-generating algorithm.

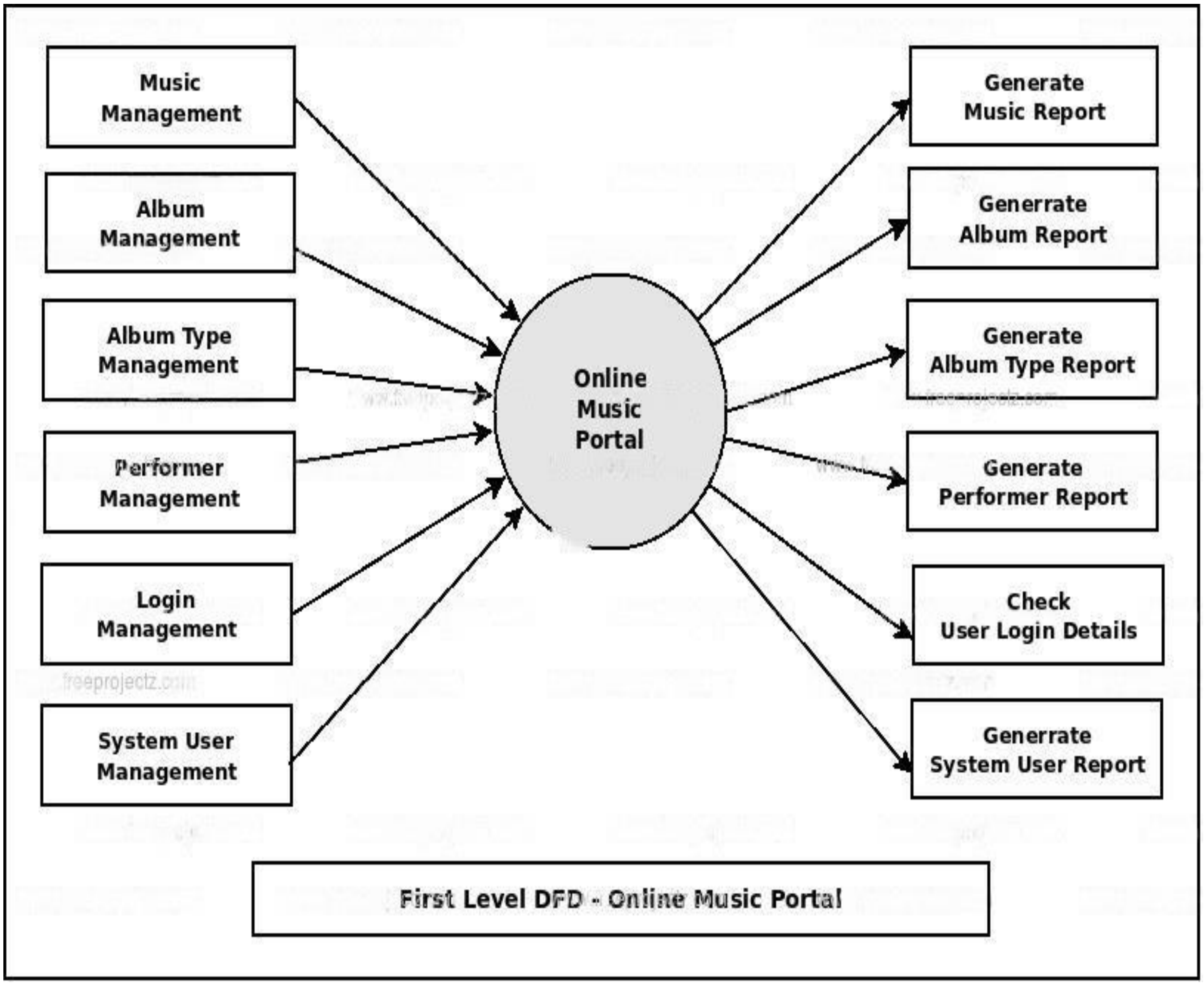
CHAPTER 5

DATA FLOW DIAGRAM FOR ONLINE MUSIC PORTAL

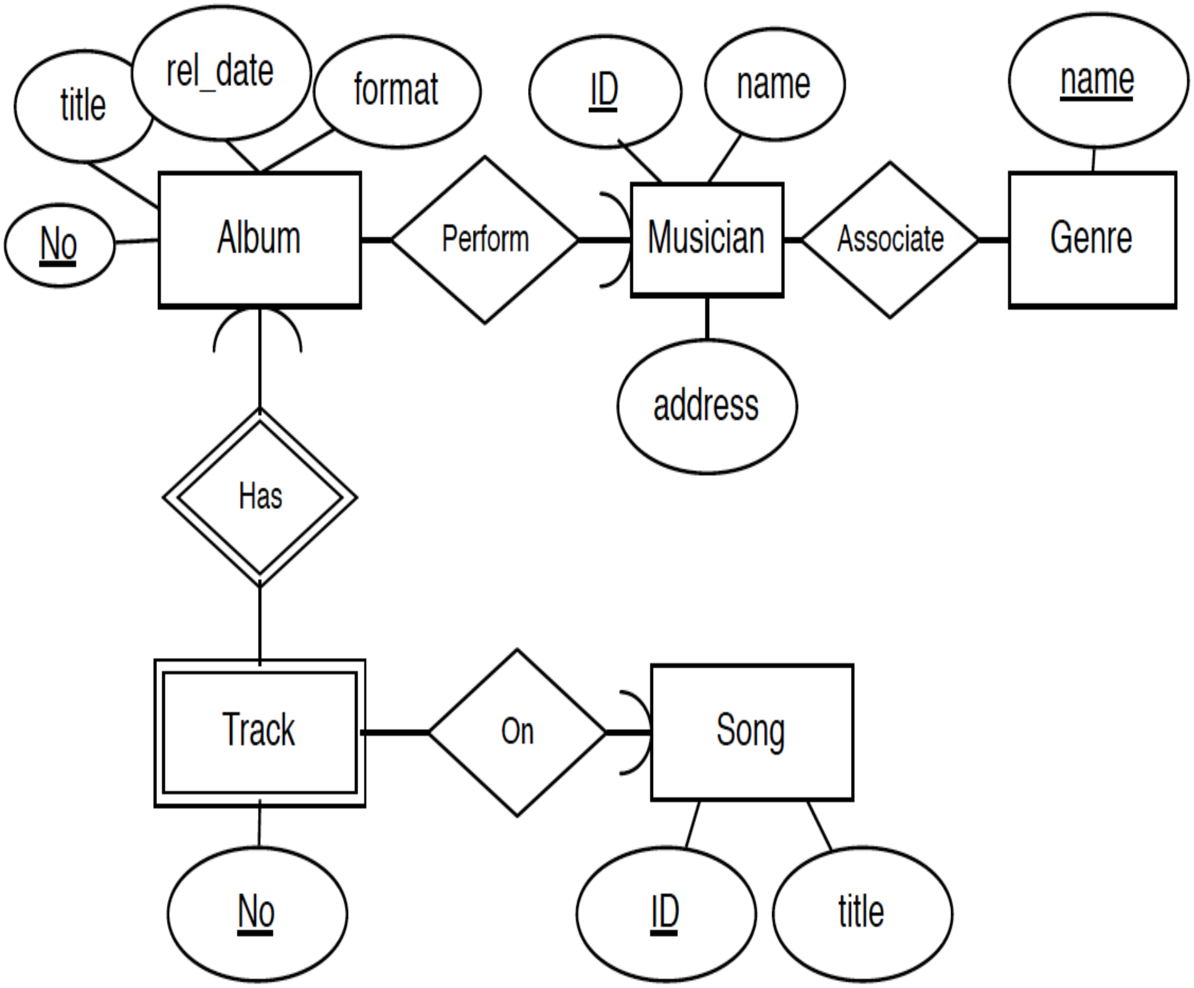
LEVEL ZERO DFD



LEVEL ONE DFD



