A Project Report
On
Virtual Musical Instrument

Submitted in partial fulfillment of the
requirement for the award of the degree of

Bachelor of Technology in computer science and Engineering

Under The Supervision of

Dr. S Annamalai

(Associate Professor)

Department of Computer Science and Engineering

Submitted By:

Ankit Kumar (18SCSE1010596)

Sumit Dixit (18SCSE1010191)

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

GALGOTIAS UNIVERSITY, GREATER NOIDA, INDIA

DECEMBER - 2021

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA

CANDIDATE'S DECLARATION

I/We hereby certify that the work which is being presented in the project, entitled "**Virtual Musical Instrument** " in partial fulfillment of the requirements for the award of the **BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING** submitted in the **School of Computing Science and Engineering of Galgotias University, Greater Noida**, is an original work carried out during the period of **JULY-2021 to DECEMBER-2021,** under the supervision of **DR.S Annamalai, Associate Professor, Department of Computer Science and Engineering** Galgotias University, Greater Noida.

The matter presented in the project has not been submitted by me/us for the award of any other degree of this or any other places.

Ankit Kumar (18SCSE1010596)

Sumit Dixit (18SCSE1010191)

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Supervisor

(DR.S Annamalai,Associate Professor)

Acknowledgment

Apart from the efforts, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project. We would like to show my greatest appreciation to "**DR. S Annamalai**" and also our dean **"Dr. Munish Sabarwal**". We can't say thank you enough for his tremendous support and help. We feel motivated and encouraged every time, We attend his meeting. Without his encouragement and guidance, this project would not have materialized. The guidance and support received from all the members who contributed and who are contributing to this project were vital for the success of the project. We are grateful for their constant support and help.

CERTIFICATE

The Final Thesis/Project/ Dissertation Viva-Voce examination of **Ankit Kumar (18SCSE1010596), Sumit Dixit(18SCSE1010191)** has been held on_____and his/her work is recommended for the award of **BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING.**

Signature of Examiner(s) **Signature of Supervisor(s)**

Signature of Project Coordinator **Signature of Dean**

Date:

Place:

Abstract:

Web Development plays a keen role in the development of websites and web applications. Websites are like interfaces that connect human beings to the computer system.
through web development techniques we can design and also add the functionalities of the website according to the need.
For this, languages like HTML, CSS, and Javascript are used.
In This project, a website is presented named as Virtual Music instrument in which we have added the sound of different musical instruments like drum, flute, guitar. Users can listen to the sound of instruments by hovering over the different parts of drum-like to play the sound individually of each part. This can be done through the keyboard also by pressing the key associated with each part. In guitar also we can click on each string of guitar to listen to the sound of each string of the guitar. and similarly with the other musical instruments also. This website will also consist of the link to visit the music player page in which there will be lists of all songs from the different types and of the different eras which will be arranged In a proper manner so that users will find it very easy to play the songs of their choice.

List of Figures:

S.no	Caption	Page No.
1.	Output image-1,	18
2.	Output image-2,3	19
3.	Output image-4	20
4.	Output Image-5	21
5.	Flowchart	24
6.	Use case Diagram	25
7.	Sourcecode Snaps	26-44

Acronyms

B.Tech.	Bachelor of Technology
B.Tech.	Bachelor of Technology

Table of Contents

Title		Page No.
Candidates Declaration		
Acknowledgment		
Abstract		
List of Figures		
Acronyms		
Chapter 1	**Introduction**	**1**
	1.1 Requirement of Virtual Musical Instrument.	1
	1.2 Advantage of Proposed System.	2
	1.3 Tools and Technology Used.	3
Chapter 2	**Literature Survey/Project Design**	**16**
	2.1 Literature Survey	16
	2.2 Project Design	17
Chapter 3	**Functionality/Working of Project**	**21**
	3.1 Flowchart	23
	3.2 Use case diagram	24
Chapter 4	**Results and Discussion**	**25**
	4.1 Source code	
Chapter 5	**Conclusion and Future scope**	**45**
	5.1 Conclusion	45
	5.2 Future scope	46
	References	**47**

Chapter 1- Introduction

1.1 Requirement of Virtual Musical Instrument:

Computers have become pervasive today. However, we are interacting with it using traditional input methods like keyboard, mouse, and most recently the touch screen. It is just like a robot that performs the task according to the command which is passed through the user as input. Nowadays many technologies are introduced and with the help of computers these technologies can be implemented and we can experience the uses of that technology. In this project, we have used web technology through which we can hear or experience music. Everybody likes to play music if not to, at least listen to music. One of the barriers to playing music for recreation is the high cost of musical instruments. Many people want to play musical instruments like piano, drum, guitar, and many such instruments once in a while but are not ready to spend more on it. Also, musical instruments require space and regular maintenance. you may feel like trying a new musical instrument after playing the old one for a long time. Also, music is one of the best ways to relieve the different kinds of stress and pressure we have in day-to-day life. This pulsive situation motivated us to apply the skills of web development and develop this project named Virtual Music Instrument.

1.2 Advantage of Proposed System:

This website gives a virtual reality of the musical instruments so if a person is unfamiliar with any kind of musical instrument he can gain experience of the sound produced by that instrument by listening to it virtually.

A user can practice the instrument by playing Sounds of different types of instruments like piano, drum, or flute before purchasing that instrument so that after buying that he will be able to know which instrument he is good at. By doing this he can also gain more confidence while playing the instrument.

By playing different kinds of instruments virtually the user can make his own compositions of music and this will be a very good platform for beginners as well.

We can’t take or carry musical instruments anywhere except some small instruments like mouth organs, flute, etc. but if someone has the mood to play the musical instrument then he can play at any time with the help of virtual musical instruments.

1.3 Tools and Technology Used:

We have used HTML, CSS, Javascript, and Bootstrap frameworks as well for the development of this website.

1. HTML-HTML stands for Hyper Text markup language.HTML consists of a series of elements that describes the structure of a webpage.it has pre-defined tags that tells the browser how to display content on the webpage. Using HTML images audio and video can also be added to a web page. HTML has various Versions all the versions of HTML are:
   (i)HTML 1.0 (Released in 1991)
   (ii)HTML 2.0 (Released in 1995)
   (iii)HTML 3.2 (Released in 1997)
   (iv)HTML 4.01 (Released in 1999)
   (v)HTML 5 (Released in 2008)
   In this project, we have used HTML 5 which is the latest version of HTML.
   Advantages of HTML:
   - It is Used to make Websites.
   - It is supported by all Browsers.
   - It is platform-independent.
   - HTML has various Tags and Attributes which can shorten the line of code
2. CSS-CSS Stands for cascading styling sheet.css handled the look and feel part of the web page. By using CSS text style, style of fonts spacing between the paragraphs, How columns are sized and layout, what background colors or images are used. layout design, Variations in display for different devices, and screen sizes as well.

Different Versions of CSS are:
(i) CSS1(Released in 1996)
(ii) CSS2(Released in 1998)
(iii) CSS3(Released in 1999)
Advantages of CSS:

- CSS saves time − You can write CSS once and then reuse the same sheet in multiple HTML pages. You can define a style for each HTML element and apply it to as many Web pages as you want.
- Pages load faster − If you are using CSS, you do not need to write HTML tag attributes every time. Just write one CSS rule of a tag and apply it to all the occurrences of that tag. So less code means faster download times.
- Easy maintenance − To make a global change, simply change the style, and all elements in all the web pages will be updated automatically.
- Superior styles to HTML − CSS has a much wider array of attributes than HTML, so you can give a far better look to your HTML page in comparison to HTML attributes.
- Multiple Device Compatibility − Style sheets allow content to be optimized for more than one type of device. By using the same HTML document, different versions of a website can be presented for handheld devices such as PDAs and cell phones or for printing.

3. Javascript-javascript is a text-based programming language in which is used both on the client-side and server-side that allows you to make web pages interactive.it helps you develop great front-end as well as back-end software using different Javascript-based frameworks like jQuery, Node.JS, etc

Different Versions of Javascript are:
(i)ECMAScript 1(Released in 1997)
(ii)ECMAScript 2(Released in 1998)
(iii)ECMAScript 3(Released in 1999)
(iv)ECMAScript 4(Not Released)
(v)ECMAScript 5(Released in2009)
(vi)ECMAScript 6(Released in 2015)
(vii)ECMAScript 7(Released in 2016)
(vii)ECMAScript 8(Released in 2017)
(ix)ECMAScript 9(Released in 2018)
Advantages of javascript:

- Less server interaction − You can validate user input before sending the page off to the server. This saves server traffic, which means less load on your server.
- Immediate feedback to the visitors − They don't have to wait for a page reload to see if they have forgotten to enter something.
- Increased interactivity − You can create interfaces that react when the user hovers over them with a mouse or activates them via the keyboard.
- Richer interfaces − You can use JavaScript to include such items as drag-and-drop components and sliders to give a Rich Interface to your site visitors.

4.BootStrap-Bootstrap is a framework to help you design websites faster and easier. It includes HTML and CSS-based design templates for typography, forms, buttons, tables, navigation, modals, image carousels, etc. It also gives you support for JavaScript plugins.

There are many versions of Bootstrap in which the latest version is Bootstrap version 5 which is supported to all the browsers.

Advantages of Bootstrap:

- The bootstrap framework comes up with predefined terms, thus giving you the ability to use these codes instead of having to create the codes from scratch.
- A consistent framework that supports major of all browsers and CSS compatibility fixes
- Several JavaScript plugins using the jQuery
- Good documentation and community support
- Loads of free and professional templates, WordPress themes, and plugins.

5. Atom-**Atom** is a free and open-source code editor for macOS, Linux, and Microsoft Wi with support for plug-ins written in Javascript, and embedded Git control. Developed by GitHub, Atom is a desktop application built using web technologies. Most of the extending packages have free software licenses and are community-built and maintained. Atom is based on Electron (formerly known as Atom Shell), a framework that enables cross-platform desktop applications using Chromium and Node.jsAtom is written in Coffee Script and Less, but much of it has been converted to JavaScript.Atom was released from beta, as of version 1.0, on 25 June 2015. Its developers call it a "hackable text editor for the 21st Century" as it is fully customizable in HTML, CSS, and JavaScript.Atom is a "hackable" text editor, which means it is customizable. There is an init script one can customize using CoffeeScript, a style sheet to customize the looks of Atom, and a keymap to map or remap key combinations to commands. One can

even make a package to wrap all of this functionality into a single package, written in their choice of CoffeeScript or JavaScript.

Like most other configurable text editors, Atom enables users to install third-party packages and themes to customize the features and looks of the editor. Packages can be installed, managed, and published via Atom's package manager APM. All types of packages, including but not limited to: Syntactic highlighting support for other languages than the default, debuggers, etc. can be installed via APM.

Chapter-2 Literature Survey/Project Design

Literature Survey

In this project, we are trying to show the use and importance of web development with the help of features of web development. We can design websites like this that can entertain people. Many people if they want to learn music they must be aware of the musical tunes and also the musical instruments on which tunes are to be played so before buying an instrument it will be good for them to get some experience about the musical tunes so that users can get knowledge about the instruments and tune as well. As this website consists of musical sound of various musical instruments so it can be helpful to the users in many ways especially for music lovers as before buying an instrument of his choice he will be able to learn how to play the instrument that he wants to buy and also hear the sound of it so that he can gain a lot of experience. For designing the website we have used HTML(Hypertext markup language), CSS(cascading Styling sheet), javascript, and Bootstrap framework as well.HTML consists of a series of elements that describes the structure of a web page.it also tells the webpage that how to display content on the webpage.CSS describes that how HTML elements are displayed on the screen.it can control the layout of multiple web Pages at once.it can be used to style the web page such as background color, text styling and in CSS we have features of Media queries by the help of the elements of the web page can be properly aligned according to our choice in the different kinds of

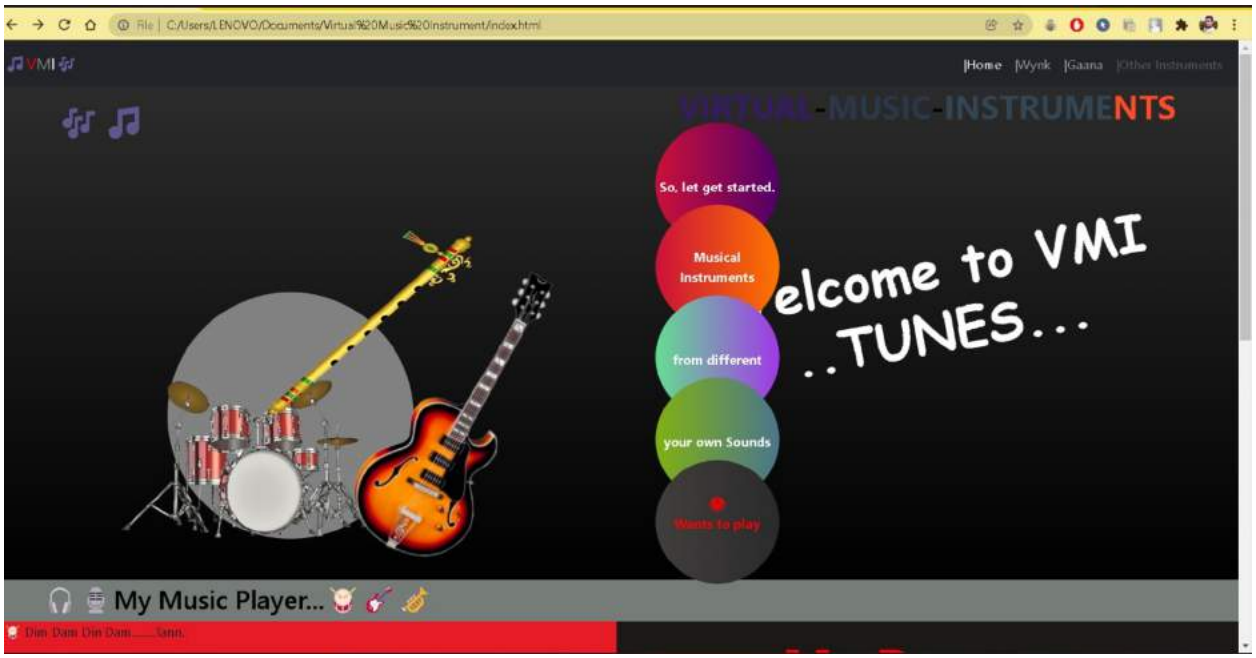
devices such as tabs, Smartphones, and computer screen. of the elements of web pages in the different devices.

Javascript is a Lightweight object-oriented programming language that is used to make web pages more interactive.it provides dynamic interactivity on the website when it is applied with HTML Document.it also helps users to build modern web applications to interact directly without reloading the page every time. JavaScript is commonly used to dynamically modify HTML and CSS to update a user interface by the DOM API. It is mainly used in web applications. Javascript can also be used for creating games. It has various libraries and frameworks for creating a game. Some JavaScript game engines such as **PhysicsJS, Pixi.js** help us to create a web game. We can also use the **WebGL** (web graphics library), which is the JavaScript API to render 2D and 3D images on browsers.

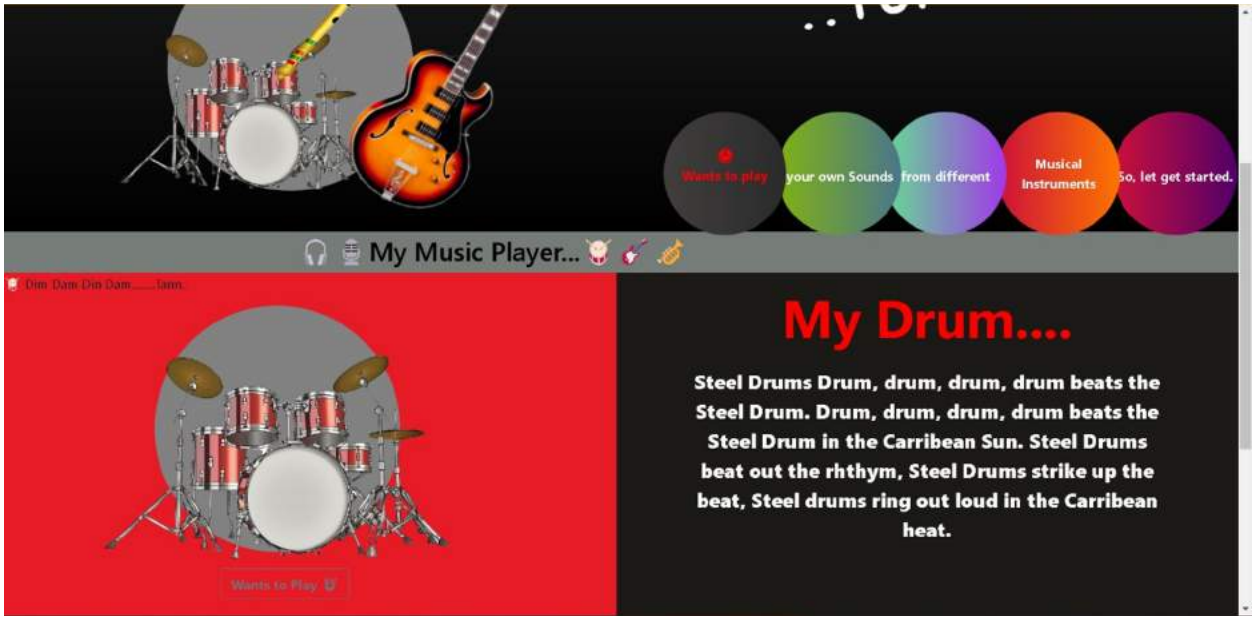
Bootstrap is a CSS framework that is used to develop responsive and mobile-first websites.it helps you design websites faster and easier. It includes HTML and CSS-based design templates for typography, forms, buttons, tables, navigation, modals, image carousels, etc. It also gives you support for JavaScript plugins. There are various versions of Bootstrap and the latest version is Bootstrap Version 5.

Project Design

When the website opens then at the starting of the website we have designed the home screen in which all the links are provided, the links for playing the musical instruments as well as the link to visit the music player.