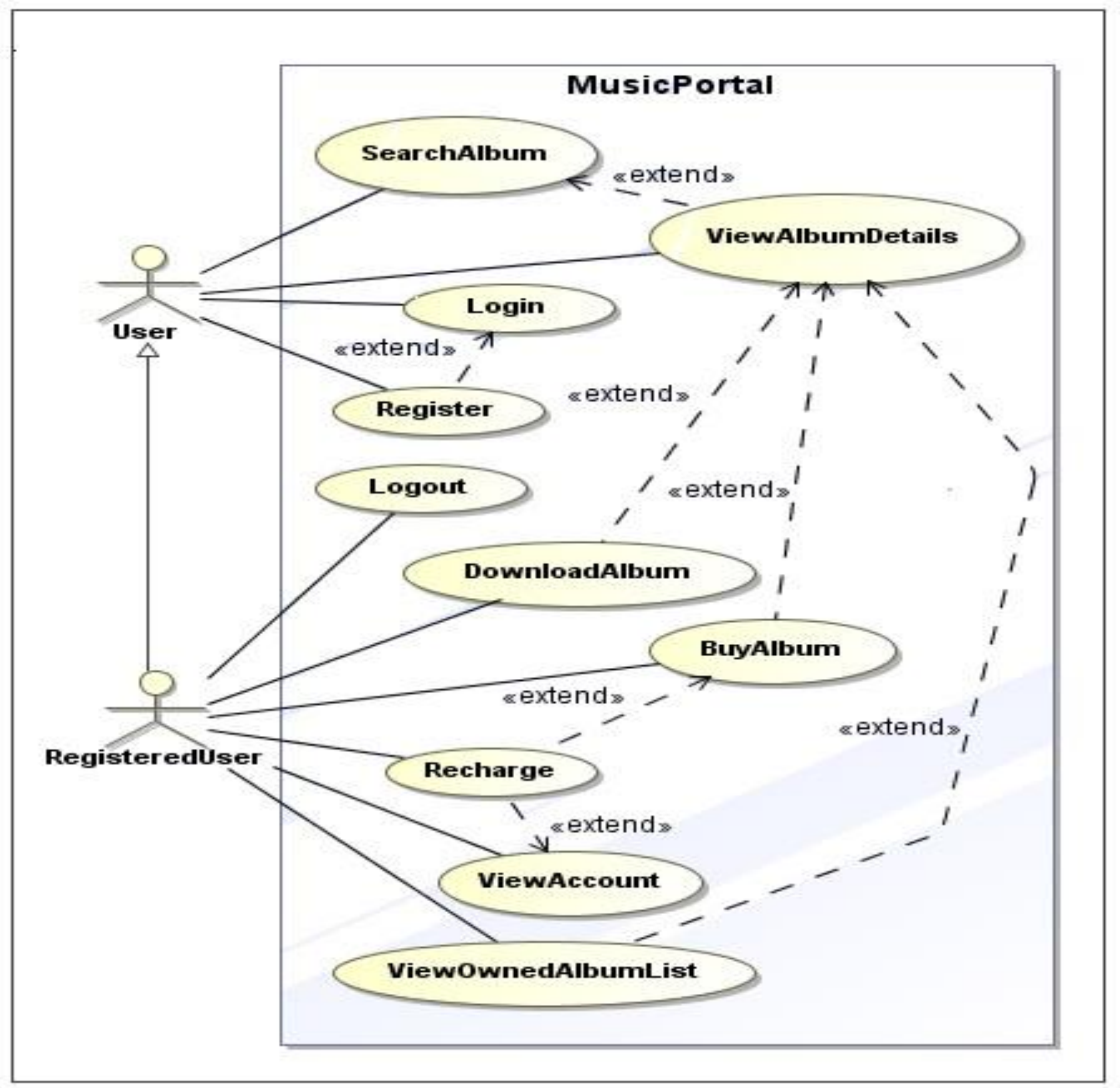
ENTITY-RELATION DIAGRAM



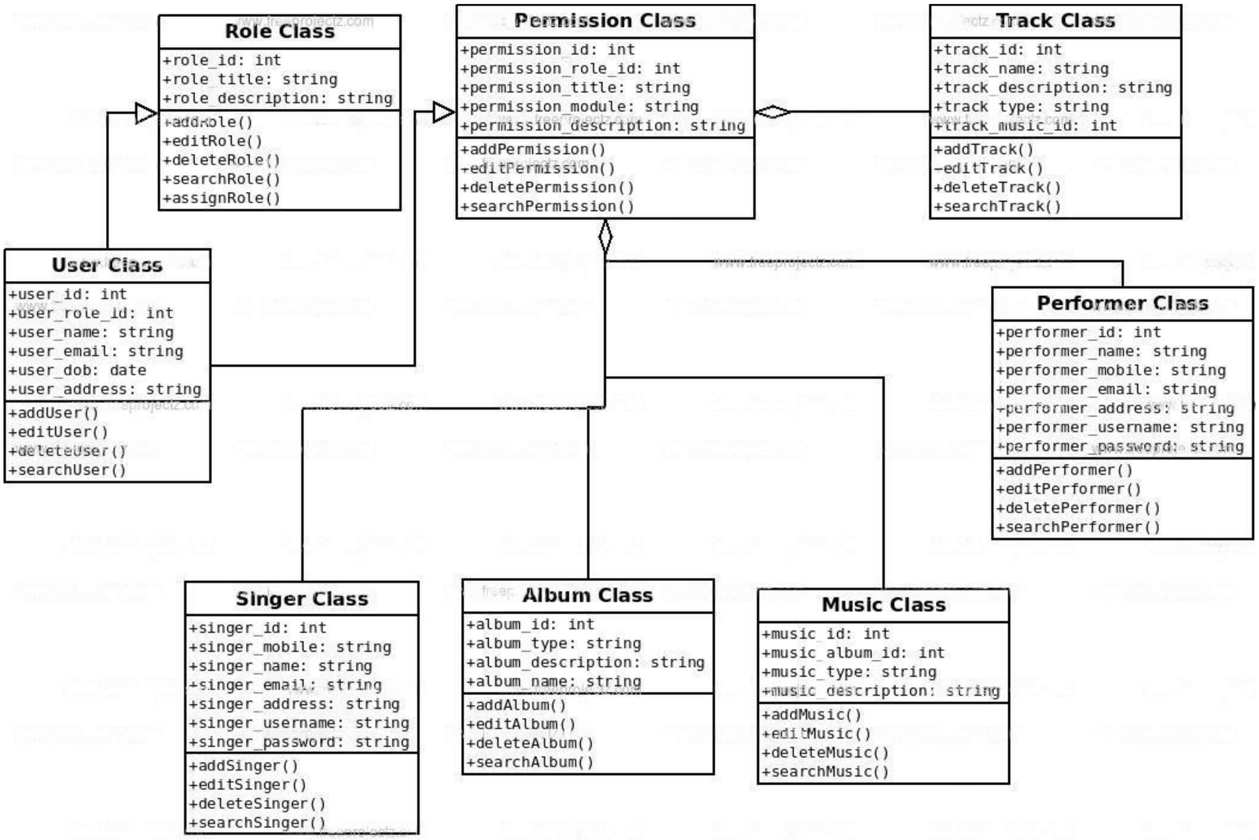
CHAPTER 6

UML DIAGRAMS FOR ONLINE MUSIC PORTAL

Usecase Diagram

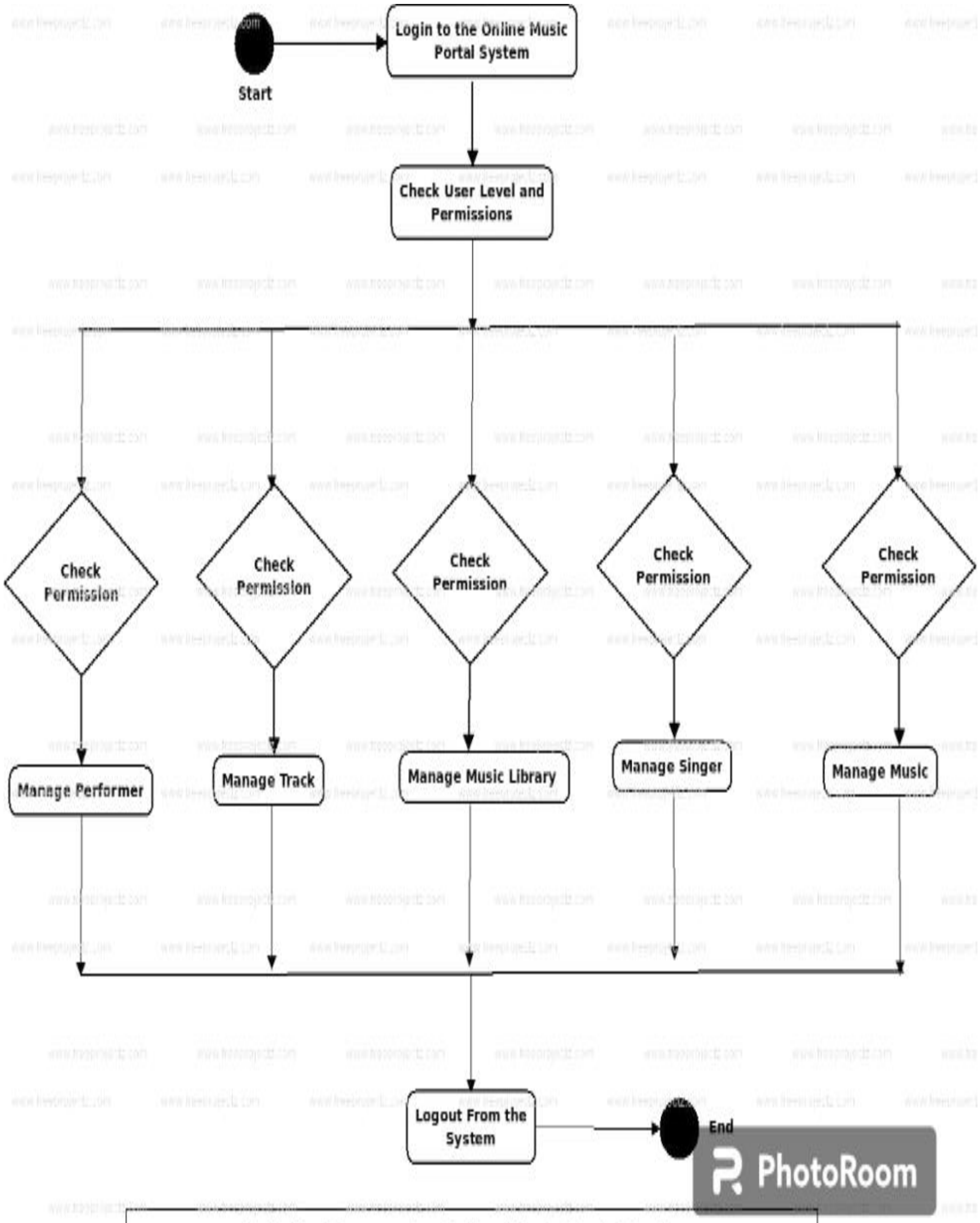