Link to Musical Instruments:

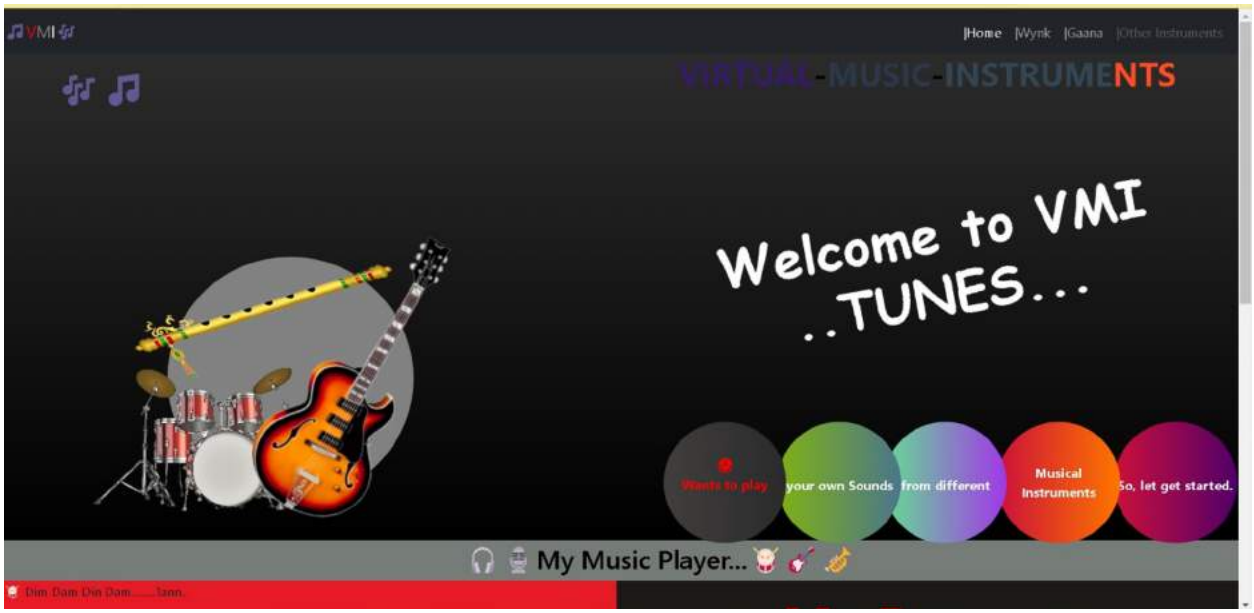


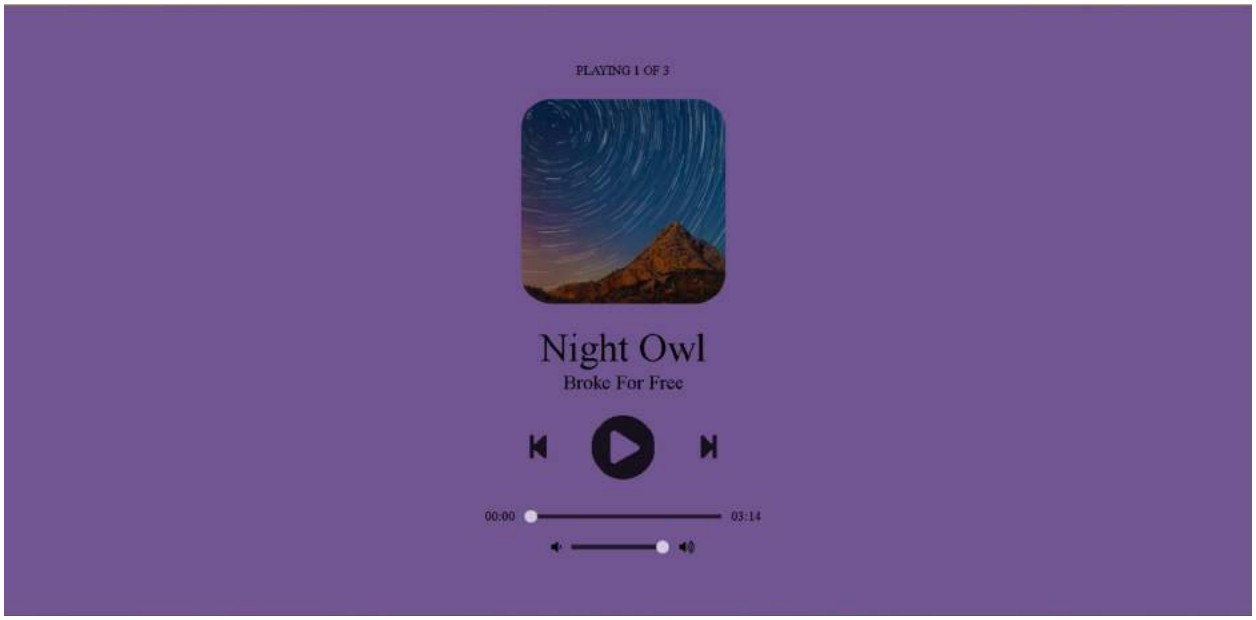
If the user wants to play a musical instrument then he must click on the wants to play button and the instrument player will open.

On scrolling, the mouse over these the sound of the drum can be heard this can also be done by pressing respective keys of the keyboard.

Link To Music Player:

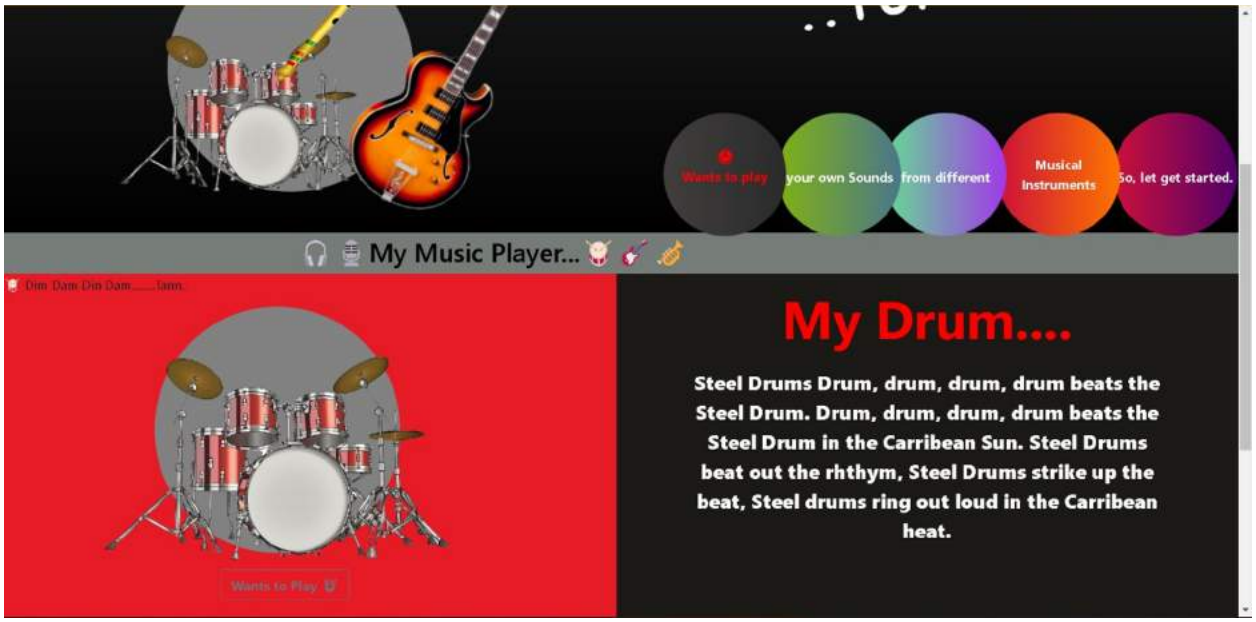
On clicking on the Music Player displaying on the screen it will take to the music player.





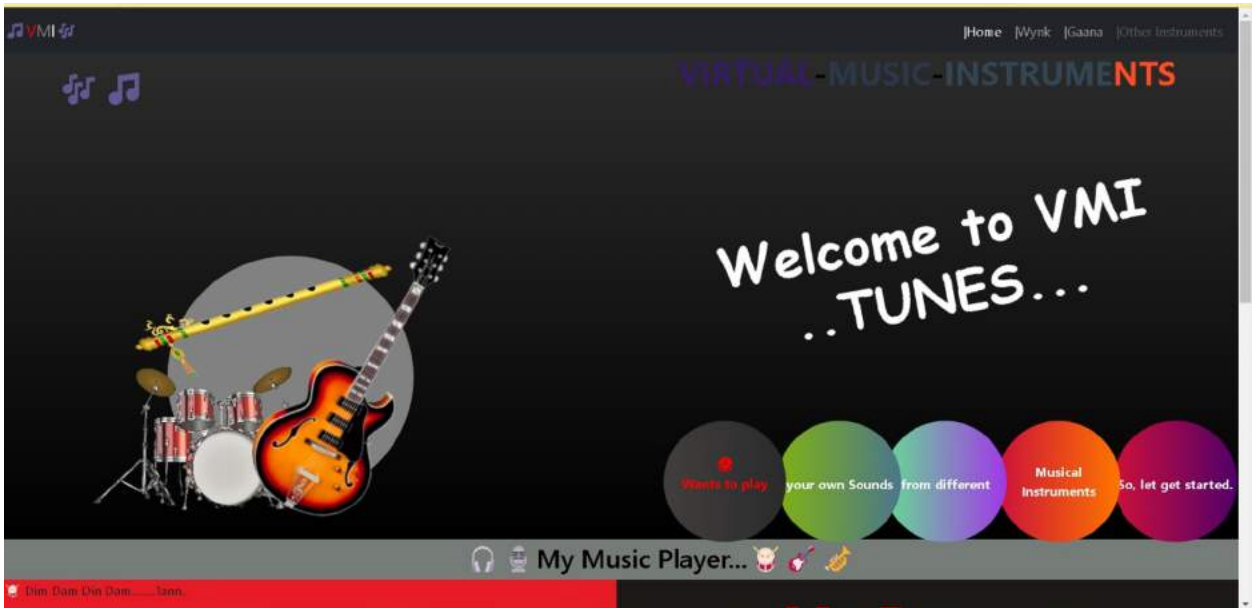
It has a forward-backward as well as a play and pause button. Through that, we can change the song.

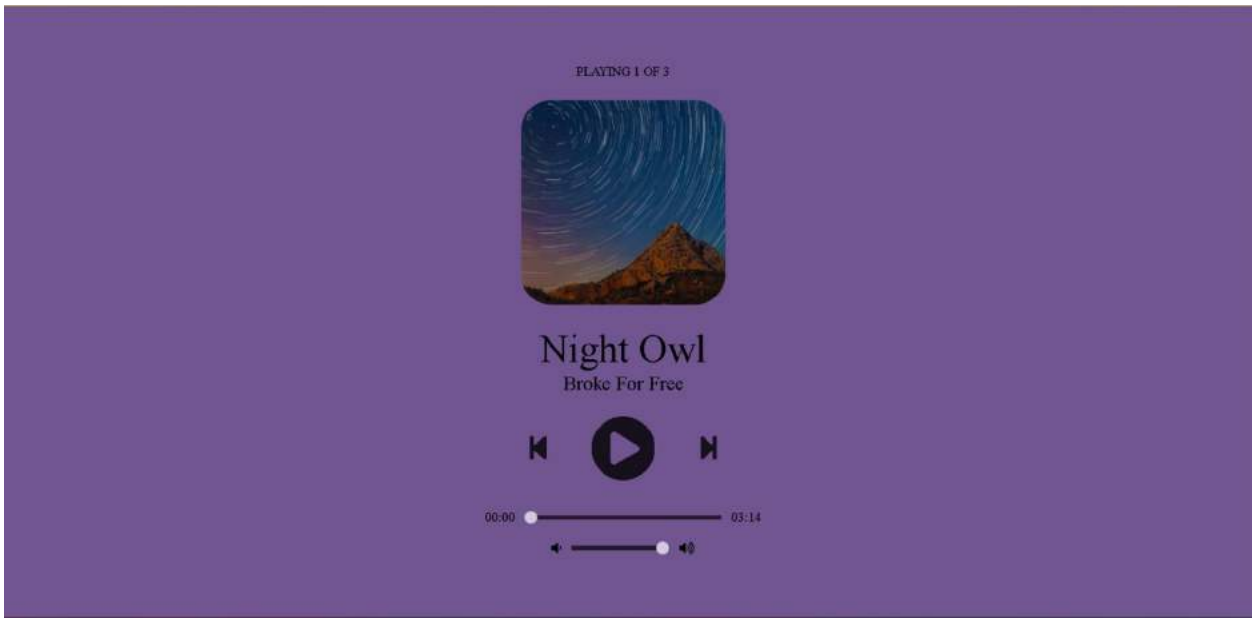
Chapter-3 Functionality/Working of Project



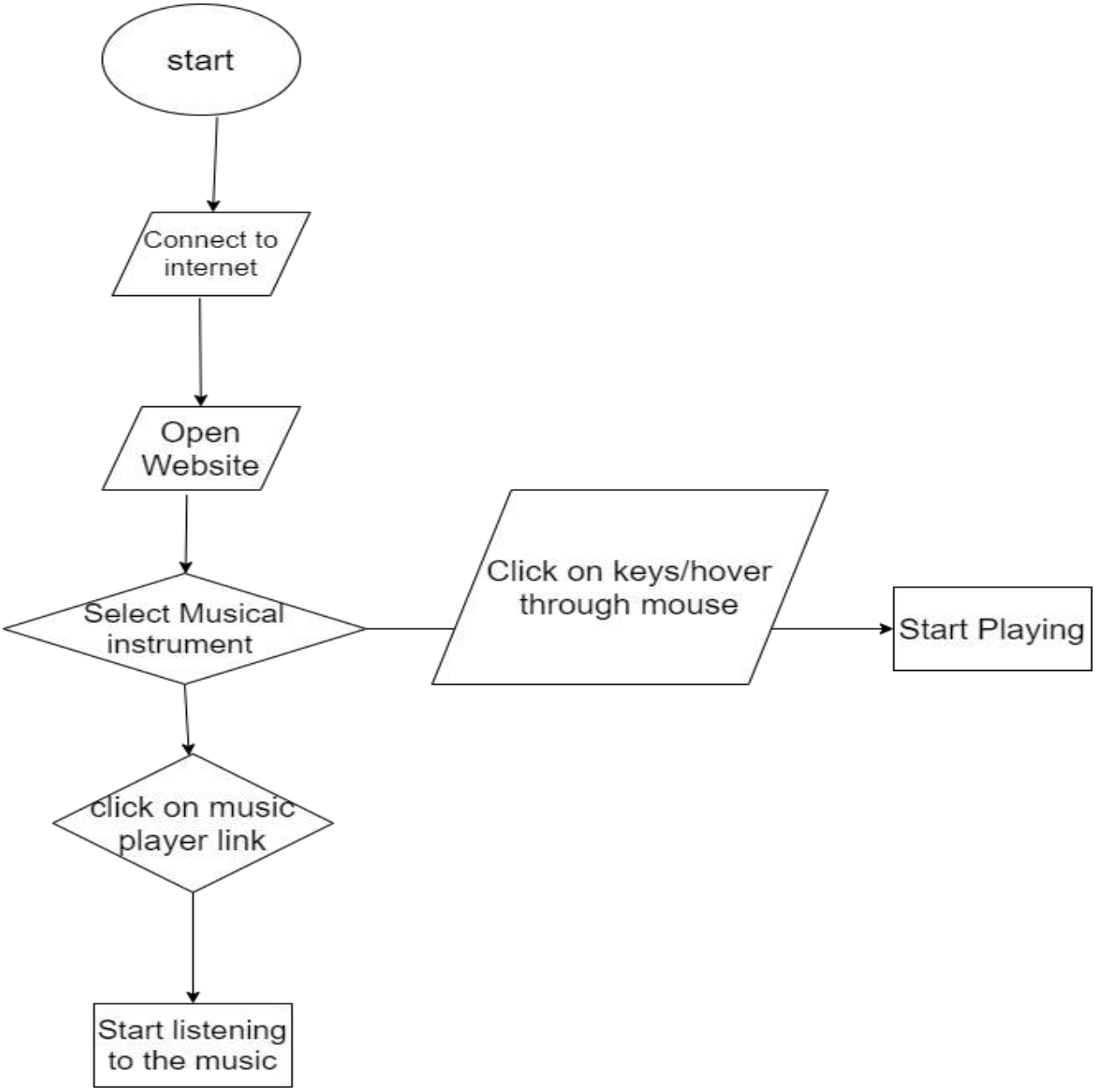
On clicking on the button below the musical instrument a new page will open in which there will be the sound of all the parts of that instrument.

On clicking on the musical player link, users can play music.

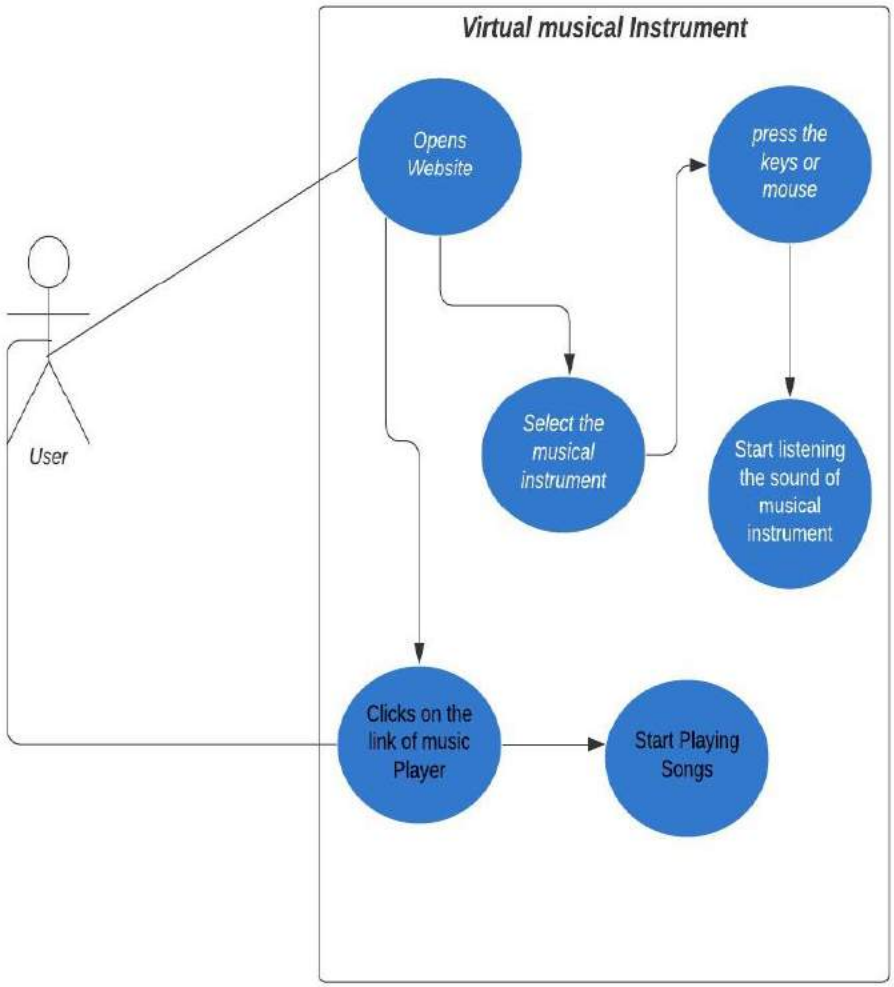




Flowchart


start
Connect to internet
Open Website
Select Musical instrument
Click on keys/hover through mouse
Start Playing
click on music player link
Start listening to the music