Class diagram



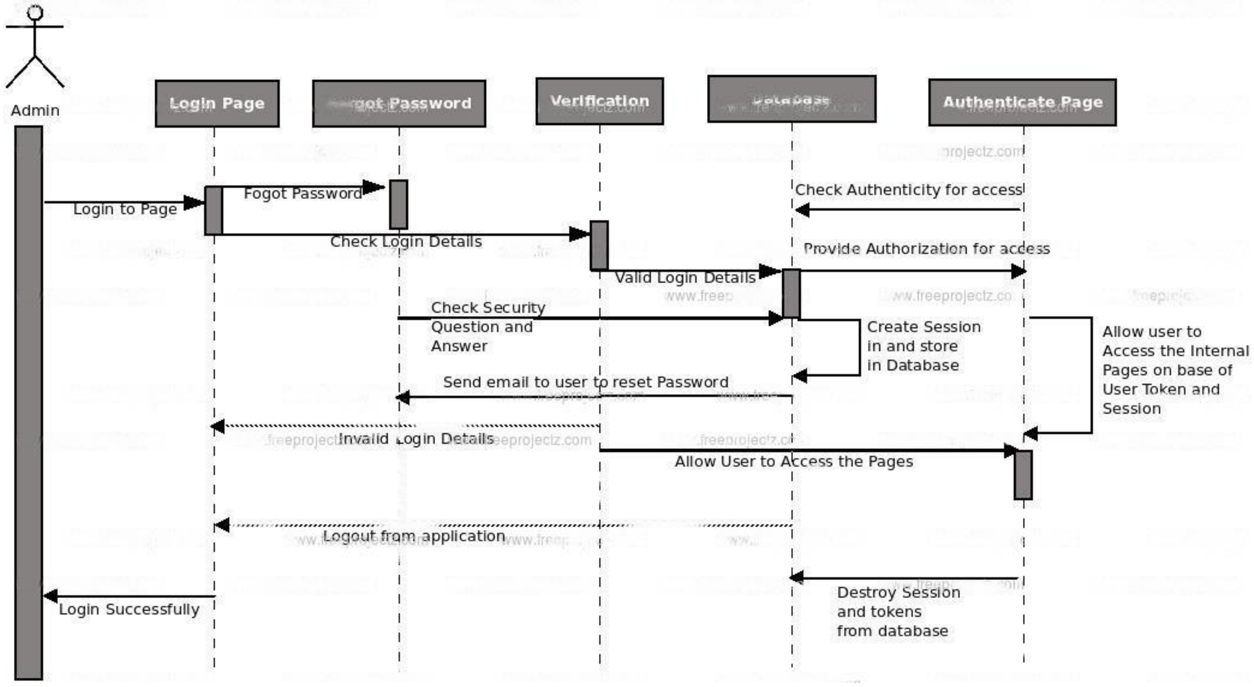
Class Diagram of Online Music Portal System

Activity diagram



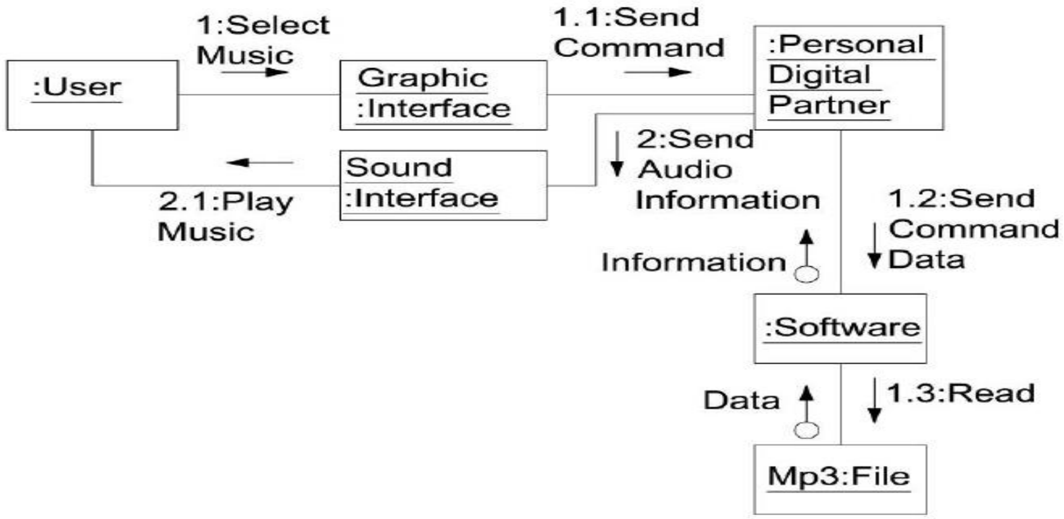
Activity Diagram for Online Music Portal System