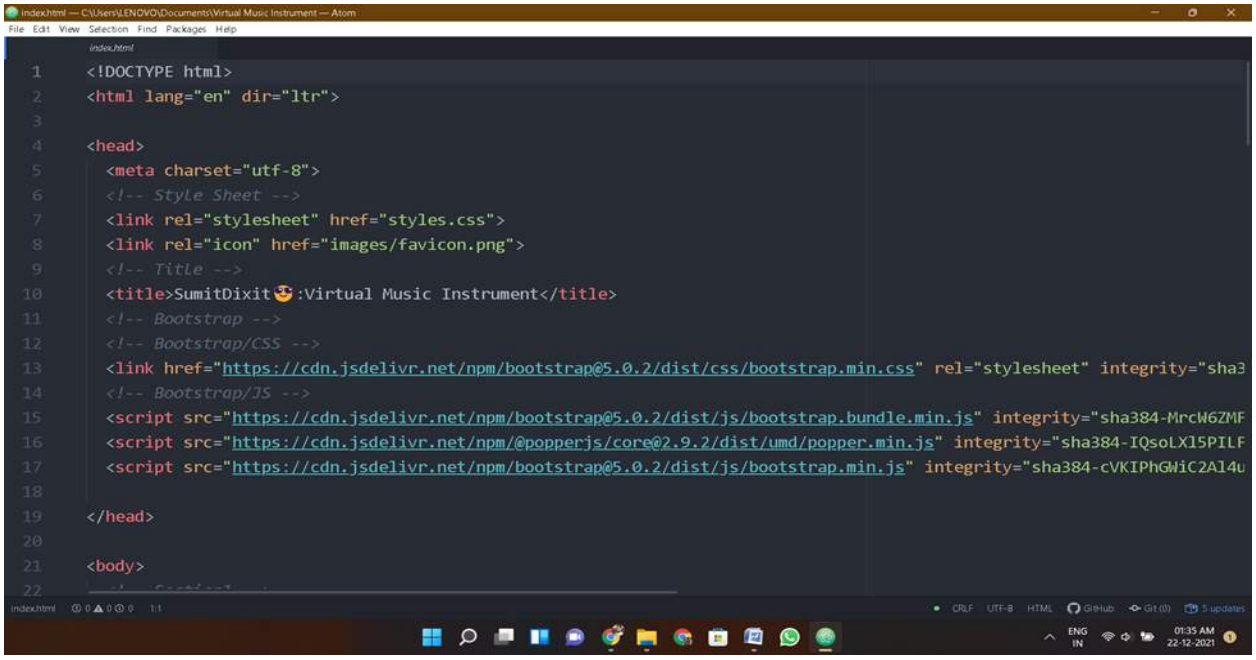
Use Case Diagram



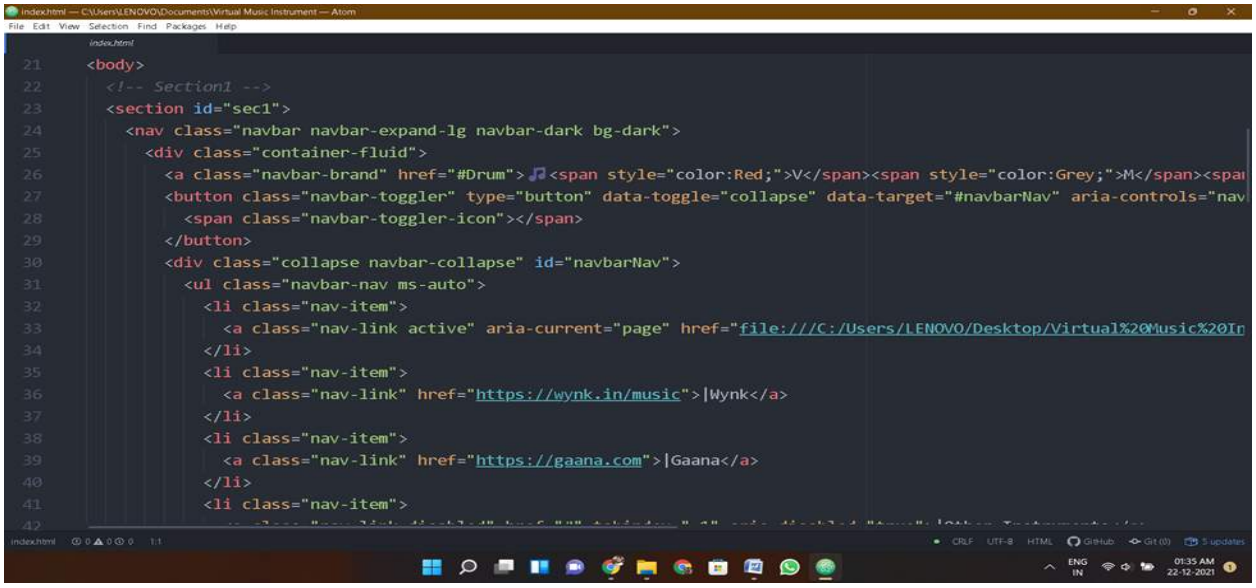
Chapter-4 Results and Discussion

Source Code

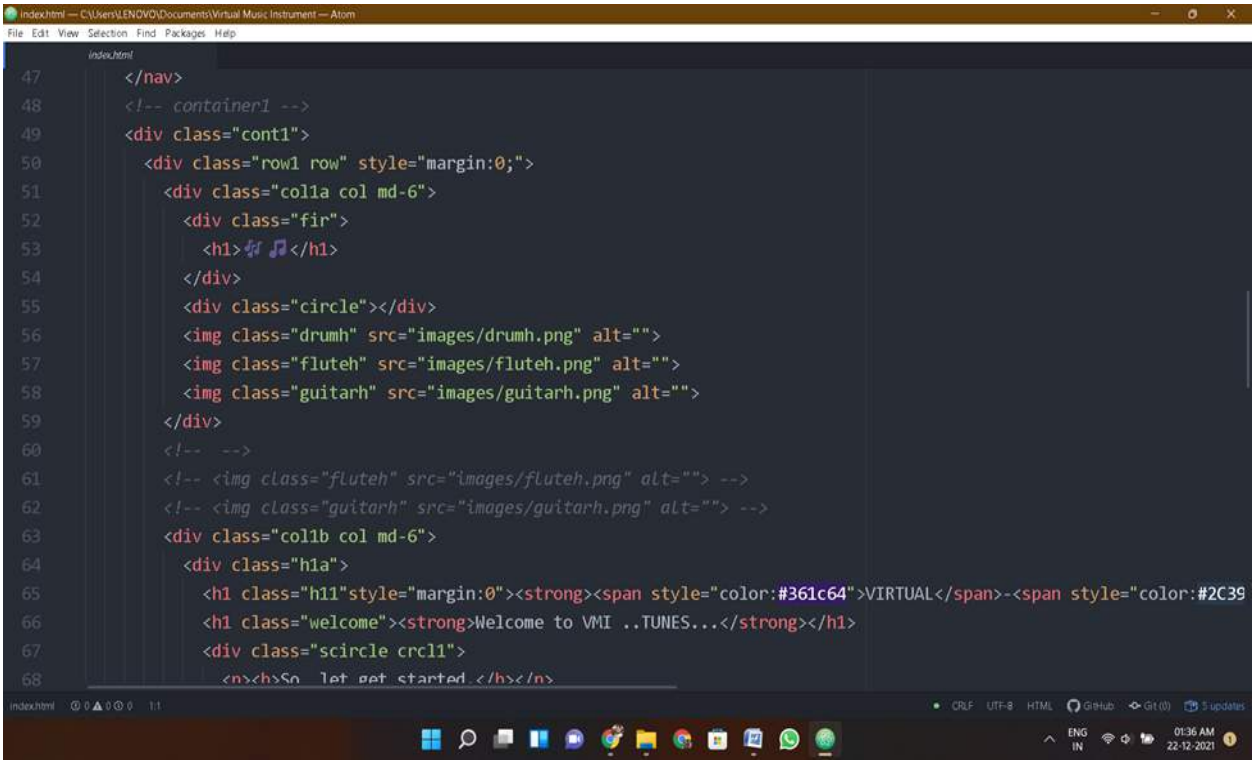
Home Page (HTML)



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

<head>
  <meta charset="utf-8">
  <!-- Style Sheet -->
  <link rel="stylesheet" href="styles.css">
  <link rel="icon" href="images/favicon.png">
  <!-- Title -->
  <title>SumitDixit😍:Virtual Music Instrument</title>
  <!-- Bootstrap -->
  <!-- Bootstrap/CSS -->
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha3
  <!-- Bootstrap/JS -->
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js" integrity="sha384-MrcW6ZMF
  <script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.9.2/dist/umd/popper.min.js" integrity="sha384-IQsoLXl5PILF
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.min.js" integrity="sha384-cVKIPhGWiC2Al4u

</head>

<body>
```



```
<body>
  <!-- Section1 -->
  <section id="sec1">
    <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
      <div class="container-fluid">
        <a class="navbar-brand" href="#Drum">🎵<span style="color:Red;">V</span><span style="color:Grey;">M</span><spa
        <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNav" aria-controls="nav
          <span class="navbar-toggler-icon"></span>
        </button>
        <div class="collapse navbar-collapse" id="navbarNav">
          <ul class="navbar-nav ms-auto">
            <li class="nav-item">
              <a class="nav-link active" aria-current="page" href="file:///C:/Users/LENOVO/Desktop/Virtual%20Music%20In
            </li>
            <li class="nav-item">
              <a class="nav-link" href="https://wynk.in/music">|Wynk</a>
            </li>
            <li class="nav-item">
              <a class="nav-link" href="https://gaana.com">|Gaana</a>
            </li>
            <li class="nav-item">
```


index.html — C:\Users\LENOVO\Documents\Virtual Music Instrument — Atom
File Edit View Selection Find Packages Help
index.html
</nav>
<!-- container1 -->

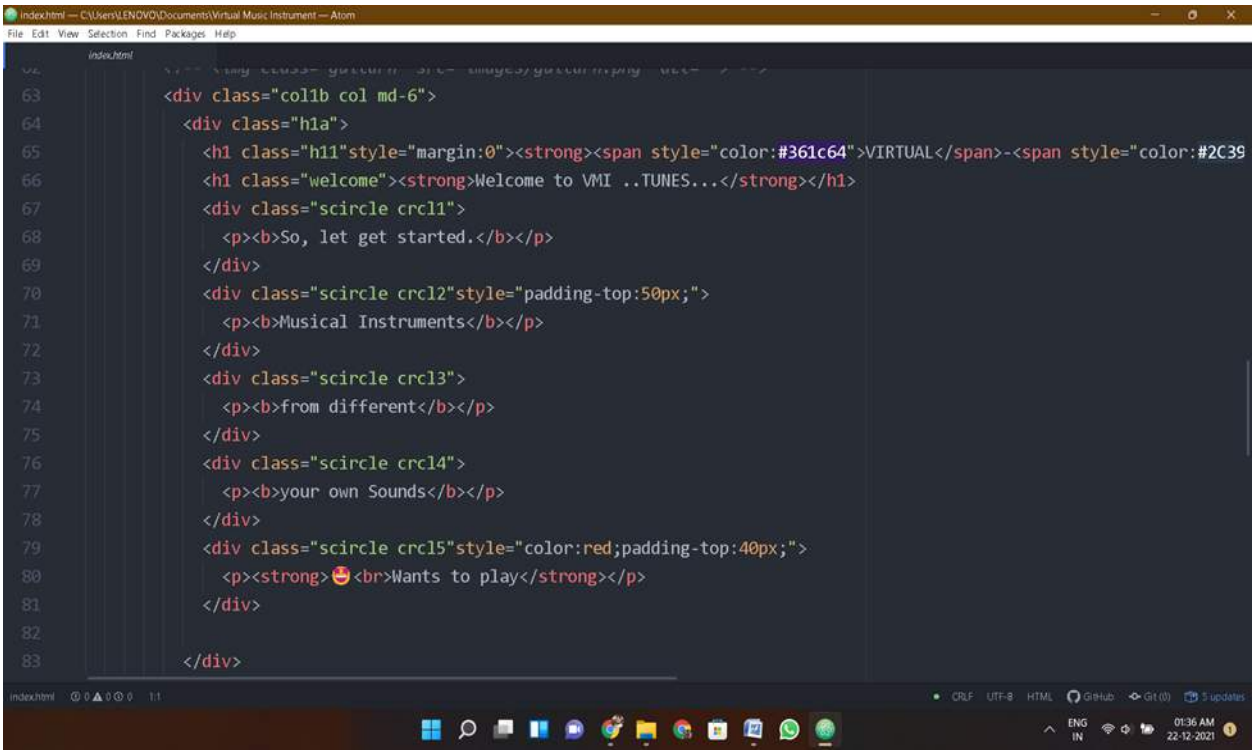
<h1>🎶 🎵</h1>

<img class="drumh" src="images/drumh.png" alt="">
<img class="fluteh" src="images/fluteh.png" alt="">
<img class="guitarh" src="images/guitarh.png" alt="">

<!-- -->
<!-- <img class="fluteh" src="images/fluteh.png" alt=""> -->
<!-- <img class="guitarh" src="images/guitarh.png" alt=""> -->

<h1 class="h11"style="margin:0">VIRTUAL-<span style="color:#2C39
<h1 class="welcome">Welcome to VMI ..TUNES...</h1>

So, let get started.
CRLF UTF-8 HTML GitHub Git (0) 5 updates
ENG IN 01:36 AM 22-12-2021


index.html — C:\Users\LENOVO\Documents\Virtual Music Instrument — Atom
File Edit View Selection Find Packages Help
index.html

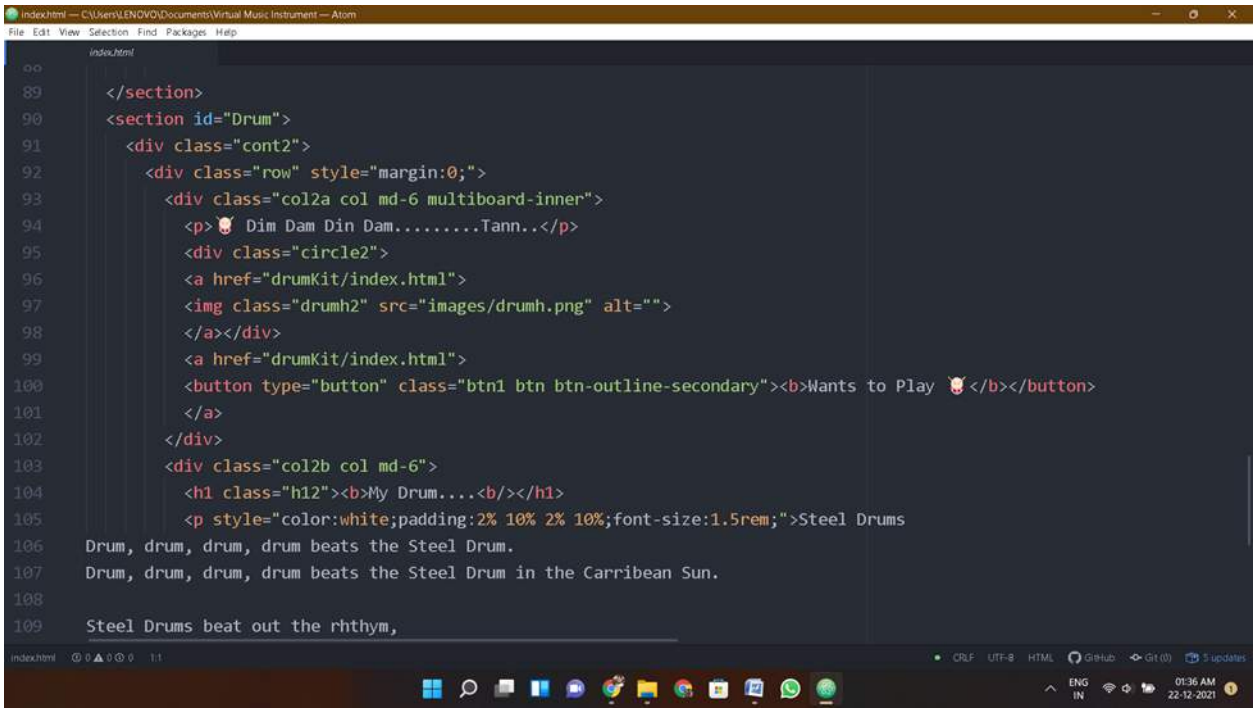
<h1 class="h11"style="margin:0">VIRTUAL-<span style="color:#2C39
<h1 class="welcome">Welcome to VMI ..TUNES...</h1>

So, let get started.
<div class="scircle crc12"style="padding-top:50px;">
Musical Instruments

from different

your own Sounds
<div class="scircle crc15"style="color:red;padding-top:40px;">

Wants to play
CRLF UTF-8 HTML GitHub Git (0) 5 updates
ENG IN 01:36 AM 22-12-2021

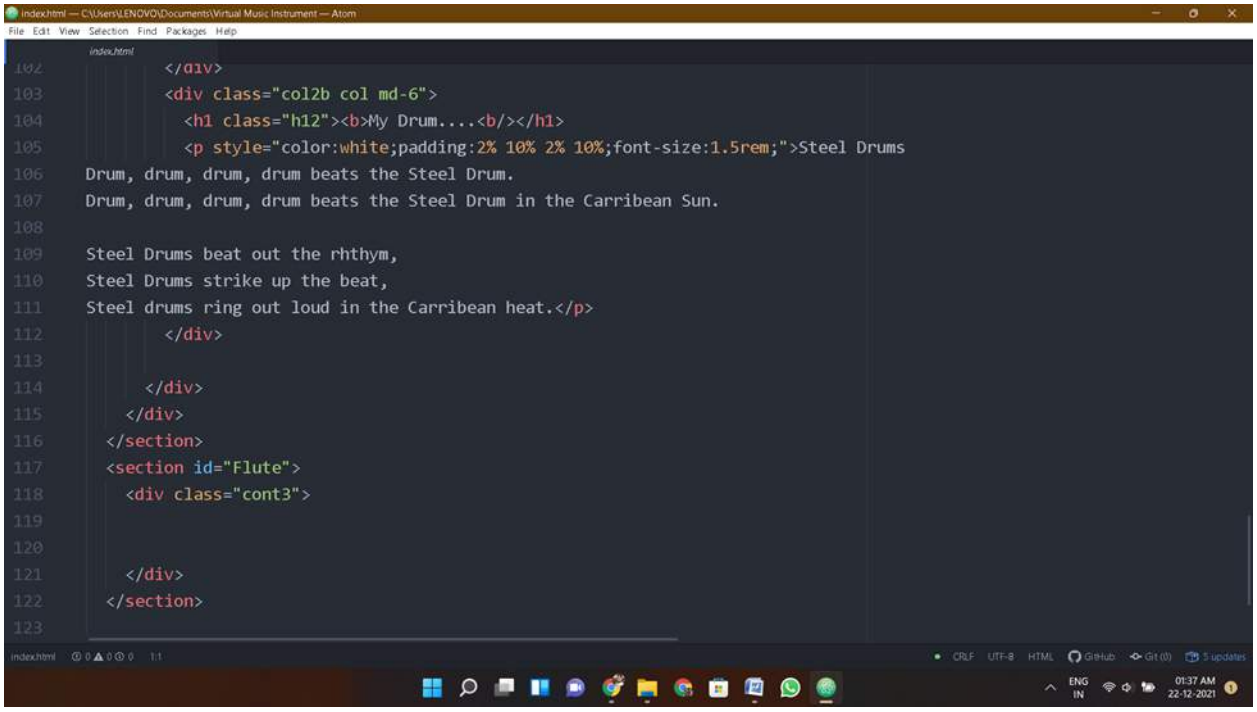

index.html — C:\Users\LENOVO\Documents\Virtual Music Instrument — Atom
File Edit View Selection Find Packages Help
index.html
</section>
<section id="Drum">

 Dim Dam Din Dam.....Tann..

<img class="drumh2" src="images/drumh.png" alt="">

<button type="button" class="btn1 btn btn-outline-secondary">Wants to Play </button>

<h1 class="h12">My Drum....</h1>
Steel Drums
Drum, drum, drum, drum beats the Steel Drum.
Drum, drum, drum, drum beats the Steel Drum in the Carribean Sun.
Steel Drums beat out the rhthym,
index.html
CRLF UTF-8 HTML GitHub Git (0) 5 updates
ENG IN
01:36 AM 22-12-2021

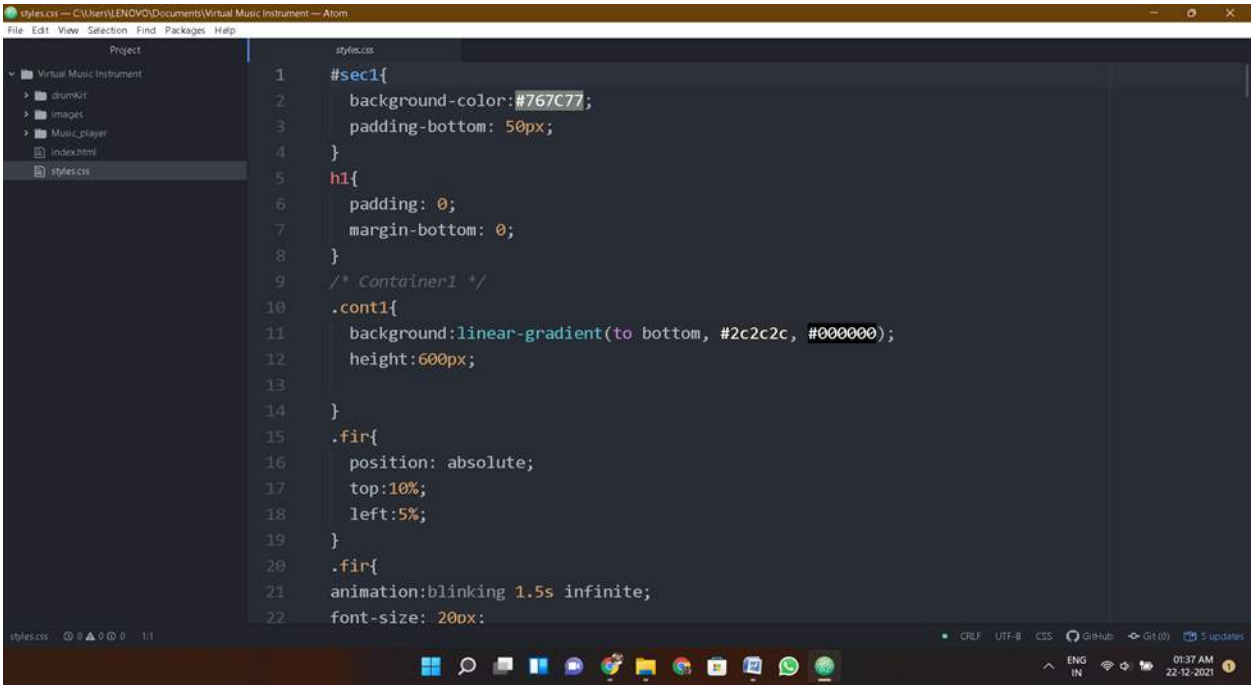

index.html — C:\Users\LENOVO\Documents\Virtual Music Instrument — Atom
File Edit View Selection Find Packages Help
index.html

<h1 class="h12">My Drum....</h1>
Steel Drums
Drum, drum, drum, drum beats the Steel Drum.
Drum, drum, drum, drum beats the Steel Drum in the Carribean Sun.
Steel Drums beat out the rhthym,
Steel Drums strike up the beat,
Steel drums ring out loud in the Carribean heat.

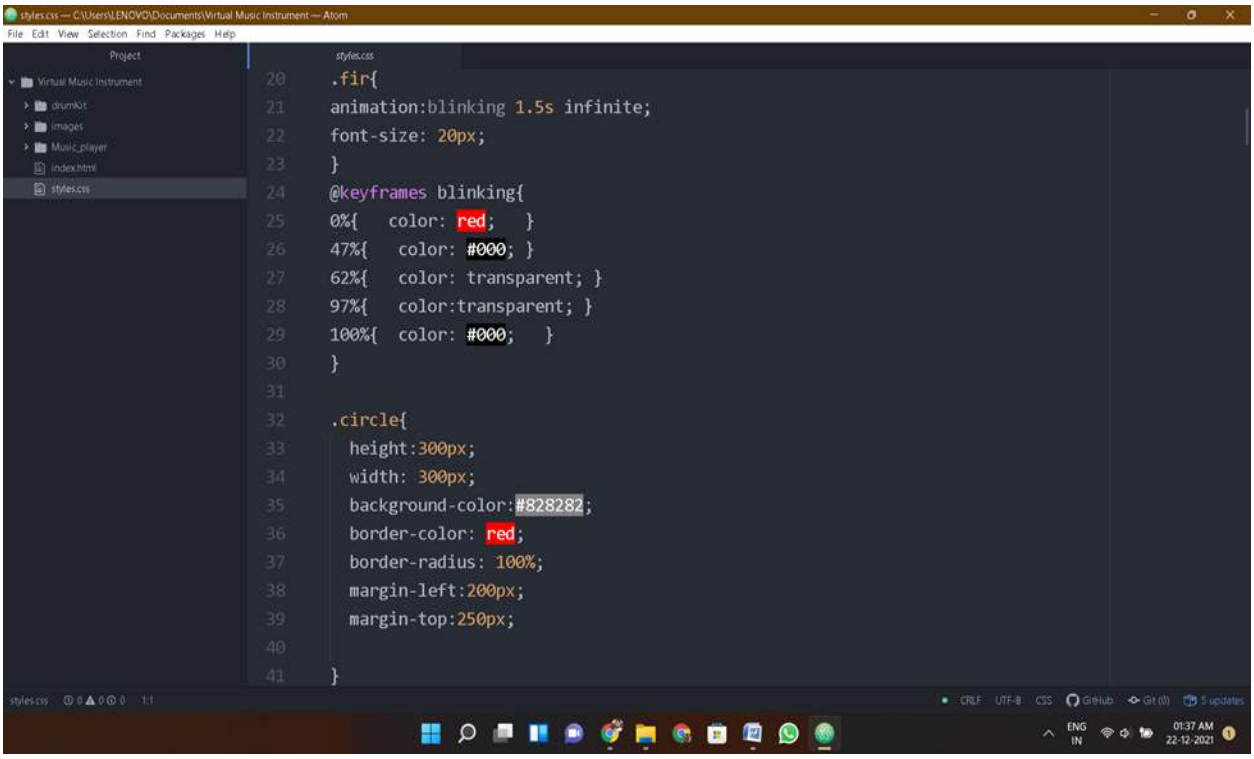
</section>
<section id="Flute">

</section>
index.html
CRLF UTF-8 HTML GitHub Git (0) 5 updates
ENG IN
01:37 AM 22-12-2021

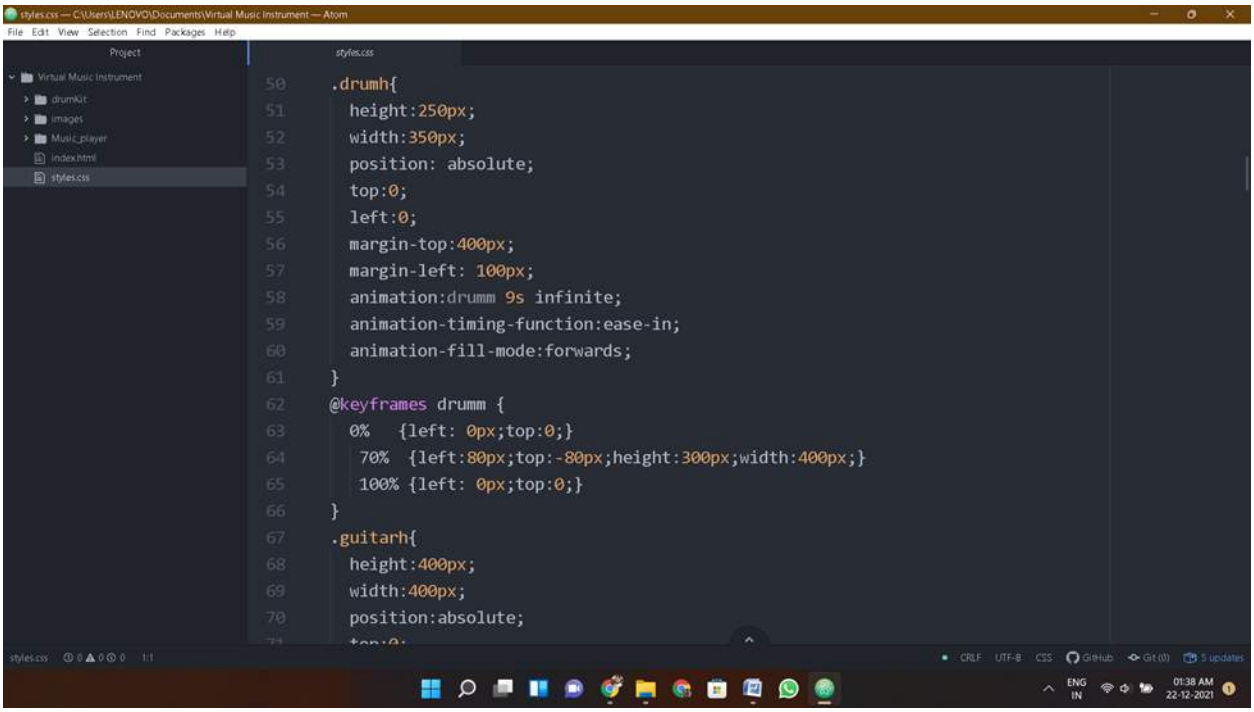
CSS:

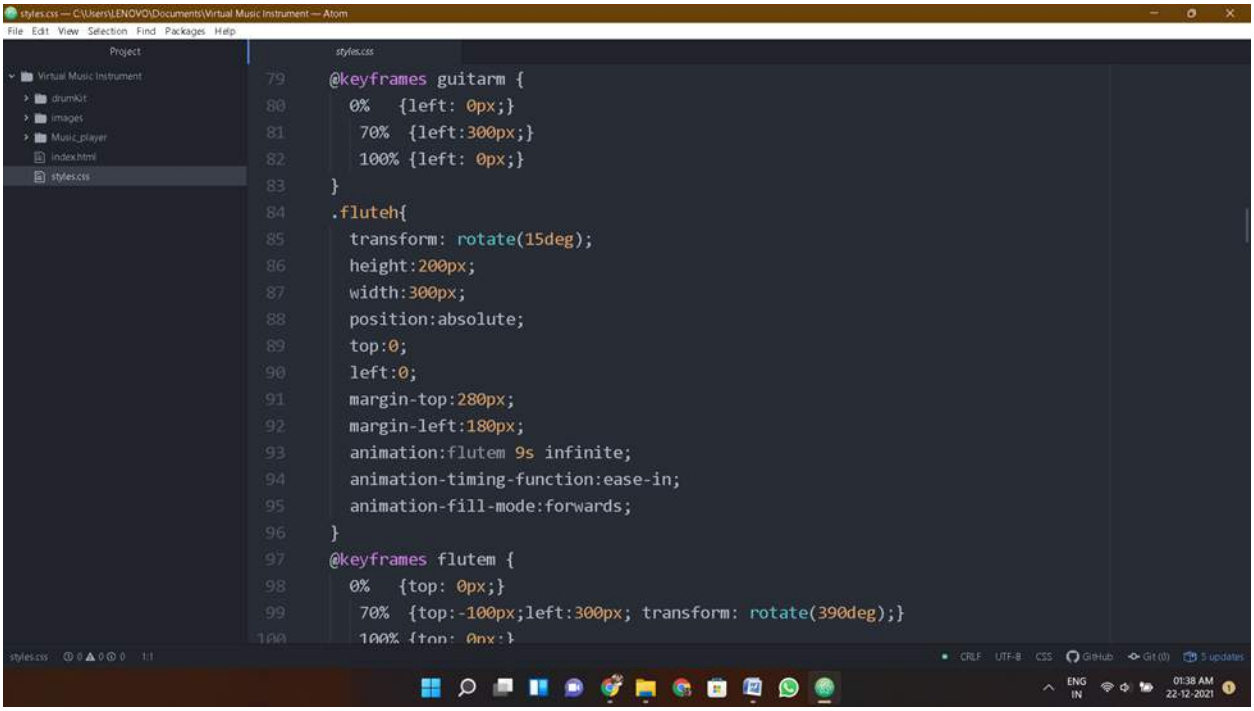


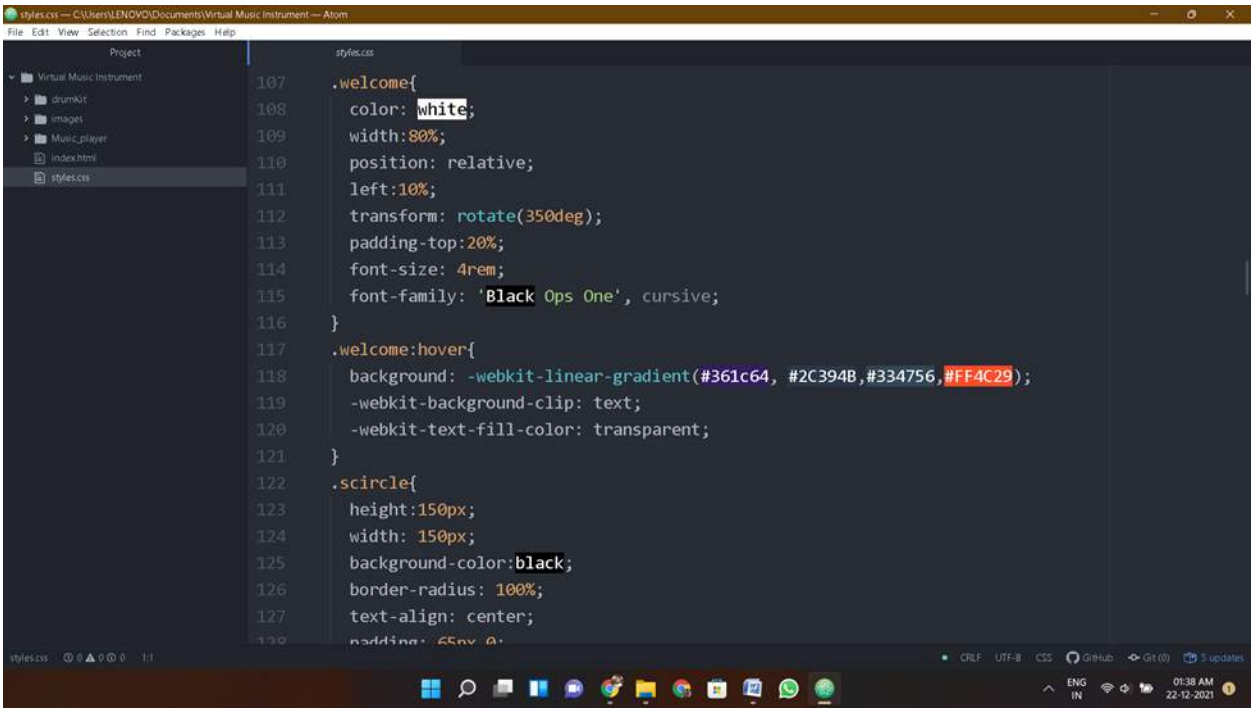
```
#sec1{
  background-color:#767C77;
  padding-bottom: 50px;
}
h1{
  padding: 0;
  margin-bottom: 0;
}
/* Container1 */
.cont1{
  background:linear-gradient(to bottom, #2c2c2c, #000000);
  height:600px;

}
.fir{
  position: absolute;
  top:10%;
  left:5%;
}
.fir{
animation:blinking 1.5s infinite;
font-size: 20px;
```



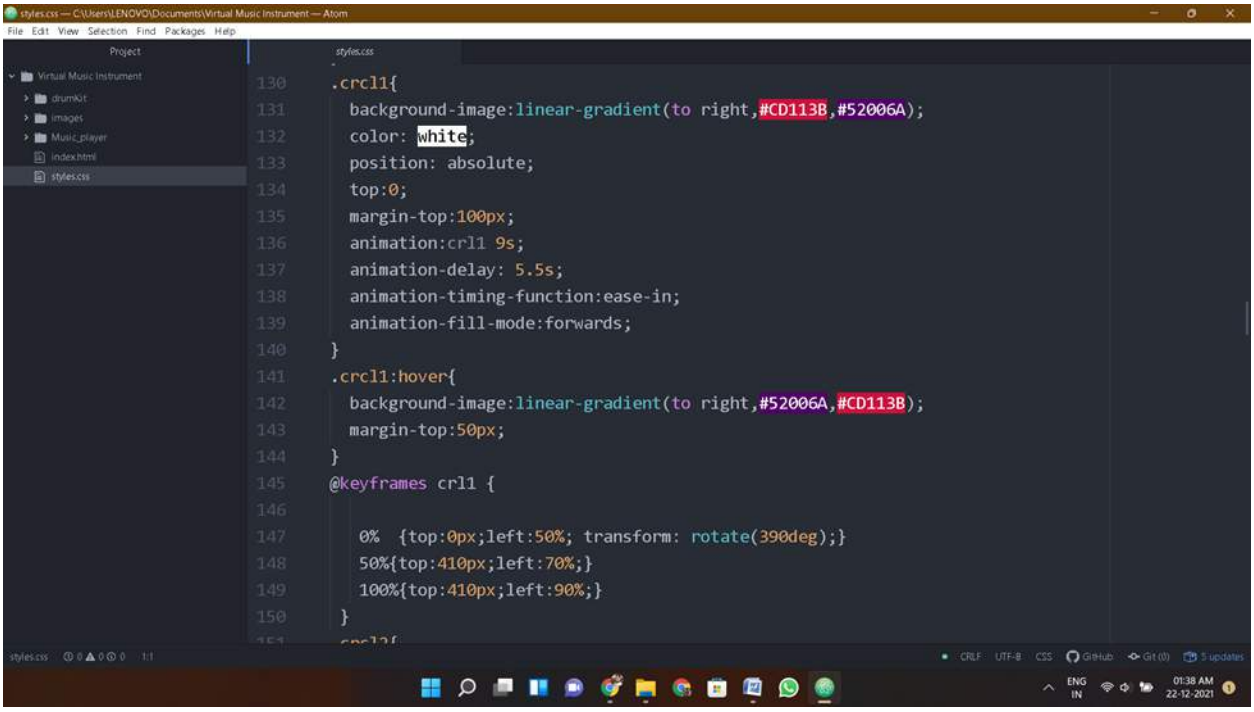
```
.fir{
animation:blinking 1.5s infinite;
font-size: 20px;
}
@keyframes blinking{
0%{   color: red;   }
47%{   color: #000; }
62%{   color: transparent; }
97%{   color:transparent; }
100%{  color: #000;   }
}

.circle{
  height:300px;
  width: 300px;
  background-color:#828282;
  border-color: red;
  border-radius: 100%;
  margin-left:200px;
  margin-top:250px;