Sequence diagram

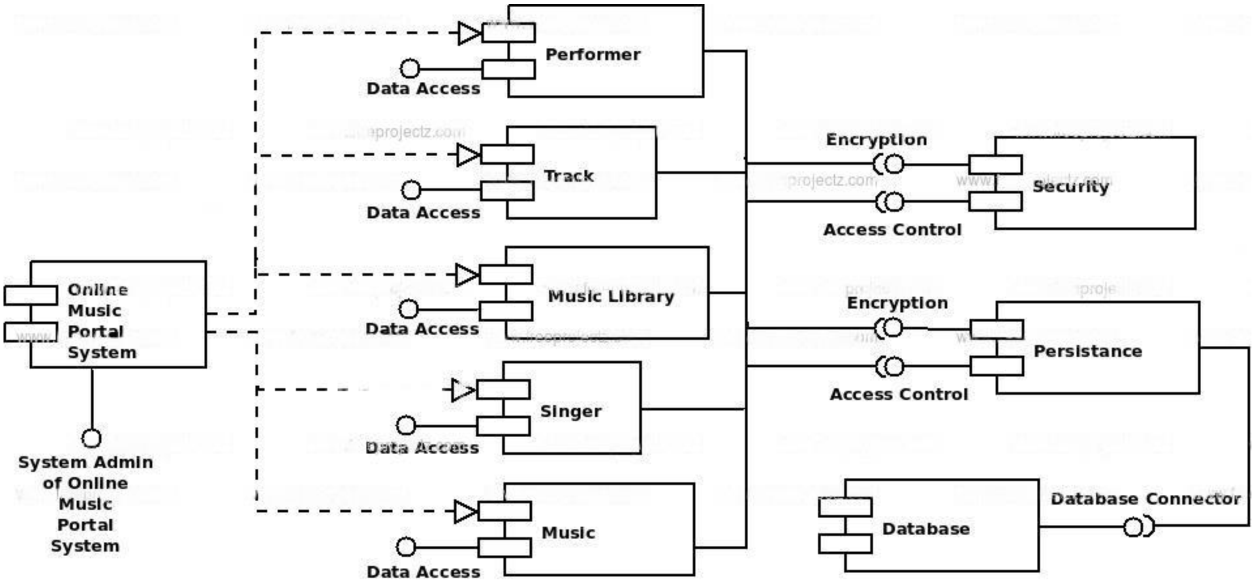


Admin
Login Page
got Password
Verification
Database
Authenticate Page
Login to Page
Fogot Password
Check Login Details
Valid Login Details
Check Authenticity for access
Provide Authorization for access
Check Security Question and Answer
Create Session in and store in Database
Allow user to Access the Internal Pages on base of User Token and Session
Send email to user to reset Password
Invalid Login Details
Allow User to Access the Pages
Logout from application
Destroy Session and tokens from database
Login Successfully

Collaboration diagram



Component diagram



Component Diagram of Online Music Portal System

CHAPTER 7
ONLINE MUSIC PORTAL PROGRAM CODE

MAIN PAGE

HTML

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.7.2/font/bootstrap-icons.css">
    <link rel="stylesheet" href="style.css">
    <title>Music On</title>
</head>

<body>
<header>
    <div class="menu_side">
        <h1>MUSIC</h1>
        <div class="playlist">
            <h4 class="active"><span></span><i class="bi bi-music-note-beamed"></i> Playlist</h4>
            <a href="http://127.0.0.1:5500/pre.html"><h4 ><span></span><i class="bi
bi-music-note-beamed"></i>Last Listening</h4></a>
            <h4 ><span></span><i class="bi bi-music-note-beamed"></i>
Recommended</h4>
        </div>
        <div class="menu_song">
            <li class="songItem">
                <span>01</span>
                <img src="img/1.jpg" alt="Alan">
                <h5>
                    Alone
                    <div class="subtitle">Alan Walker</div>
                </h5>
                    <i class="bi playListPlay bi-play-circle-fill" id="1"></i>
            </li>
            <li class="songItem">
                <span>02</span>
                <img src="img/2.jpg" alt="Alan">
                <h5>
                    On My Way
                    <div class="subtitle">Alan Walker</div>
                </h5>
                    <i class="bi playListPlay bi-play-circle-fill" id="2"></i>
            </li>
            <li class="songItem">
                <span>03</span>
                <img src="img/3.jpg" alt="Alan">
```

```
    <h5>
        LILY
        <div class="subtitle">Alan Walker</div>
    </h5>
        <i class="bi playListPlay bi-play-circle-fill" id="3"></i>
</li>
<li class="songItem">
    <span>04</span>
    <img src="img/4.jpg" alt="Alan">
    <h5>
        On My Way
        <div class="subtitle">Alan Walker</div>
    </h5>
        <i class="bi playListPlay bi-play-circle-fill" id="4"></i>
</li>
<li class="songItem">
    <span>05</span>
    <img src="img/5.jpg" alt="Alan">
    <h5>
        My stupid heart
        <div class="subtitle">eratugal</div>
    </h5>
        <i class="bi playListPlay bi-play-circle-fill" id="5"></i>
</li>
<li class="songItem">
    <span>06</span>
    <img src="img/1.jpg" alt="Alan">
```

```
            <h5>
                You
                <div class="subtitle">Armaan malik</div>
            </h5>
                <i class="bi playListPlay bi-play-circle-fill" id="6"></i>
        </li>
    </div>
</div>


<div class="song_side">
    <nav>
        <ul>
            <li>Discover <span></span></li>
           <a href="lib.html"> <li>MY LIBRARY</li></a>
           <a href="com.html"><li>COMMENT</li></a>
        </ul>
        <div class="search">
            <i class="bi bi-search"></i>
            <input type="text" placeholder="Search Music...">

        </div>
        <div class="user">
            <img src="img/00.jpg"title="Towmi">
        </div>
    </nav>
    <div class="content">
    <h1>MUSIC WORLD</h1>
    <p>
        “Music gives a soul to the universe, wings to the mind,
        <br>
        flight to the imagination, and life to everything.”
        <br>
        – Plato
    </p>
    <div class="buttons">
        <button>PLAY</button>
        <button>FOLLOW</button>
    </div>
</div>

<div class="popular_song">
    <div class="h4">
        <h4>Popular Song</h4>
        <div class="btn_s">
            <i id="left_scroll" class="bi bi-arrow-left-short"></i>
            <i id="right_scroll" class="bi bi-arrow-right-short"></i>
        </div>
    </div>
    <div class="pop_song">
        <li class="songItem">
            <div class="img_play">
                <img src="img/1.jpg" alt="alan">
                <i class="bi playListPlay bi-play-circle-fill" id="7"></i>
    </div>
    <h5>On My Way
        <br>
        <div class="subtitle">Alan Walker</div>
    </h5>
</li>
<li class="songItem">
    <div class="img_play">
        <img src="img/1.jpg" alt="alan">
        <i class="bi playListPlay bi-play-circle-fill" id="8"></i>
        <!-- change All id  -->
    </div>
    <h5>On My Way
        <br>
        <div class="subtitle">Alan Walker</div>
    </h5>
</li>
<li class="songItem">
    <div class="img_play">
        <img src="img/1.jpg" alt="alan">
        <i class="bi playListPlay bi-play-circle-fill" id="9"></i>
    </div>
    <h5>On My Way
        <br>
        <div class="subtitle">Alan Walker</div>
    </h5>
</li>
```

```
 <li class="songItem">
     <div class="img_play">
         <img src="img/1.jpg" alt="alan">
         <i class="bi playListPlay bi-play-circle-fill" id="10"></i>
     </div>
<li class="songItem">
     <div class="img_play">
         <img src="img/1.jpg" alt="alan">
         <i class="bi playListPlay bi-play-circle-fill" id="14"></i>
     </div>
     <h5>On My Way
         <br>
         <div class="subtitle">Alan Walker</div>
     </h5>
 </li>
 <li class="songItem">
     <div class="img_play">
         <img src="img/1.jpg" alt="alan">
         <i class="bi playListPlay bi-play-circle-fill" id="15"></i>
     </div>
     <h5>On My Way
         <br>
         <div class="subtitle">Alan Walker</div>
     </h5>
 </li>
 <li class="songItem">
     <div class="img_play">
```

```
        <img src="img/1.jpg" alt="alan">
        <i class="bi playListPlay bi-play-circle-fill" id="16"></i>
    </div>
    <h5>On My Way
        <br>
        <div class="subtitle">Alan Walker</div>
    </h5>
</li>
<li class="songItem">
    <div class="img_play">
        <img src="img/1.jpg" alt="alan">
        <i class="bi playListPlay bi-play-circle-fill" id="17"></i>
    </div>
    <h5>On My Way
        <br>
        <div class="subtitle">Alan Walker</div>
    </h5>
</li>
<li class="songItem">
    <div class="img_play">
        <img src="img/1.jpg" alt="alan">
        <i class="bi playListPlay bi-play-circle-fill" id="18"></i>
    </div>
    <h5>On My Way
        <br>
        <div class="subtitle">Alan Walker</div>
    </h5>
```

```
</li>
<li class="songItem">
    <div class="img_play">
        <img src="img/1.jpg" alt="alan">
        <i class="bi playListPlay bi-play-circle-fill" id="19"></i>
    </div>
    <h5>On My Way
        <br>
        <div class="subtitle">Alan Walker</div>
    </h5>
</li>
<li class="songItem">
    <div class="img_play">
        <img src="img/1.jpg" alt="alan">
        <i class="bi playListPlay bi-play-circle-fill" id="20"></i>
    </div>
    <h5>On My Way
        <br>
        <div class="subtitle">Alan Walker</div>
    </h5>
</li>
<li class="songItem">
    <div class="img_play">
        <img src="img/1.jpg" alt="alan">
        <i class="bi playListPlay bi-play-circle-fill" id="21"></i>
    </div>
    <h5>On My Way
```

```
        <br>
        <div class="subtitle">Alan Walker</div>
      </h5>
    </li>
    <li class="songItem">
      <div class="img_play">
        <img src="img/1.jpg" alt="alan">
        <i class="bi playListPlay bi-play-circle-fill" id="24"></i>
      </div>
      <h5>On My Way
        <br>
        <div class="subtitle">Alan Walker</div>
      </h5>
    </li>
    <li class="songItem">
      <div class="img_play">
        <img src="img/1.jpg" alt="alan">
        <i class="bi playListPlay bi-play-circle-fill" id="25"></i>
      </div>
      <h5>On My Way
        <br>
        <div class="subtitle">Alan Walker</div>
      </h5>
    </li>
  </div>
</div>
```

```
    <div class="popular_artists">
        <div class="h4">
            <h4>Popular Artists</h4>
            <div class="btn_s">
                <i id="left_scrolls" class="bi bi-arrow-left-short"></i>
                <i id="right_scrolls" class="bi bi-arrow-right-short"></i>
            </div>
        </div>

        <div class="item">
            <li>
                <img src="img/jus.jpg"  title="Justin bieber">
            </li>
            <li>
                <img src="img/jit.jpg"  title="Arjit singh">
            </li>
            <li>
                <img src="img/malik.jpg"  title="Armaan maalik">
            </li>
            <li>
                <a href="pra.html"><img src="img/pra.jpg"  title="Pradeep
Kumar"></a>
            </li>
            <li>
                <img src="img/sid.jpg" title="Sid Sriram">
            </li>
            <li>
```

```
                <a href="u1.html"><img src="img/u1.jpg" title="Yuvan"></a>
            </li>
            <li>
                <a href= "il.html"> <img src="img/il.jpg"  title="Ilayaraja"></a>
            </li>
            <li>
                <img src="img/hj.jpg" title="Harris jayaraj">
            </li>
            <li>
                <a href="ar.html"><img src="img/arr.jpg"  title="AR
RAHMAN"></a>
            </li>
            <li>
                <img src="img/ani.jpg" title="Aniruth">
            </li>
            <li>
                <img src="img/walk.jpg"title="Alan Walker">
            </li>
        </div>
    </div>
</div>


<div class="master_play">
    <div class="wave">
        <div class="wave1"></div>
        <div class="wave1"></div>
```

```
            <div class="wave1"></div>
        </div>
        <img src="img/26th.jpg"  id="poster_master_play">
        <h5 id="title">Vande Mataram<br>
            <div class="subtitle">Bankim Chandra</div>
        </h5>
        <div class="icon">
            <i class="bi bi-skip-start-fill" id="back"></i>
            <i class="bi bi-play-fill" id="masterPlay"></i>
            <i class="bi bi-skip-end-fill" id="next"></i>
        </div>
        <span id="currentStart">0:00</span>
        <div class="bar">
            <input type="range" id="seek" min="0" value="0" max="100">
            <div class="bar2" id="bar2"></div>
            <div class="dot"></div>
        </div>
        <span id="currentEnd">0:00</span>

        <div class="vol">
            <i class="bi bi-volume-down-fill" id="vol_icon"></i>
            <input type="range" id="vol" min="0" value="30" max="100">
            <div class="vol_bar"></div>
            <div class="dot" id="vol_dot"></div>
        </div>
    </div>
</header>
```

```
  <script src="app.js"></script>
</body>

</html>
```

JAVASCRIPT

```
        const music = new Audio('vande.mp3');

// create Array

const songs = [
  {
    id:'1',
    songName:` Alone<br>
    <div class="subtitle">Alan Walker</div>`,
    poster: "img/1.jpg"
  },
  {
    id:'2',
    songName:` Alan Walker-Fade <br>
    <div class="subtitle">Alan Walker</div>`,
    poster: "img/2.jpg"
  },
  // all object type
  {
    id:"3",
    songName: `LILY <br><div class="subtitle"> Alan Walker</div>`,
    poster: "img/3.jpg",
},
{
   id:"4",
   songName: `On my way <br><div class="subtitle">Alan walker</div>`,
   poster: "img/4.jpg",
},
{
   id:"5",
   songName: `My stupid heart<br><div class="subtitle">Ertugrul</div>`,
   poster: "img/5.jpg",
},
{
   id:"6",
   songName: `You <br><div class="subtitle">Armaan Maalik</div>`,
   poster: "img/6.jpg",
},
{
   id:"7",
   songName: `Photograph <br><div class="subtitle">Ed sheern</div>`,
   poster: "img/7.jpg",
},