}
```


styles.css — C:\Users\LENOVO\Documents\Virtual Music Instrument — Atom
File Edit View Selection Find Packages Help
Project
Virtual Music Instrument
drumKit
images
Music_player
index.html
styles.css
styles.css
.drumh{
height:250px;
width:350px;
position: absolute;
top:0;
left:0;
margin-top:400px;
margin-left: 100px;
animation:drumm 9s infinite;
animation-timing-function:ease-in;
animation-fill-mode:forwards;
}
@keyframes drumm {
0% {left: 0px;top:0;}
70% {left:80px;top:-80px;height:300px;width:400px;}
100% {left: 0px;top:0;}
}
.guitarh{
height:400px;
width:400px;
position:absolute;
styles.css
CRLF UTF-8 CSS GitHub Git (0) 5 updates
ENG IN
01:38 AM 22-12-2021

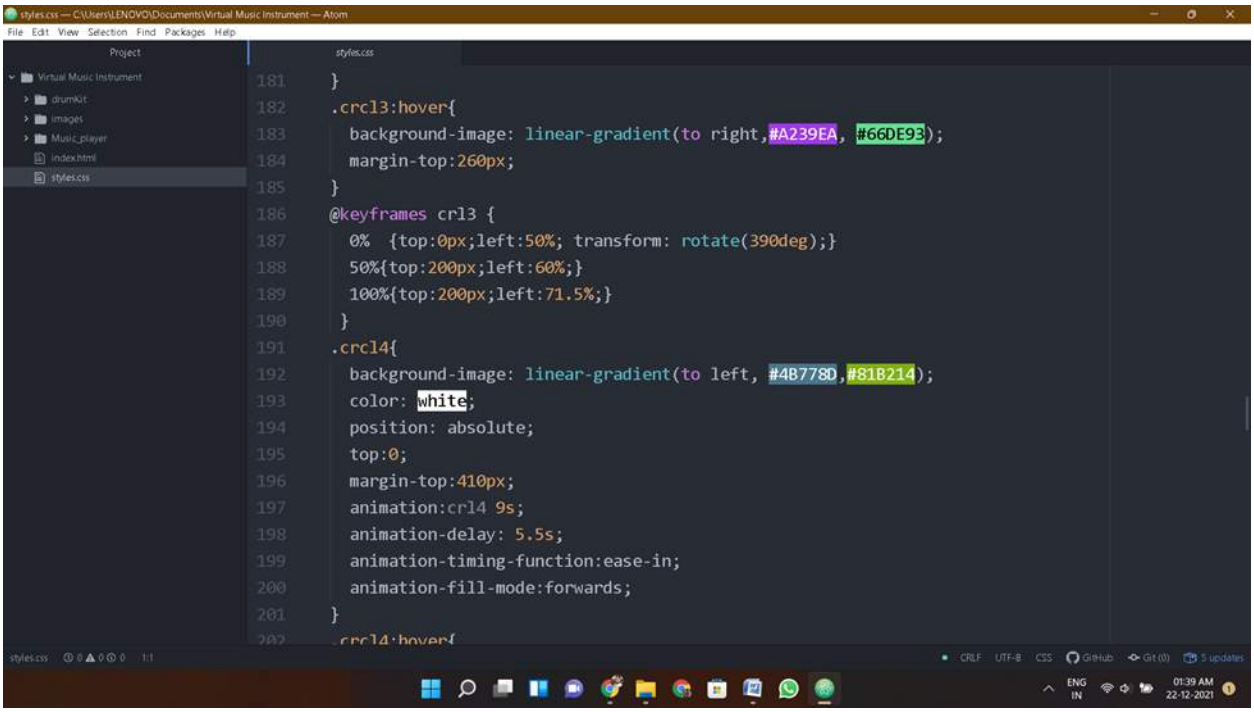

styles.css — C:\Users\LENOVO\Documents\Virtual Music Instrument — Atom
File Edit View Selection Find Packages Help
Project
Virtual Music Instrument
drumKit
images
Music_player
index.html
styles.css
styles.css
@keyframes guitarm {
0% {left: 0px;}
70% {left:300px;}
100% {left: 0px;}
}
.fluteh{
transform: rotate(15deg);
height:200px;
width:300px;
position:absolute;
top:0;
left:0;
margin-top:280px;
margin-left:180px;
animation:flutem 9s infinite;
animation-timing-function:ease-in;
animation-fill-mode:forwards;
}
@keyframes flutem {
0% {top: 0px;}
70% {top:-100px;left:300px; transform: rotate(390deg);}
styles.css
CRLF UTF-8 CSS GitHub Git (0) 5 updates
ENG IN
01:38 AM 22-12-2021



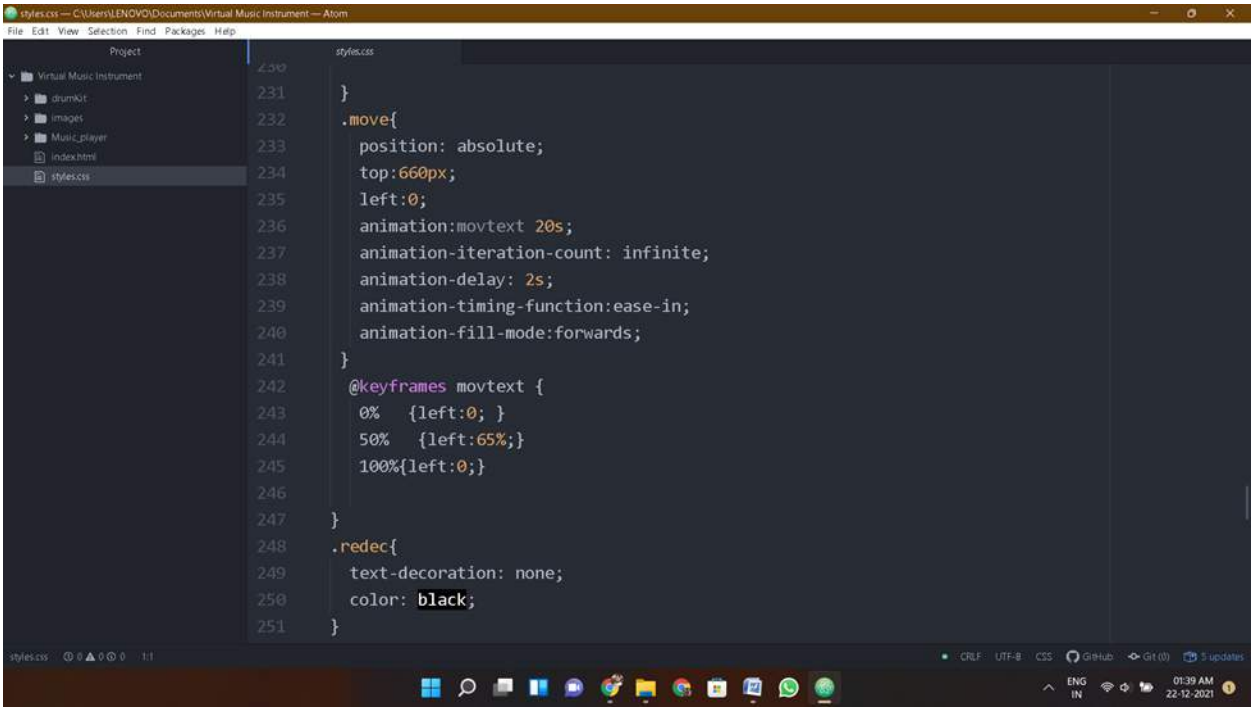
```
.welcome{
  color: white;
  width:80%;
  position: relative;
  left:10%;
  transform: rotate(350deg);
  padding-top:20%;
  font-size: 4rem;
  font-family: 'Black Ops One', cursive;
}
.welcome:hover{
  background: -webkit-linear-gradient(#361c64, #2C394B,#334756,#FF4C29);
  -webkit-background-clip: text;
  -webkit-text-fill-color: transparent;
}
.scircle{
  height:150px;
  width: 150px;
  background-color:black;
  border-radius: 100%;
  text-align: center;
```



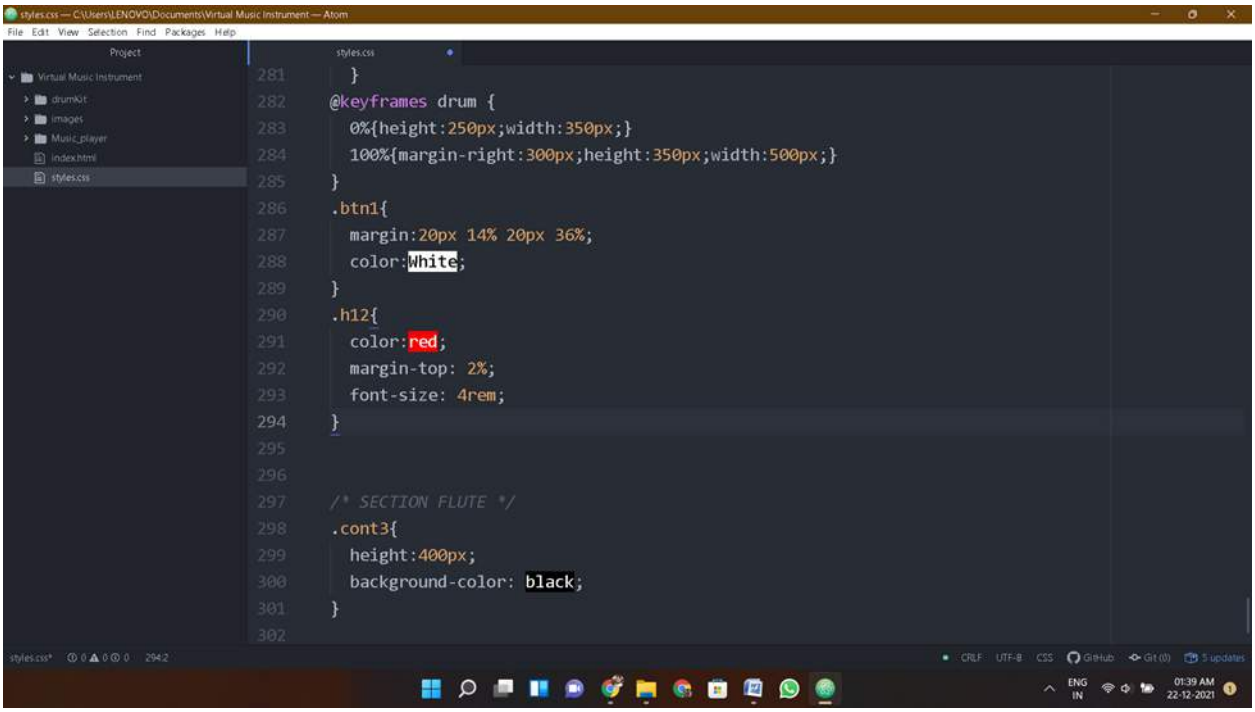
```
.crcl1{
  background-image:linear-gradient(to right,#CD113B,#52006A);
  color: white;
  position: absolute;
  top:0;
  margin-top:100px;
  animation:crl1 9s;
  animation-delay: 5.5s;
  animation-timing-function:ease-in;
  animation-fill-mode:forwards;
}
.crcl1:hover{
  background-image:linear-gradient(to right,#52006A,#CD113B);
  margin-top:50px;
}
@keyframes crl1 {

   0%  {top:0px;left:50%; transform: rotate(390deg);}
   50%{top:410px;left:70%;}
   100%{top:410px;left:90%;}
 }
```



```
}
.crcl3:hover{
  background-image: linear-gradient(to right,#A239EA, #66DE93);
  margin-top:260px;
}
@keyframes crl3 {
  0%  {top:0px;left:50%; transform: rotate(390deg);}
  50%{top:200px;left:60%;}
  100%{top:200px;left:71.5%;}
 }
.crcl4{
  background-image: linear-gradient(to left, #4B778D,#81B214);
  color: white;
  position: absolute;
  top:0;
  margin-top:410px;
  animation:crl4 9s;
  animation-delay: 5.5s;
  animation-timing-function:ease-in;
  animation-fill-mode:forwards;
}
```

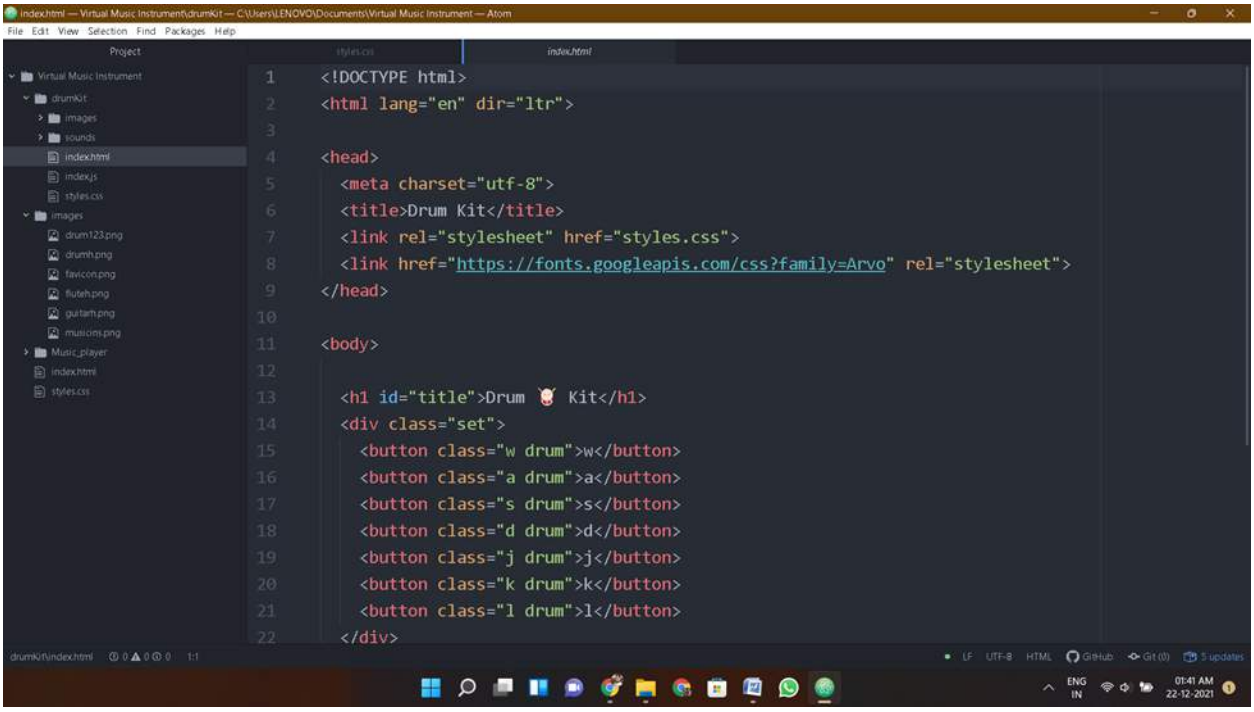


```
}
.move{
  position: absolute;
  top:660px;
  left:0;
  animation:movtext 20s;
  animation-iteration-count: infinite;
  animation-delay: 2s;
  animation-timing-function:ease-in;
  animation-fill-mode:forwards;
}
 @keyframes movtext {
  0%   {left:0; }
  50%   {left:65%;}
  100%{left:0;}

}
.redec{
  text-decoration: none;
  color: black;
}
```

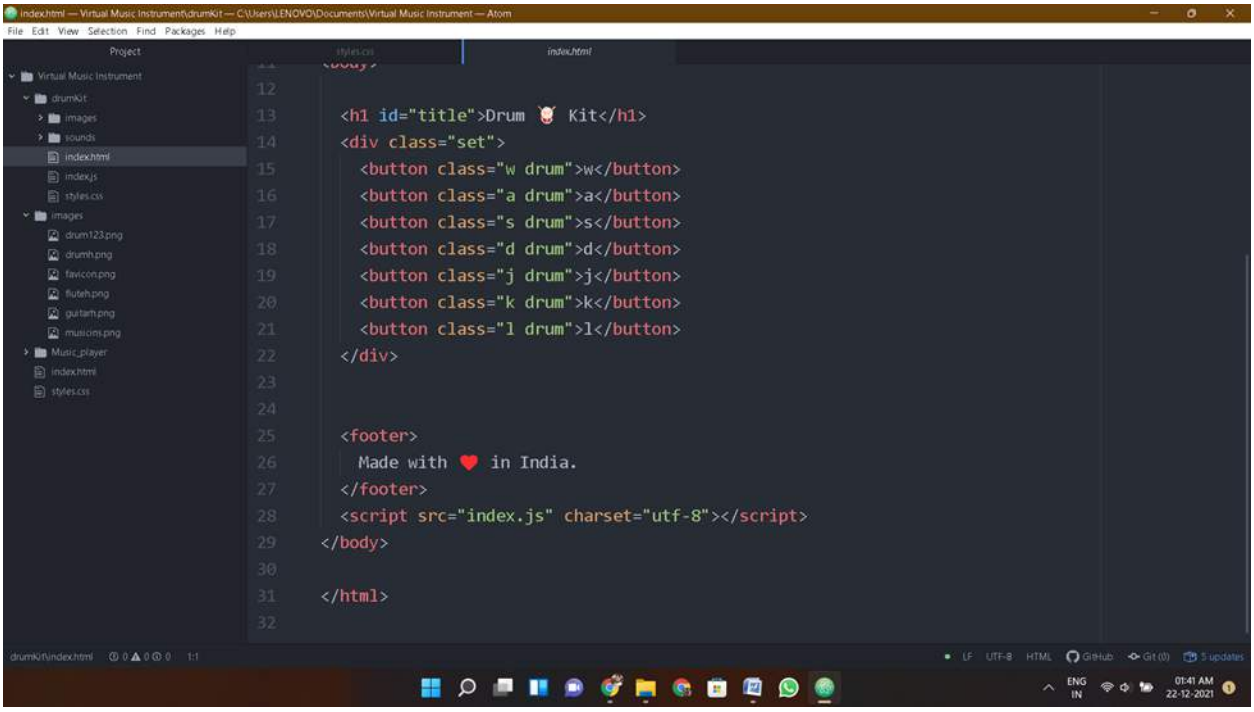

styles.css — C:\Users\LENOVO\Documents\Virtual Music Instrument — Atom
File Edit View Selection Find Packages Help
Project
Virtual Music Instrument
drumKit
images
Music_player
index.html
styles.css
styles.css
281 }
282 @keyframes drum {
283 0%{height:250px;width:350px;}
284 100%{margin-right:300px;height:350px;width:500px;}
285 }
286 .btn1{
287 margin:20px 14% 20px 36%;
288 color:White;
289 }
290 .h12{
291 color:red;
292 margin-top: 2%;
293 font-size: 4rem;
294 }
295
296
297 /* SECTION FLUTE */
298 .cont3{
299 height:400px;
300 background-color: black;
301 }
302
styles.css* 0 0 0 0 294:2
CRLF UTF-8 CSS GitHub Git (0) 5 updates
ENG IN
01:39 AM 22-12-2021

Musical Instruments:

HTML:

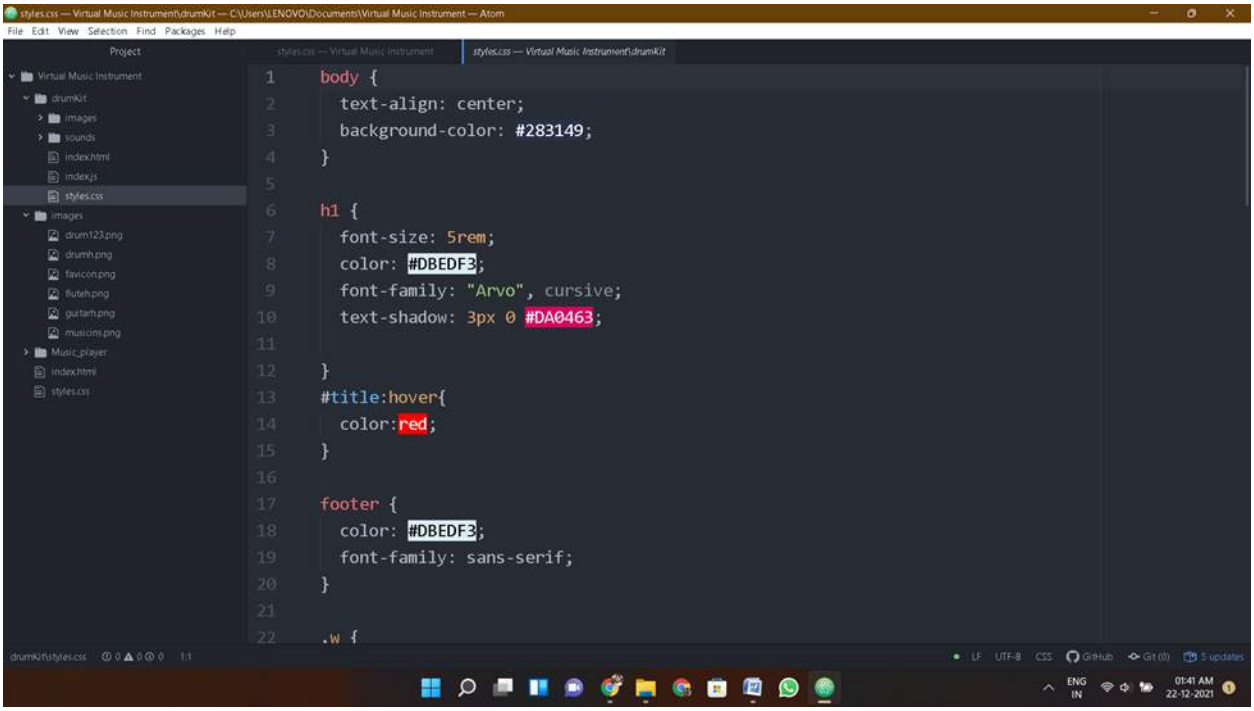