{
   id:"8",
   songName: `Sirikalam Parakalam <br><div class="subtitle">Aniruth</div>`,
   poster: "img/10.jpg",
},
{
   id:"11",
   songName: `Maalai neram <br><div class="subtitle">Andrea</div>`,
   poster: "img/11.jpg",
},
{
   id:"12",
   songName: `Cheap thrills <br><div class="subtitle">Sia</div>`,
   poster: "img/12.jpg",
},
{
   id:"13",
   songName: `Dandelions <br><div class="subtitle">Ruth B</div>`,
   poster: "img/13.jpg",
},
{
   id:"14",
   songName: `Thunder <br><div class="subtitle">Imagine Dragons</div>`,
   poster: "img/14.jpg",
},
{
   id:"15",
   songName: `Bad liar<br><div class="subtitle">Imagine Dragons </div>`,
   poster: "img/15.jpg",
},
{
   id:"16",
        songName: `Shayad<br><div class="subtitle">Arjit singh</div>`,
        poster: "img/16.jpg",
    },
    {
        id:"17",
        songName: `En roja neeye<br><div class="subtitle">Hesham abdul
wahab</div>`,
        poster: "img/17.jpg",
    },
    {
        id:"18",
        songName: `Pudhidha<br><div class="subtitle">Gopi sundar</div>`,
        poster: "img/18.jpg",
    },
    {
        id:"19",
        songName: `poradalam <br><div class="subtitle">Besabriyan</div>`,
        poster: "img/19.jpg",
    },
    {
        id:"20",
        songName: `Shape of you<br><div class="subtitle">Ed sheern</div>`,
        poster: "img/20.jpg",
    },
    {
        id:"21",
        songName: `Little things<br><div class="subtitle">Bobby</div>`,
        poster: "img/21.jpg",
    },
    {
        id:"22",
        songName: `Legends Never Die<br><div class="subtitle">Against the
current</div>`,
        poster: "img/22.jpg",
    },
    {
        id:"23",
        songName: `1000 YEARS<br><div class="subtitle">Alan</div>`,
        poster: "img/23.jpg",
    },
    {
        id:"24",
        songName: `Treat you better<br><div class="subtitle">Shawn
mendes</div>`,
        poster: "img/24.jpg",
    },
    {
        id:"25",
        songName: `10,000 hours<br><div class="subtitle">Justin bieber</div>`,
        poster: "img/25.jpg",
    },
]
```

```
Array.from(document.getElementsByClassName('songItem')).forEach((element,
i)=>{
    element.getElementsByTagName('img')[0].src = songs[i].poster;
    element.getElementsByTagName('h5')[0].innerHTML = songs[i].songName;
})



let masterPlay = document.getElementById('masterPlay');
let wave = document.getElementsByClassName('wave')[0];

masterPlay.addEventListener('click',()=>{
    if (music.paused || music.currentTime <=0) {
        music.play();
        masterPlay.classList.remove('bi-play-fill');
        masterPlay.classList.add('bi-pause-fill');
        wave.classList.add('active2');
    } else {
        music.pause();
        masterPlay.classList.add('bi-play-fill');
        masterPlay.classList.remove('bi-pause-fill');
        wave.classList.remove('active2');
    }
} )



const makeAllPlays = () =>{
```

```
Array.from(document.getElementsByClassName('playListPlay')).forEach((element
)=>{
            element.classList.add('bi-play-circle-fill');
            element.classList.remove('bi-pause-circle-fill');
    })
}
const makeAllBackgrounds = () =>{

Array.from(document.getElementsByClassName('songItem')).forEach((element)=
>{
            element.style.background = "rgb(105, 105, 170, 0)";
    })
}

let index = 0;
let poster_master_play = document.getElementById('poster_master_play');
let title = document.getElementById('title');
Array.from(document.getElementsByClassName('playListPlay')).forEach((element
)=>{
    element.addEventListener('click', (e)=>{
        index = e.target.id;
        makeAllPlays();
        e.target.classList.remove('bi-play-circle-fill');
        e.target.classList.add('bi-pause-circle-fill');
        music.src = `audio/${index}.mp3`;
        poster_master_play.src =`img/${index}.jpg`;
        music.play();
        let song_title = songs.filter((ele)=>{
            return ele.id == index;
        })

        song_title.forEach(ele =>{
            let {songName} = ele;
            title.innerHTML = songName;
        })
        masterPlay.classList.remove('bi-play-fill');
        masterPlay.classList.add('bi-pause-fill');
        wave.classList.add('active2');
        music.addEventListener('ended',()=>{
            masterPlay.classList.add('bi-play-fill');
            masterPlay.classList.remove('bi-pause-fill');
            wave.classList.remove('active2');
        })
        makeAllBackgrounds();
        Array.from(document.getElementsByClassName('songItem'))[`${index-1}`].style.background = "rgb(105, 105, 170, .1)";
    })
})


let currentStart = document.getElementById('currentStart');
let currentEnd = document.getElementById('currentEnd');
let seek = document.getElementById('seek');
let bar2 = document.getElementById('bar2');
let dot = document.getElementsByClassName('dot')[0];

music.addEventListener('timeupdate',()=>{
    let music_curr = music.currentTime;
    let music_dur = music.duration;

    let min = Math.floor(music_dur/60);
    let sec = Math.floor(music_dur%60);
    if (sec<10) {
        sec = `0${sec}`
    }
    currentEnd.innerText = `${min}:${sec}`;

    let min1 = Math.floor(music_curr/60);
    let sec1 = Math.floor(music_curr%60);
    if (sec1<10) {
        sec1 = `0${sec1}`
    }
    currentStart.innerText = `${min1}:${sec1}`;

    let progressbar = parseInt((music.currentTime/music.duration)*100);
    seek.value = progressbar;
    let seekbar = seek.value;
    bar2.style.width = `${seekbar}%`;
    dot.style.left = `${seekbar}%`;
})
```

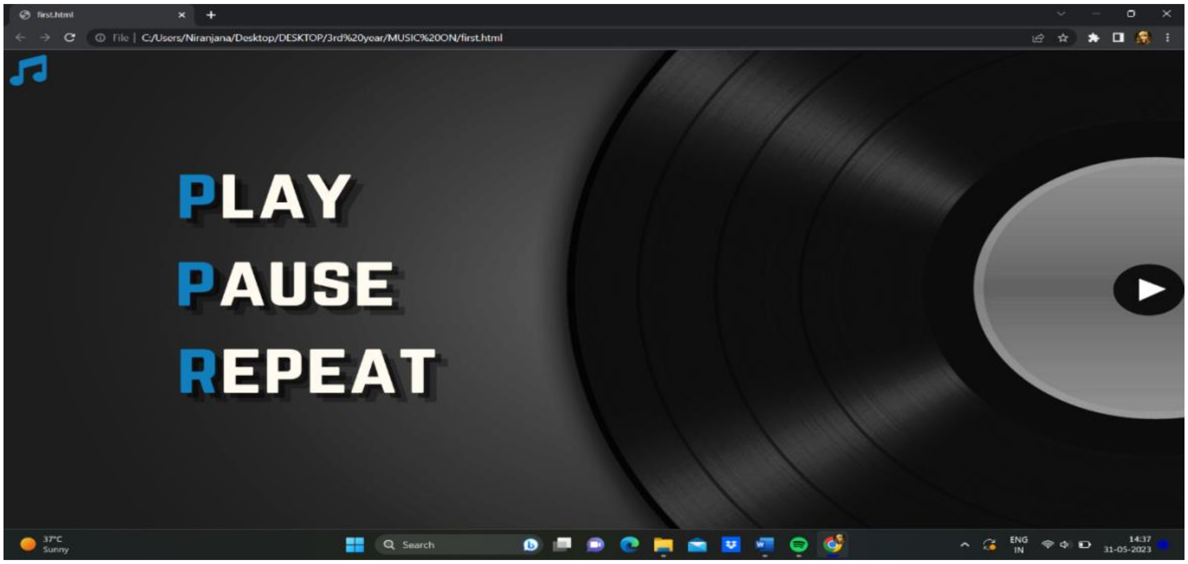
```
seek.addEventListener('change', ()=>{
    music.currentTime = seek.value * music.duration/100;
})

music.addEventListener('ended', ()=>{
    masterPlay.classList.add('bi-play-fill');
    masterPlay.classList.remove('bi-pause-fill');
    wave.classList.remove('active2');
})



let vol_icon = document.getElementById('vol_icon');
let vol = document.getElementById('vol');
let vol_dot = document.getElementById('vol_dot');
let vol_bar = document.getElementsByClassName('vol_bar')[0];