```
<!DOCTYPE html>
<html lang="en" dir="ltr">

<head>
  <meta charset="utf-8">
  <title>Drum Kit</title>
  <link rel="stylesheet" href="styles.css">
  <link href="https://fonts.googleapis.com/css?family=Arvo" rel="stylesheet">
</head>

<body>

  <h1 id="title">Drum 🥁 Kit</h1>
  <div class="set">
    <button class="w drum">w</button>
    <button class="a drum">a</button>
    <button class="s drum">s</button>
    <button class="d drum">d</button>
    <button class="j drum">j</button>
    <button class="k drum">k</button>
    <button class="l drum">l</button>
  </div>
```

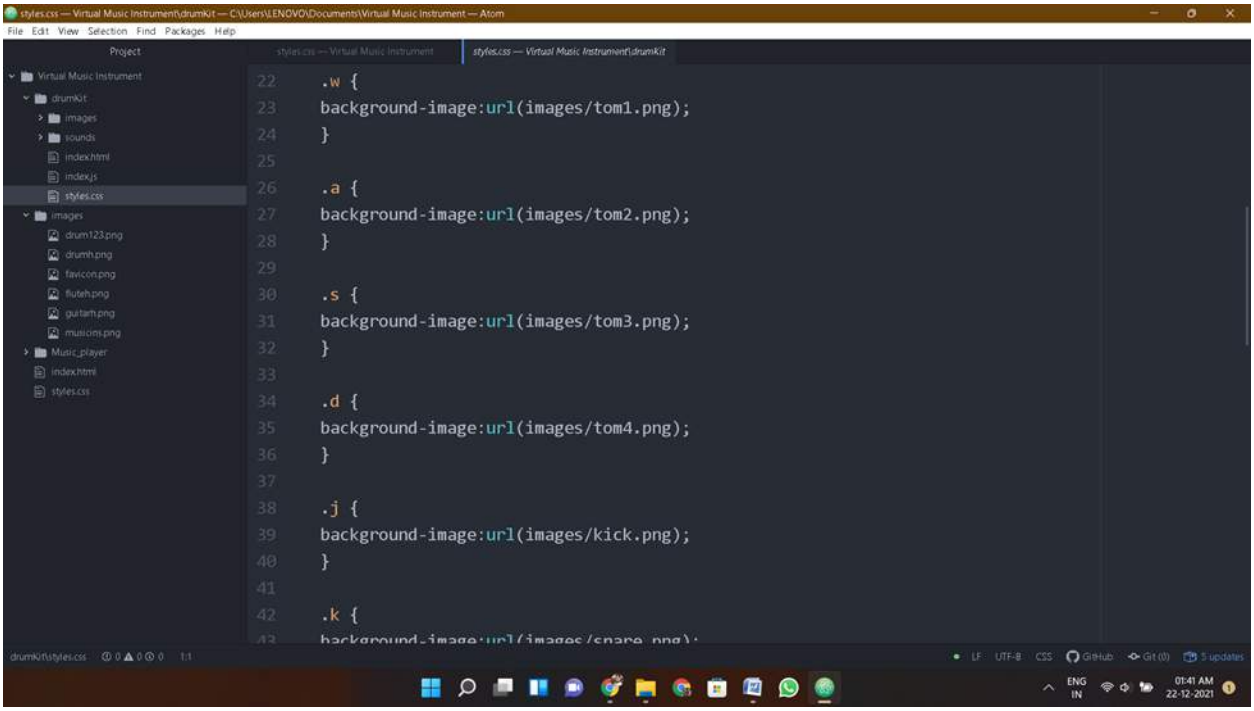


```
  <h1 id="title">Drum 🥁 Kit</h1>
  <div class="set">
    <button class="w drum">w</button>
    <button class="a drum">a</button>
    <button class="s drum">s</button>
    <button class="d drum">d</button>
    <button class="j drum">j</button>
    <button class="k drum">k</button>
    <button class="l drum">l</button>
  </div>


  <footer>
    Made with ❤️ in India.
  </footer>
  <script src="index.js" charset="utf-8"></script>
</body>

</html>
```

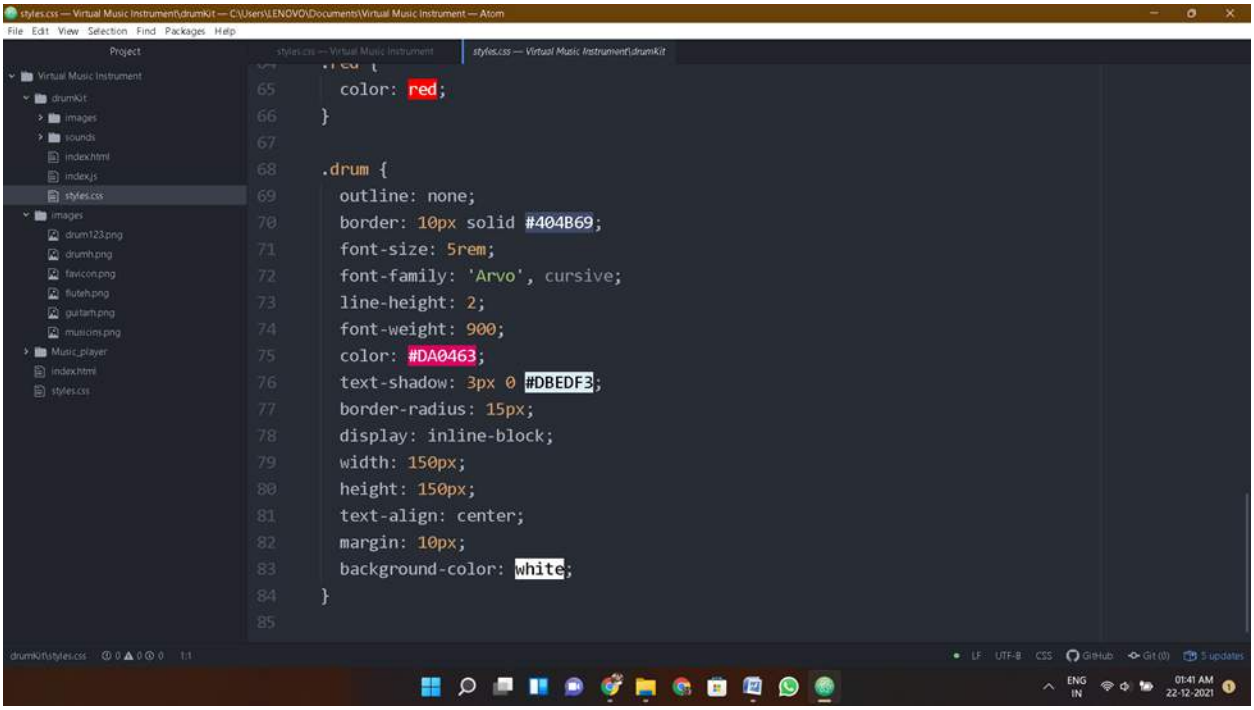
CSS:



```
body {
  text-align: center;
  background-color: #283149;
}

h1 {
  font-size: 5rem;
  color: #DBEDF3;
  font-family: "Arvo", cursive;
  text-shadow: 3px 0 #DA0463;

}
#title:hover{
  color:red;
}

footer {
  color: #DBEDF3;
  font-family: sans-serif;
}

.w {
```

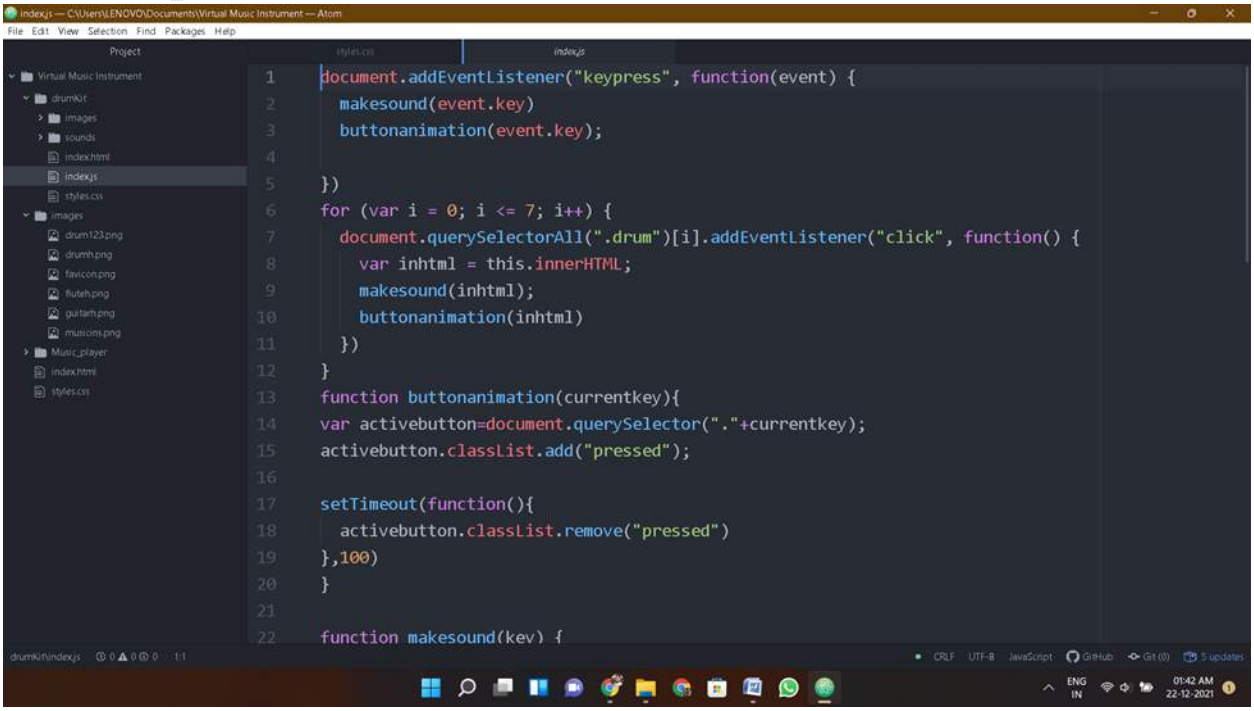


```
.w {
background-image:url(images/tom1.png);
}

.a {
background-image:url(images/tom2.png);
}

.s {
background-image:url(images/tom3.png);
}

.d {
background-image:url(images/tom4.png);
}

.j {
background-image:url(images/kick.png);
}

.k {
```

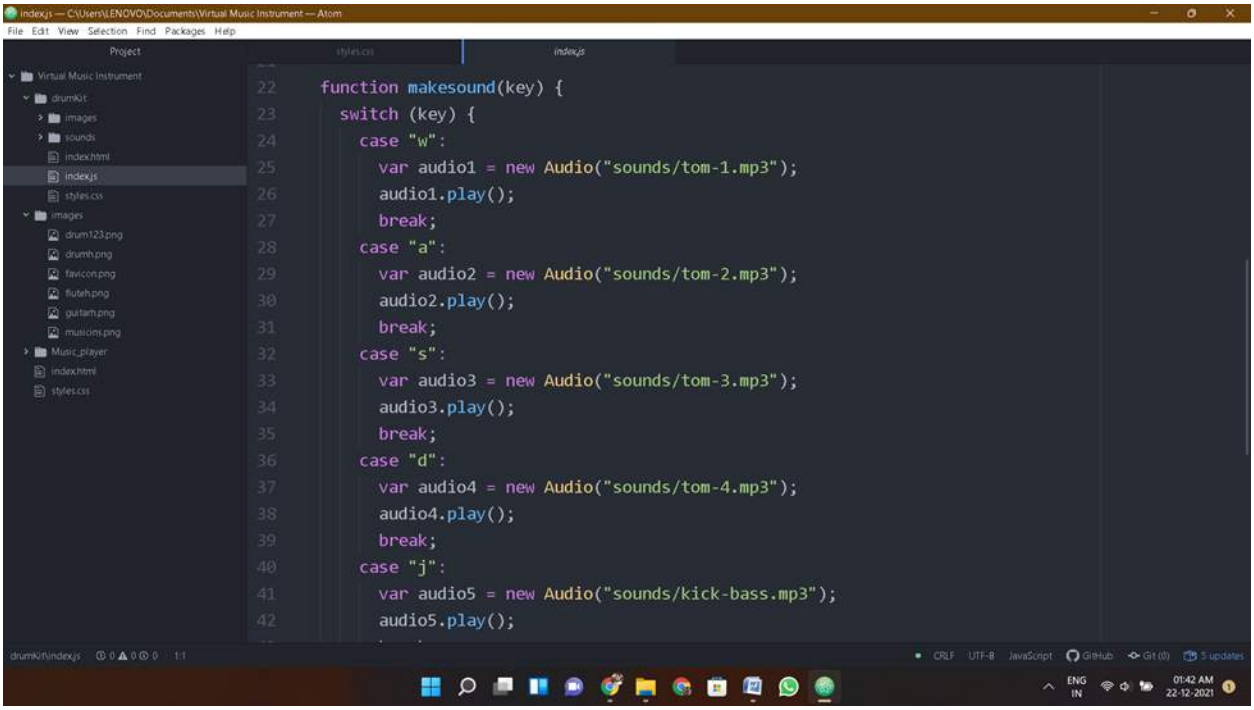


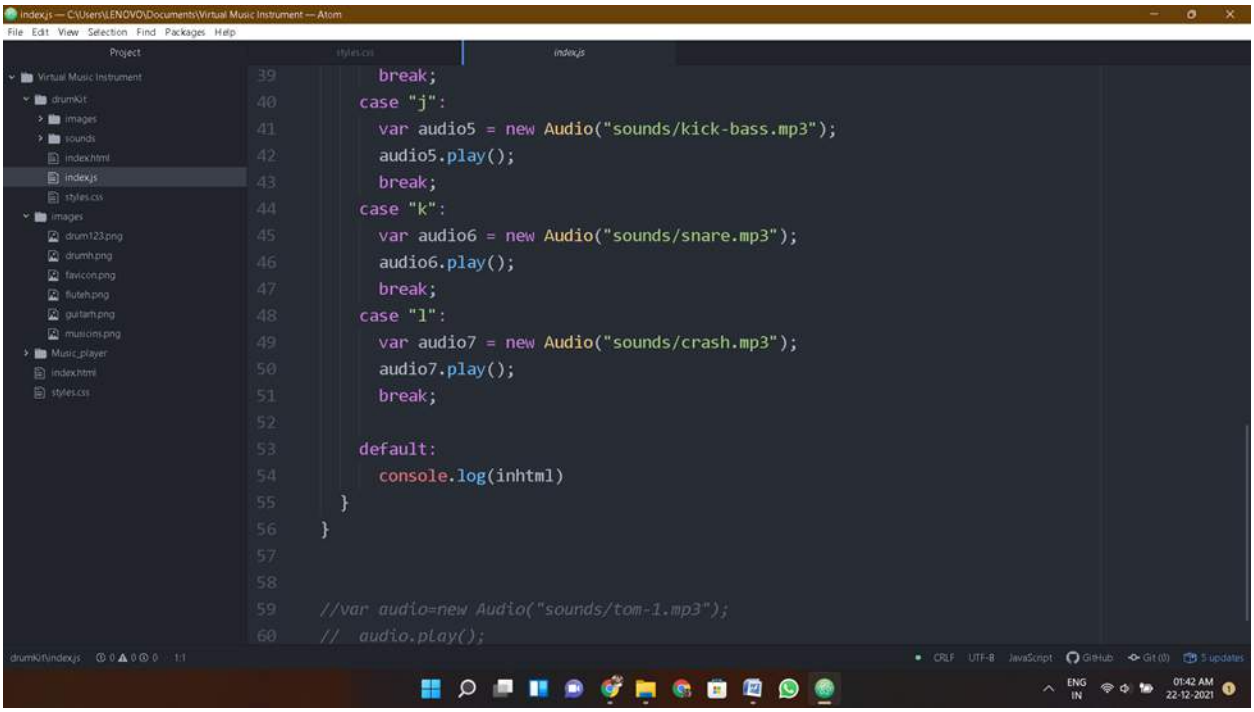
```
  color: red;
}

.drum {
  outline: none;
  border: 10px solid #404B69;
  font-size: 5rem;
  font-family: 'Arvo', cursive;
  line-height: 2;
  font-weight: 900;
  color: #DA0463;
  text-shadow: 3px 0 #DBEDF3;
  border-radius: 15px;
  display: inline-block;
  width: 150px;
  height: 150px;
  text-align: center;
  margin: 10px;
  background-color: white;
}
```

Javascript:



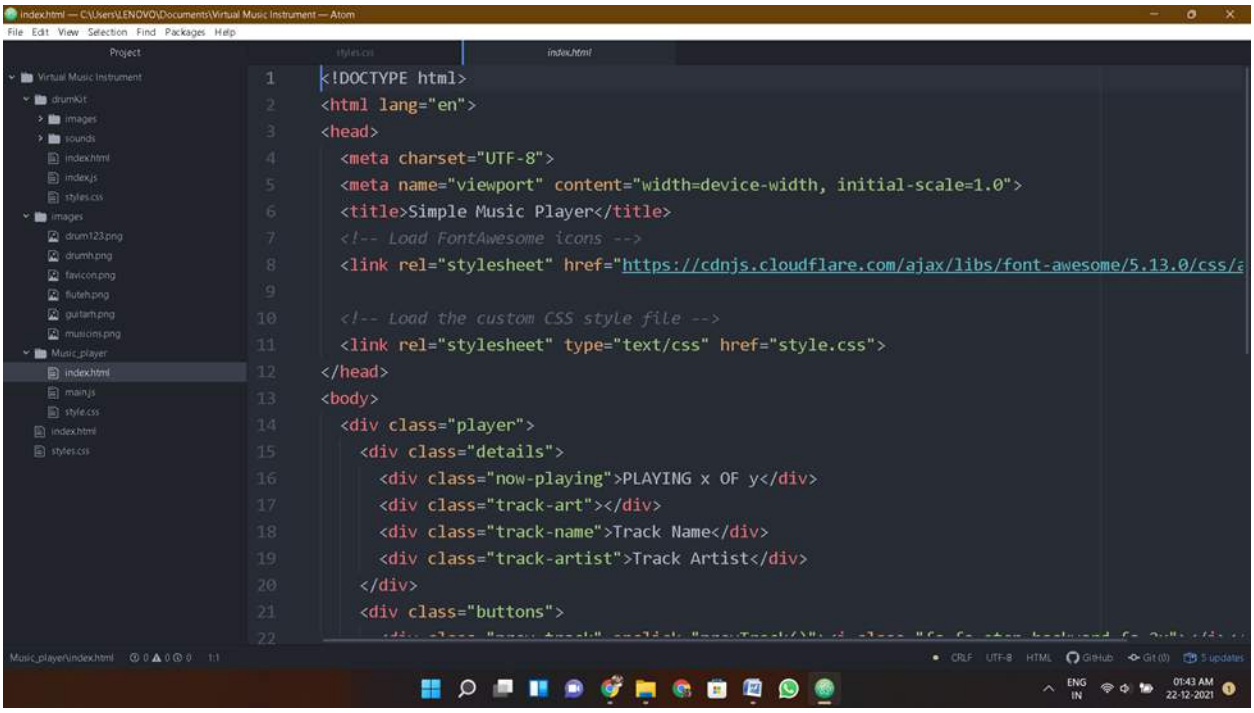
```
document.addEventListener("keypress", function(event) {
  makesound(event.key)
  buttonanimation(event.key);