vol.addEventListener('change', ()=>{
    if (vol.value == 0) {
        vol_icon.classList.remove('bi-volume-down-fill');
        vol_icon.classList.add('bi-volume-mute-fill');
        vol_icon.classList.remove('bi-volume-up-fill');
    }
    if (vol.value > 0) {
        vol_icon.classList.add('bi-volume-down-fill');
        vol_icon.classList.remove('bi-volume-mute-fill');
        vol_icon.classList.remove('bi-volume-up-fill');
    }
    if (vol.value > 50) {
        vol_icon.classList.remove('bi-volume-down-fill');
        vol_icon.classList.remove('bi-volume-mute-fill');
        vol_icon.classList.add('bi-volume-up-fill');
    }

    let vol_a = vol.value;
    vol_bar.style.width = `${vol_a}%`;
    vol_dot.style.left = `${vol_a}%`;
    music.volume = vol_a/100;
})



let back = document.getElementById('back');
let next = document.getElementById('next');

back.addEventListener('click', ()=>{
    index -= 1;
    if (index < 1) {
        index = Array.from(document.getElementsByClassName('songItem')).length;
    }
    music.src = `audio/${index}.mp3`;
    poster_master_play.src =`img/${index}.jpg`;
    music.play();
    let song_title = songs.filter((ele)=>{
        return ele.id == index;
    })

    song_title.forEach(ele =>{
        let {songName} = ele;
        title.innerHTML = songName;
    })
    makeAllPlays()

    document.getElementById(`${index}`).classList.remove('bi-play-fill');
    document.getElementById(`${index}`).classList.add('bi-pause-fill');
    makeAllBackgrounds();
    Array.from(document.getElementsByClassName('songItem'))[`${index-
1}`].style.background = "rgb(105, 105, 170, .1)";

})
next.addEventListener('click', ()=>{
    index -= 0;
    index += 1;
    if (index > Array.from(document.getElementsByClassName('songItem')).length)
{
        index = 1;
        }
    music.src = `audio/${index}.mp3`;
    poster_master_play.src =`img/${index}.jpg`;
    music.play();
    let song_title = songs.filter((ele)=>{
        return ele.id == index;
    })

    song_title.forEach(ele =>{
        let {songName} = ele;
        title.innerHTML = songName;
    })
    makeAllPlays()

    document.getElementById(`${index}`).classList.remove('bi-play-fill');
    document.getElementById(`${index}`).classList.add('bi-pause-fill');
    makeAllBackgrounds();
    Array.from(document.getElementsByClassName('songItem'))[`${index-1}`].style.background = "rgb(105, 105, 170, .1)";

})



let left_scroll = document.getElementById('left_scroll');
let right_scroll = document.getElementById('right_scroll');
let pop_song = document.getElementsByClassName('pop_song')[0];

left_scroll.addEventListener('click', ()=>{
    pop_song.scrollLeft -= 330;
})
right_scroll.addEventListener('click', ()=>{
    pop_song.scrollLeft += 330;
```

```
})


let left_scrolls = document.getElementById('left_scrolls');
let right_scrolls = document.getElementById('right_scrolls');
let item = document.getElementsByClassName('item')[0];