})
for (var i = 0; i <= 7; i++) {
  document.querySelectorAll(".drum")[i].addEventListener("click", function() {
    var inhtml = this.innerHTML;
    makesound(inhtml);
    buttonanimation(inhtml)
  })
}
function buttonanimation(currentkey){
var activebutton=document.querySelector("."+currentkey);
activebutton.classList.add("pressed");

setTimeout(function(){
  activebutton.classList.remove("pressed")
},100)
}

function makesound(key) {
```

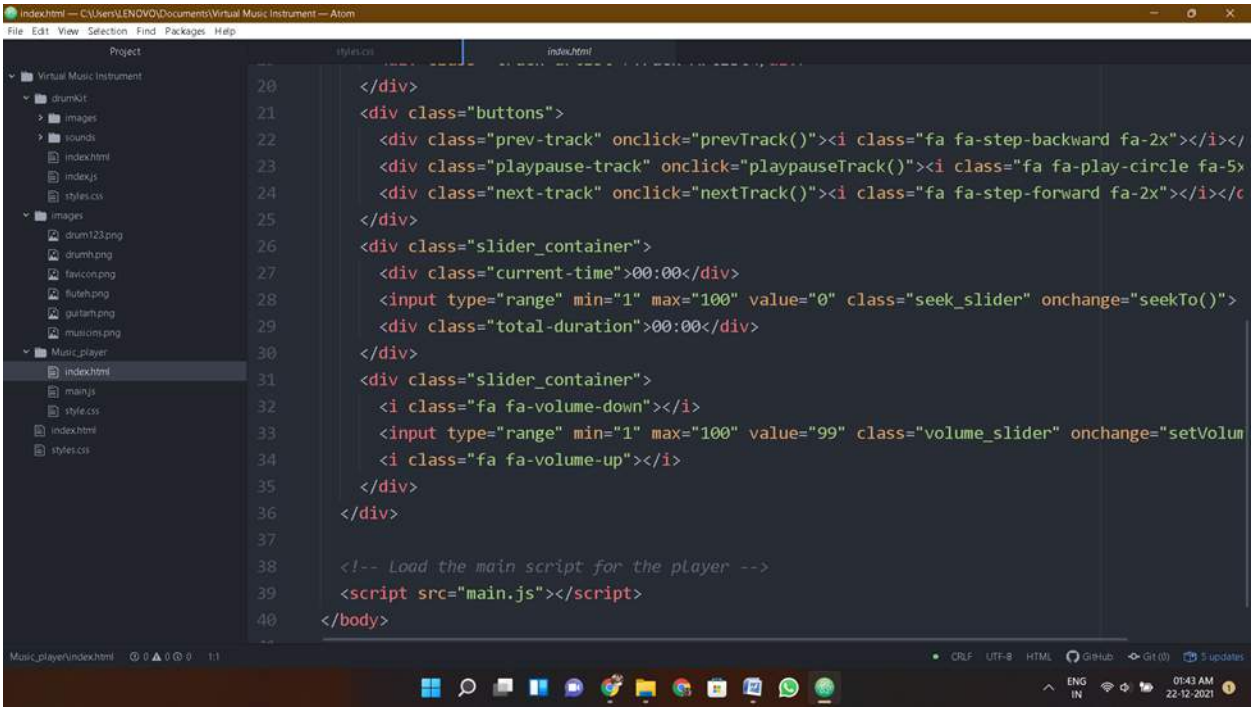

index.js — C:\Users\LENOVO\Documents\Virtual Music Instrument — Atom
File Edit View Selection Find Packages Help
Project
styles.css
index.js
Virtual Music Instrument
drumKit
images
sounds
index.html
index.js
styles.css
images
drum123.png
drumh.png
favicon.png
fluteh.png
guitarh.png
musicins.png
Music_player
index.html
styles.css
function makesound(key) {
switch (key) {
case "w":
var audio1 = new Audio("sounds/tom-1.mp3");
audio1.play();
break;
case "a":
var audio2 = new Audio("sounds/tom-2.mp3");
audio2.play();
break;
case "s":
var audio3 = new Audio("sounds/tom-3.mp3");
audio3.play();
break;
case "d":
var audio4 = new Audio("sounds/tom-4.mp3");
audio4.play();
break;
case "j":
var audio5 = new Audio("sounds/kick-bass.mp3");
audio5.play();
drumKit\index.js
CRLF UTF-8 JavaScript GitHub Git (0) 5 updates
ENG IN
01:42 AM 22-12-2021


index.js — C:\Users\LENOVO\Documents\Virtual Music Instrument — Atom
File Edit View Selection Find Packages Help
Project
styles.css
index.js
Virtual Music Instrument
drumKit
images
sounds
index.html
index.js
styles.css
images
drum123.png
drumh.png
favicon.png
fluteh.png
guitarh.png
musicins.png
Music_player
index.html
styles.css
break;
case "j":
var audio5 = new Audio("sounds/kick-bass.mp3");
audio5.play();
break;
case "k":
var audio6 = new Audio("sounds/snare.mp3");
audio6.play();
break;
case "l":
var audio7 = new Audio("sounds/crash.mp3");
audio7.play();
break;
default:
console.log(inhtml)
}
}
//var audio=new Audio("sounds/tom-1.mp3");
// audio.play();
drumKit\index.js
CRLF UTF-8 JavaScript GitHub Git (0) 5 updates
ENG IN
01:42 AM 22-12-2021

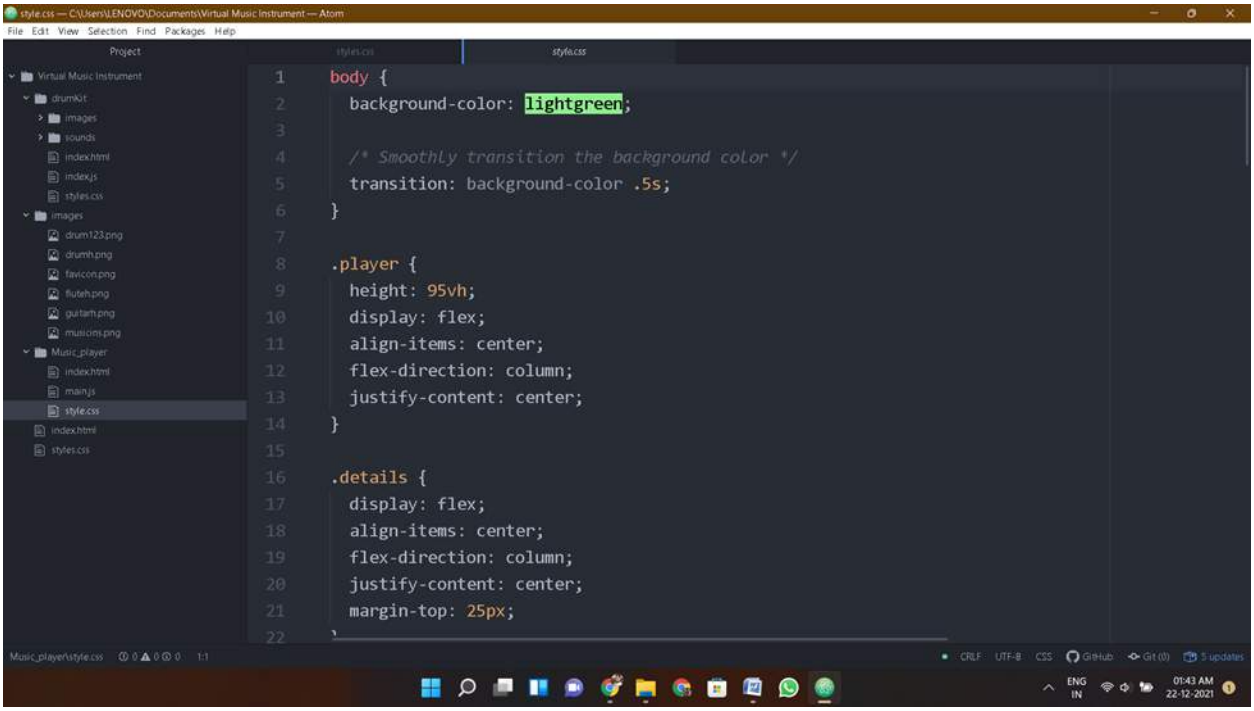
Music Player:

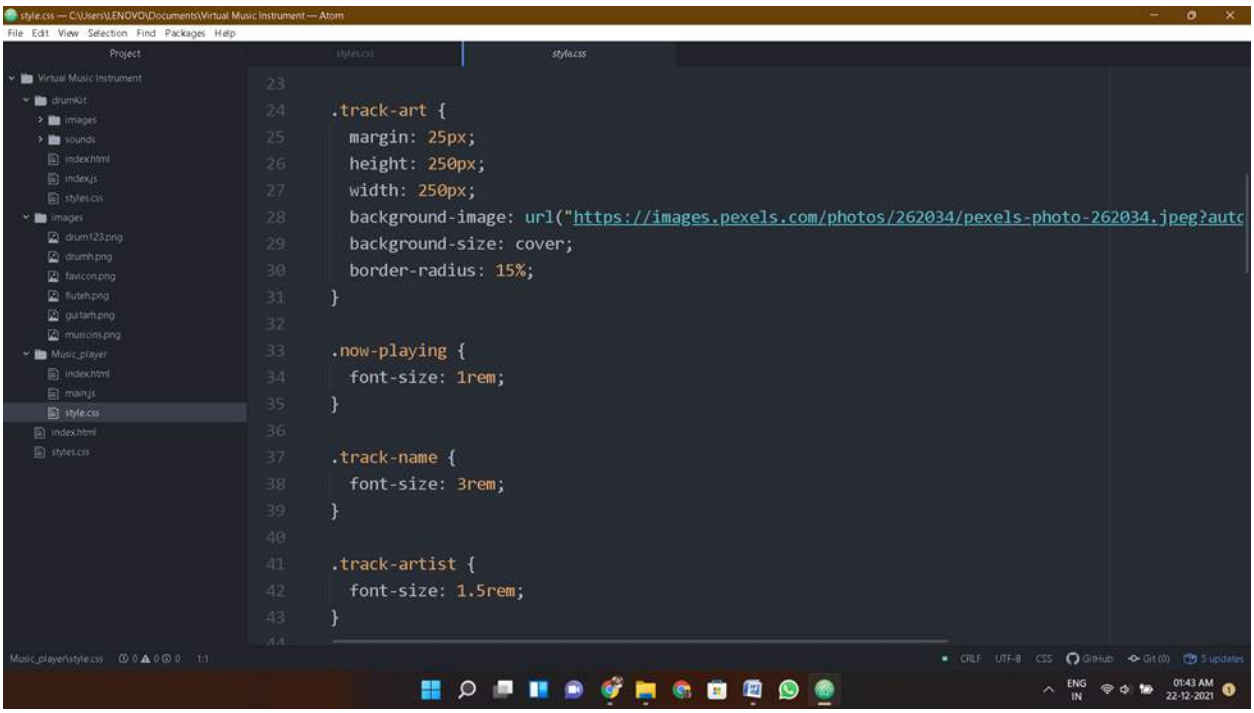
HTML:

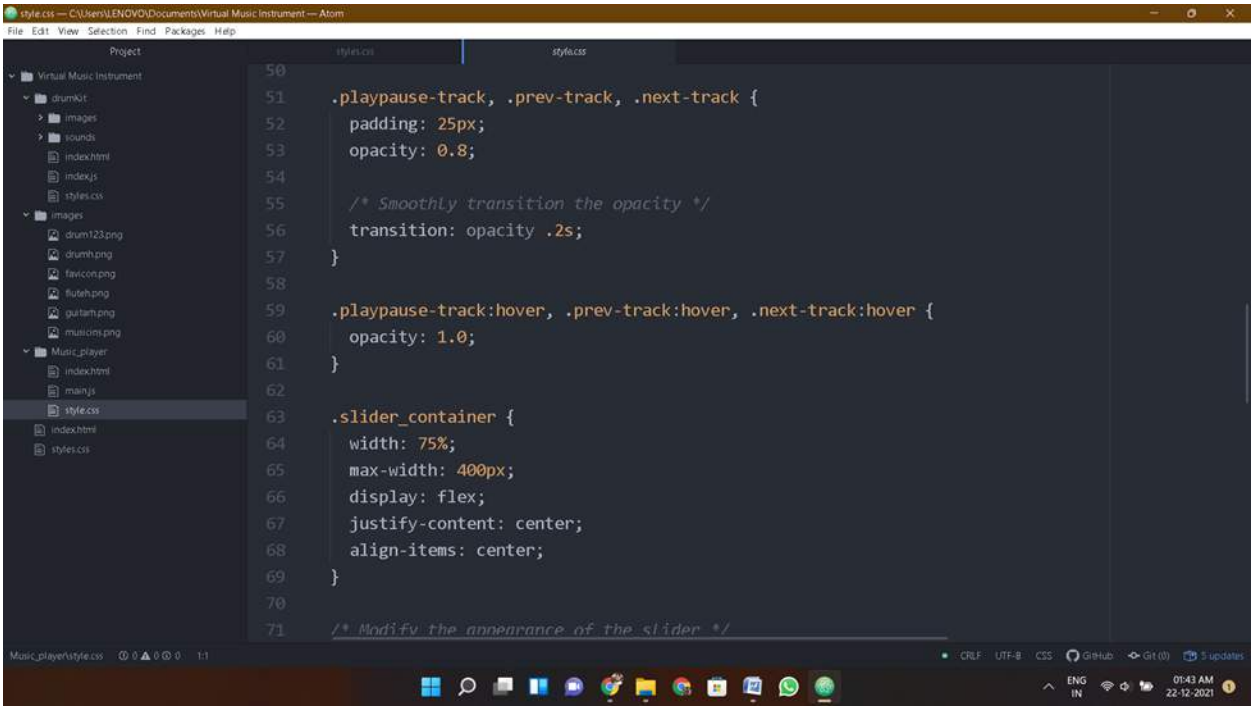


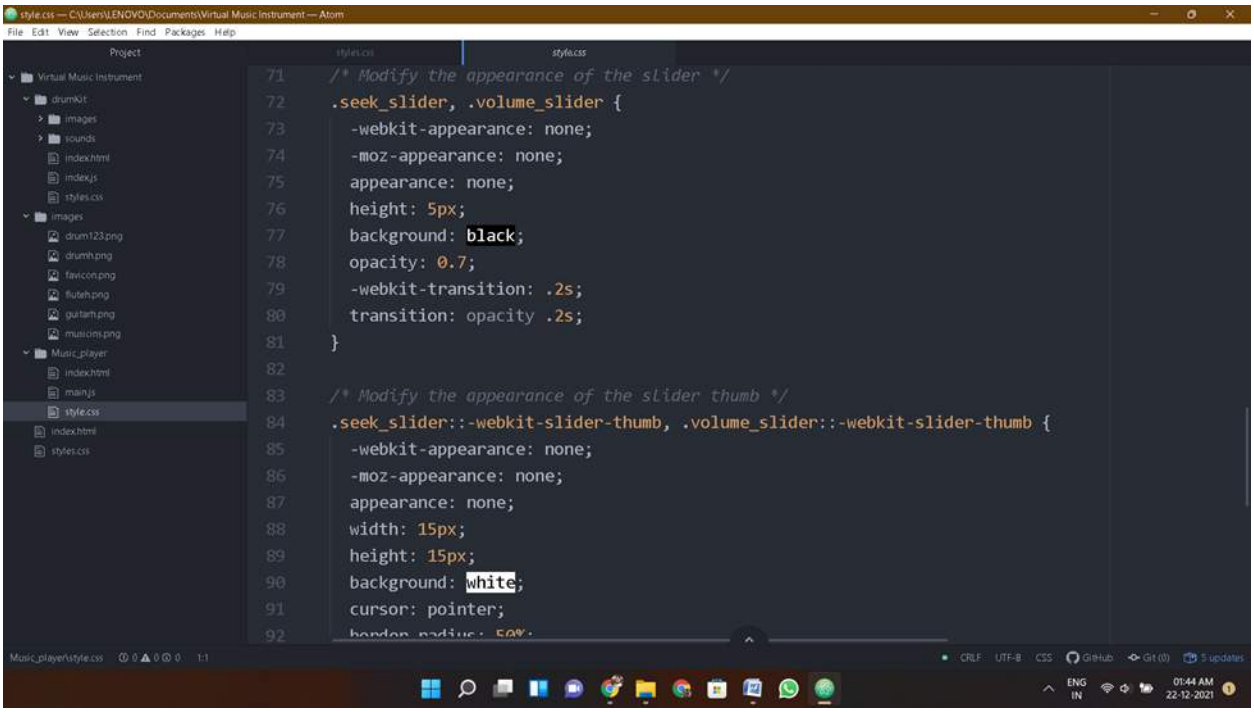


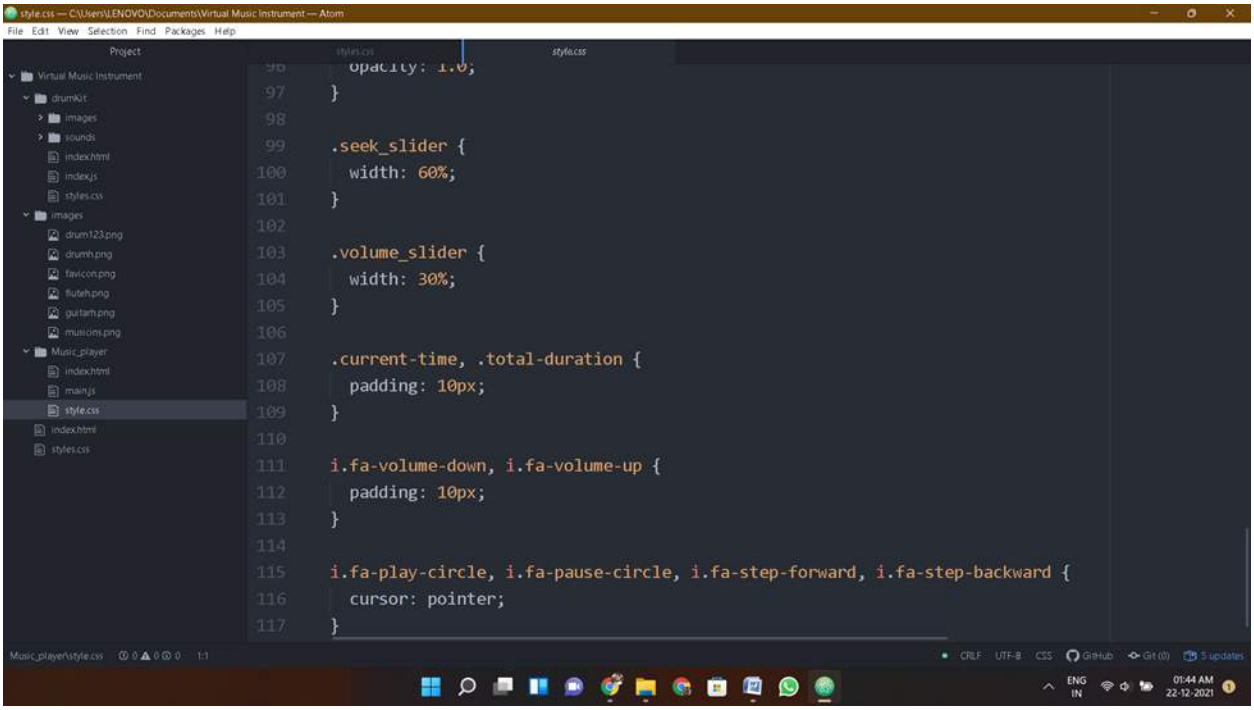
CSS:


style.css — C:\Users\LENOVO\Documents\Virtual Music Instrument — Atom
File Edit View Selection Find Packages Help
Project
Virtual Music Instrument
drumKit
images
sounds
index.html
index.js
styles.css
images
drum123.png
drumh.png
favicon.png
fluteh.png
guitarh.png
musicins.png
Music_player
index.html
main.js
style.css
index.html
styles.css
styles.css
style.css
1 body {
2 background-color: lightgreen;
3
4 /* Smoothly transition the background color */
5 transition: background-color .5s;
6 }
7
8 .player {
9 height: 95vh;
10 display: flex;
11 align-items: center;
12 flex-direction: column;
13 justify-content: center;
14 }
15
16 .details {
17 display: flex;
18 align-items: center;
19 flex-direction: column;
20 justify-content: center;
21 margin-top: 25px;
22 }
Music_player\style.css
CRLF UTF-8 CSS GitHub Git (0) 5 updates
ENG IN
01:43 AM 22-12-2021

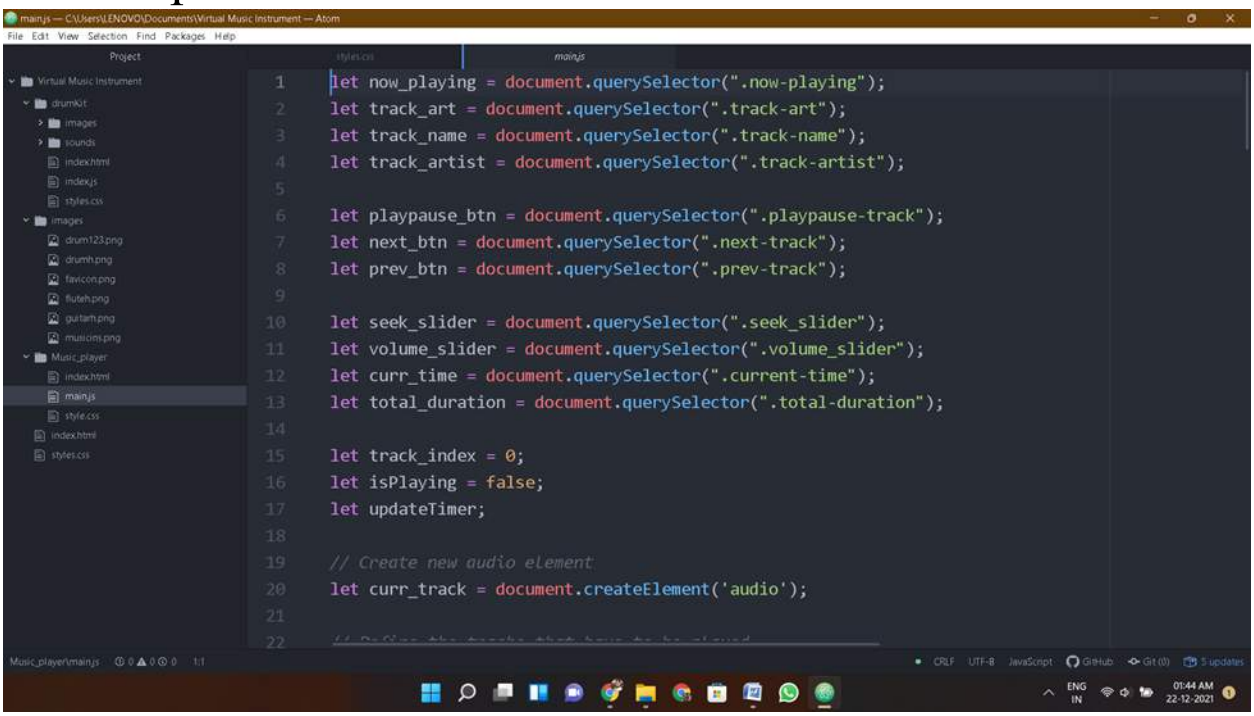

style.css — C:\Users\LENOVO\Documents\Virtual Music Instrument — Atom
File Edit View Selection Find Packages Help
Project
Virtual Music Instrument
drumKit
images
sounds
index.html
index.js
styles.css
images
drum123.png
drumh.png
favicon.png
fluteh.png
guitarh.png
musicins.png
Music_player
index.html
main.js
style.css
index.html
styles.css
styles.css
style.css
23
24 .track-art {
25 margin: 25px;
26 height: 250px;
27 width: 250px;
28 background-image: url("https://images.pexels.com/photos/262034/pexels-photo-262034.jpeg?autc
29 background-size: cover;
30 border-radius: 15%;
31 }
32
33 .now-playing {
34 font-size: 1rem;
35 }
36
37 .track-name {
38 font-size: 3rem;
39 }
40
41 .track-artist {
42 font-size: 1.5rem;
43 }
Music_player\style.css