left_scrolls.addEventListener('click', ()=>{
   item.scrollLeft -= 330;
})
right_scrolls.addEventListener('click', ()=>{
   item.scrollLeft += 330;
})
```

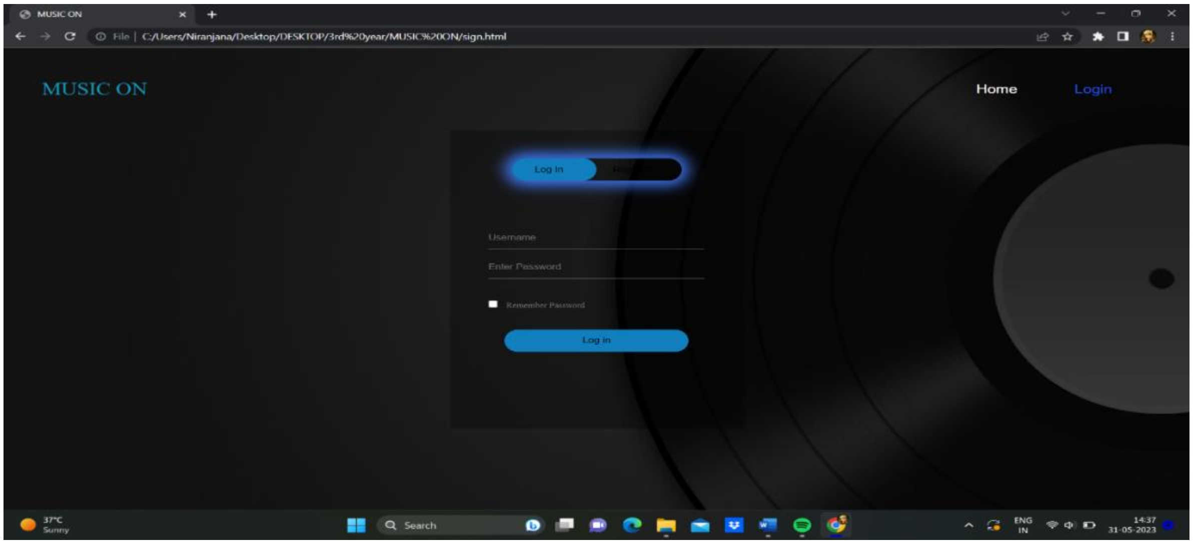
OUTPUT

ONLINE MUSIC PORTAL

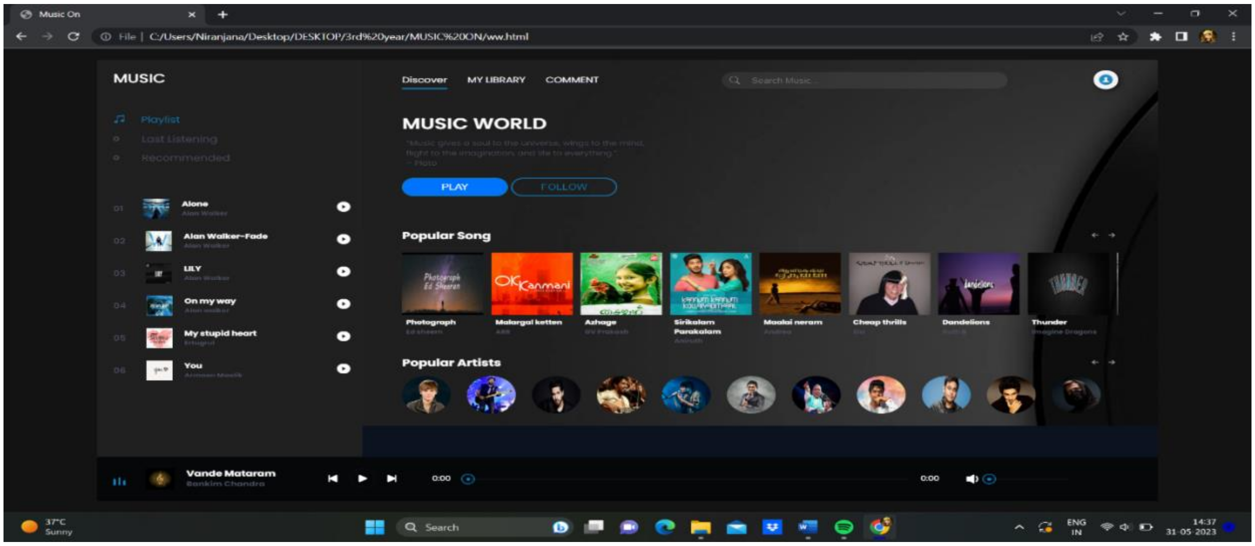
HOME PAGE


first.html
File | C:/Users/Niranjana/Desktop/DESKTOP/3rd%20year/MUSIC%20ON/first.html
PLAY
PAUSE
REPEAT
37°C
Sunny
Search
ENG
IN
14:37
31-05-2023

LOGIN PAGE



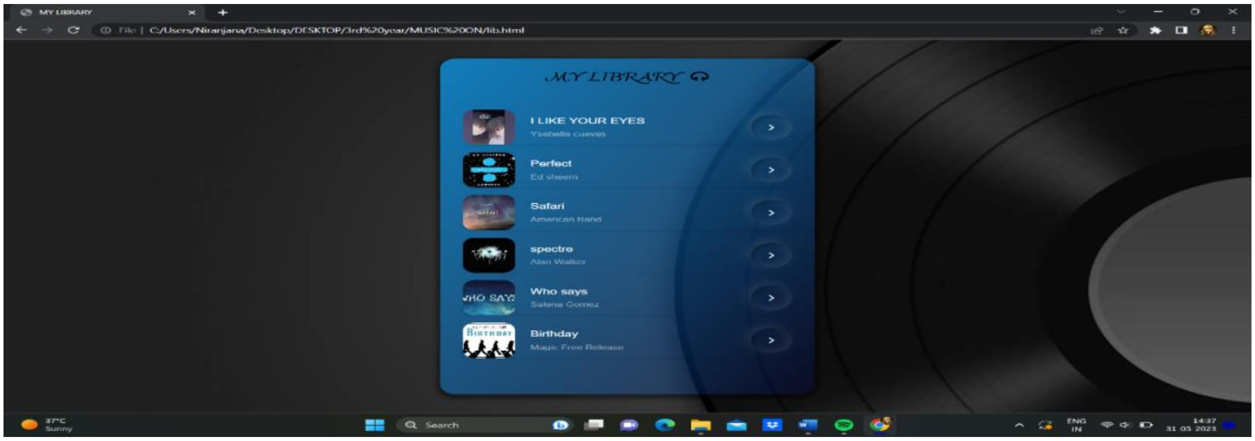
MAIN PAGE



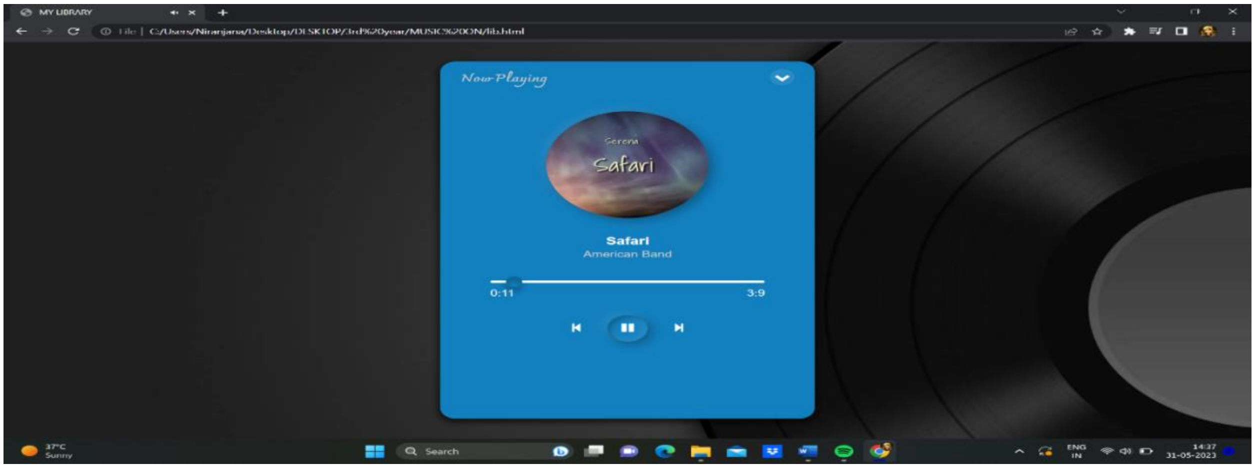
C:/Users/Niranjana/Desktop/DESKTOP/3rd%20year/MUSIC%20ON/ww.html

"Music gives a soul to the universe, wings to the mind, flight to the imagination and life to everything."
– Plato

MY LIBRARY

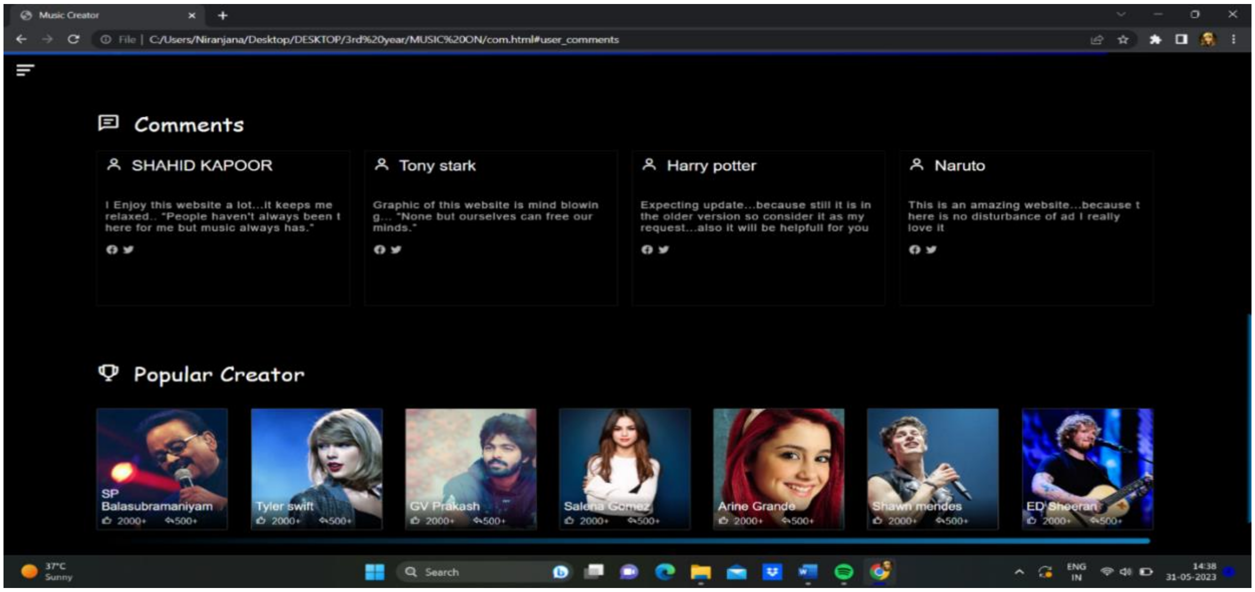


C:/Users/Niranjana/Desktop/DESKTOP/3rd%20year/MUSIC%20ON/lib.html



C:/Users/Niranjana/Desktop/DESKTOP/3rd%20year/MUSIC%20ON/lib.html

COMMENT



Comments

SHAHID KAPOOR

I Enjoy this website a lot...it keeps me
relaxed.. "People haven't always been t
here for me but music always has."

Tony stark

Graphic of this website is mind blowin
g... "None but ourselves can free our
minds."

Harry potter

Expecting update...because still it is in
the older version so consider it as my
request...also it will be helpfull for you

Naruto

This is an amazing website...because t
here is no disturbance of ad I really
love it

Conclusion

Through the development of online music portal,we get a clear understanding of overall process of the system.

The core part of the online music portal is mainly composed of music interface, song listing,Grasping the development of the online music portal has had the preliminary scale small features.

Music player system realized the basic function of player:play pause,rewind and fastforward, a volume adjust button.

This development implicated by HTML,CSS,JAVA.

Made statement of the aims and objective of the project. We define the problem on which we are working in the project. We included features and operations in the detail. Finally , the Online Music Portal was build successfully.