CRLF UTF-8 CSS GitHub Git (0) 5 updates
ENG IN
01:43 AM 22-12-2021

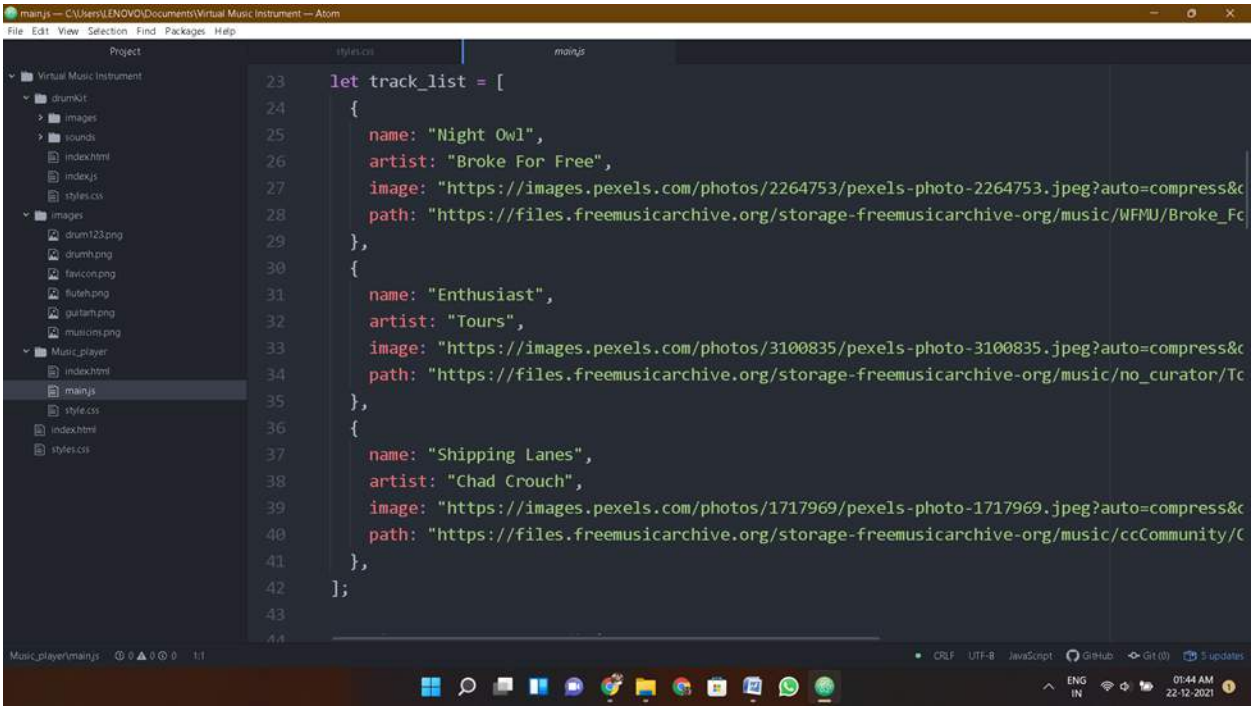

style.css — C:\Users\LENOVO\Documents\Virtual Music Instrument — Atom
File Edit View Selection Find Packages Help
Project
Virtual Music Instrument
drumKit
images
sounds
index.html
index.js
styles.css
images
drum123.png
drumh.png
favicon.png
fluteh.png
guitarh.png
musicins.png
Music_player
index.html
main.js
style.css
index.html
styles.css
styles.css
style.css
.playpause-track, .prev-track, .next-track {
padding: 25px;
opacity: 0.8;
/* Smoothly transition the opacity */
transition: opacity .2s;
}
.playpause-track:hover, .prev-track:hover, .next-track:hover {
opacity: 1.0;
}
.slider_container {
width: 75%;
max-width: 400px;
display: flex;
justify-content: center;
align-items: center;
}
/* Modify the appearance of the slider */
Music_player\style.css
CRLF UTF-8 CSS GitHub Git (0) 5 updates
ENG IN
01:43 AM 22-12-2021


style.css — C:\Users\LENOVO\Documents\Virtual Music Instrument — Atom
File Edit View Selection Find Packages Help
Project
Virtual Music Instrument
drumKit
images
sounds
index.html
index.js
styles.css
images
drum123.png
drumh.png
favicon.png
fluteh.png
guitarh.png
musicins.png
Music_player
index.html
main.js
style.css
index.html
styles.css
styles.css
style.css
/* Modify the appearance of the slider */
.seek_slider, .volume_slider {
-webkit-appearance: none;
-moz-appearance: none;
appearance: none;
height: 5px;
background: black;
opacity: 0.7;
-webkit-transition: .2s;
transition: opacity .2s;
}
/* Modify the appearance of the slider thumb */
.seek_slider::-webkit-slider-thumb, .volume_slider::-webkit-slider-thumb {
-webkit-appearance: none;
-moz-appearance: none;
appearance: none;
width: 15px;
height: 15px;
background: white;
cursor: pointer;
Music_player\style.css
CRLF UTF-8 CSS GitHub Git (0) 5 updates
ENG IN
01:44 AM 22-12-2021

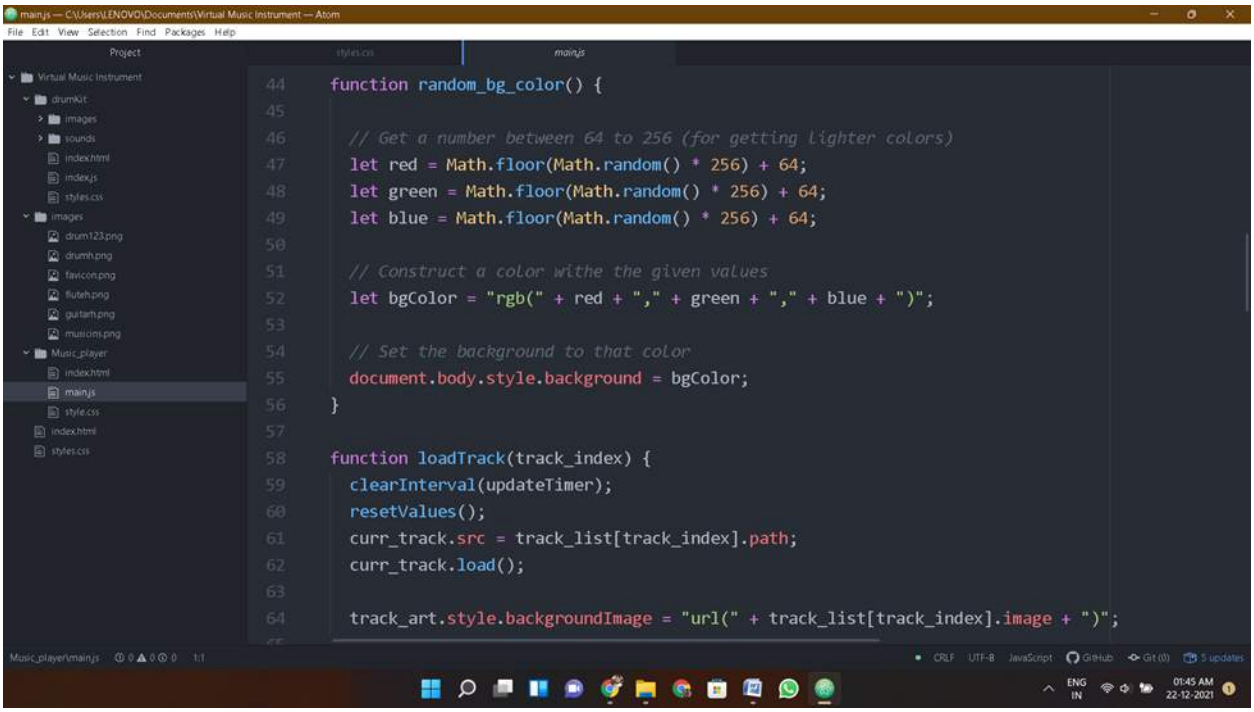

style.css — C:\Users\LENOVO\Documents\Virtual Music Instrument — Atom
File Edit View Selection Find Packages Help
Project
Virtual Music Instrument
drumKit
images
sounds
index.html
index.js
styles.css
images
drum123.png
drumh.png
favicon.png
fluteh.png
guitarh.png
musicins.png
Music_player
index.html
main.js
style.css
index.html
styles.css
styles.css
style.css
opacity: 1.0;
}
.seek_slider {
width: 60%;
}
.volume_slider {
width: 30%;
}
.current-time, .total-duration {
padding: 10px;
}
i.fa-volume-down, i.fa-volume-up {
padding: 10px;
}
i.fa-play-circle, i.fa-pause-circle, i.fa-step-forward, i.fa-step-backward {
cursor: pointer;
}
Music_player\style.css
CRLF UTF-8 CSS GitHub Git (0) 5 updates
ENG IN
01:44 AM 22-12-2021

Javascript:

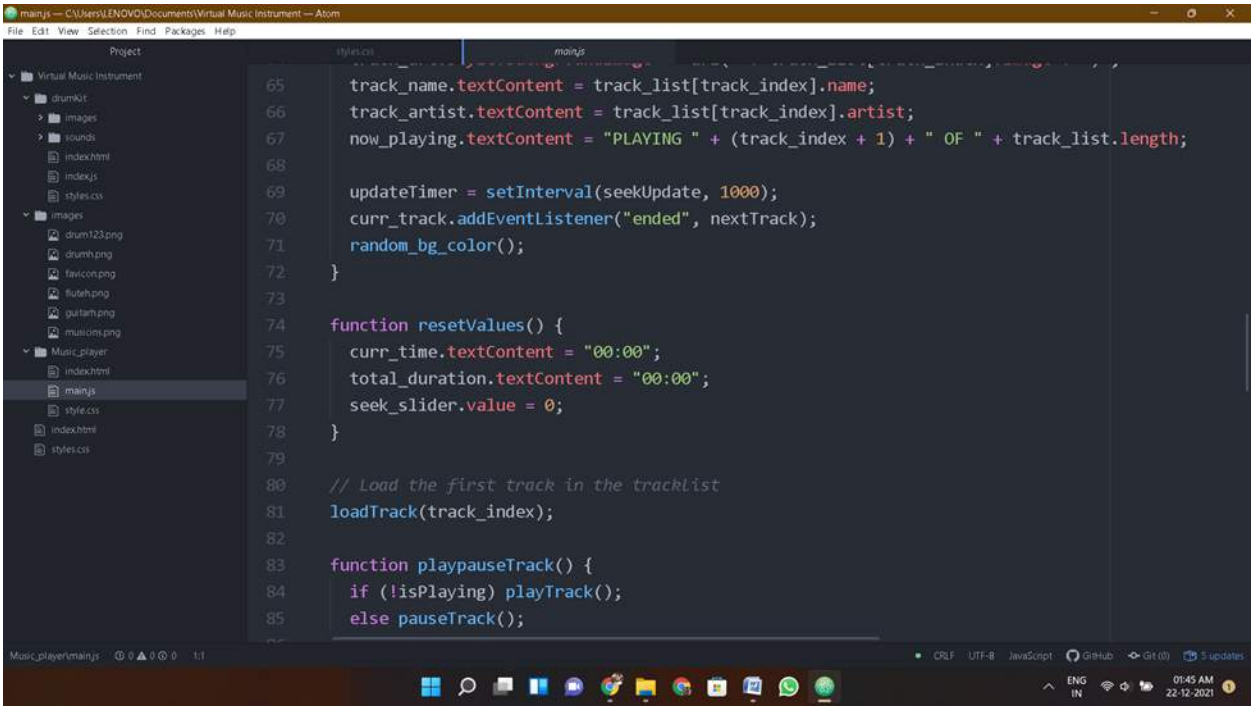

main.js — C:\Users\LENOVO\Documents\Virtual Music Instrument — Atom
File Edit View Selection Find Packages Help
Project
Virtual Music Instrument
drumKit
images
sounds
index.html
index.js
styles.css
images
drum123.png
drumh.png
favicon.png
fluteh.png
guitarh.png
musicins.png
Music_player
index.html
main.js
style.css
index.html
styles.css
styles.css
main.js
let now_playing = document.querySelector(".now-playing");
let track_art = document.querySelector(".track-art");
let track_name = document.querySelector(".track-name");
let track_artist = document.querySelector(".track-artist");
let playpause_btn = document.querySelector(".playpause-track");
let next_btn = document.querySelector(".next-track");
let prev_btn = document.querySelector(".prev-track");
let seek_slider = document.querySelector(".seek_slider");
let volume_slider = document.querySelector(".volume_slider");
let curr_time = document.querySelector(".current-time");
let total_duration = document.querySelector(".total-duration");
let track_index = 0;
let isPlaying = false;
let updateTimer;
// Create new audio element
let curr_track = document.createElement('audio');
Music_player\main.js
CRLF UTF-8 JavaScript GitHub Git (0) 5 updates
ENG IN
01:44 AM 22-12-2021

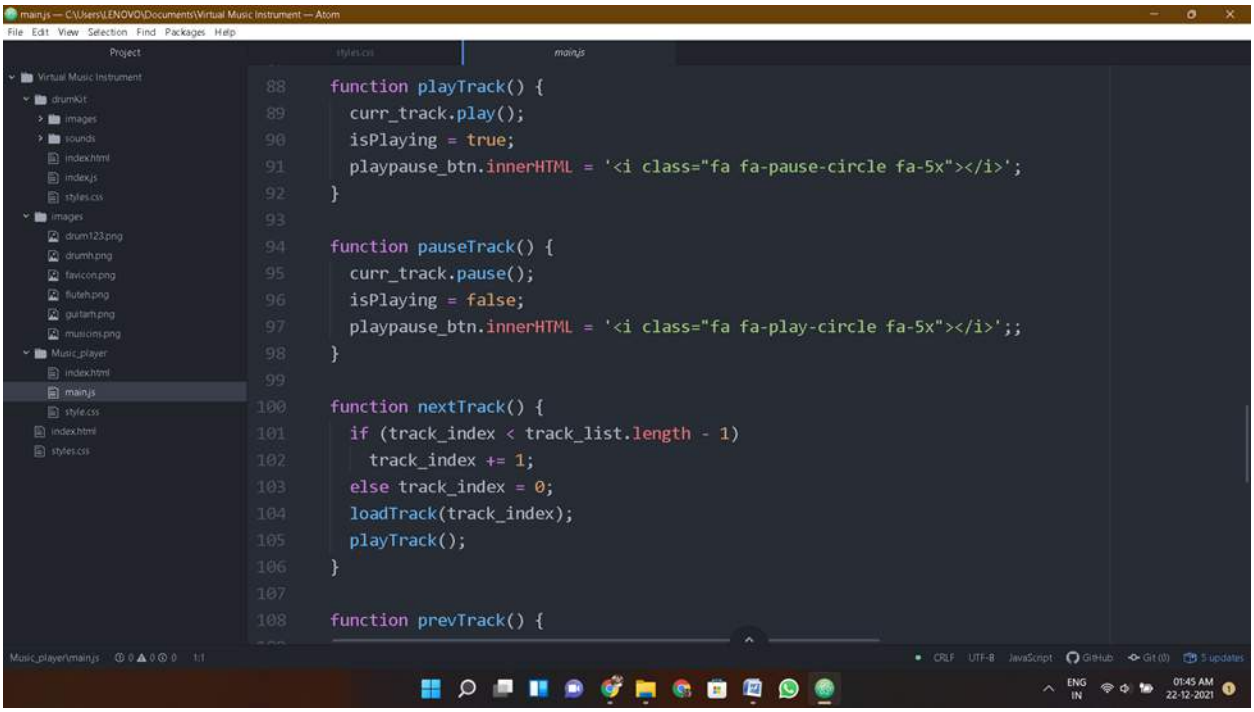


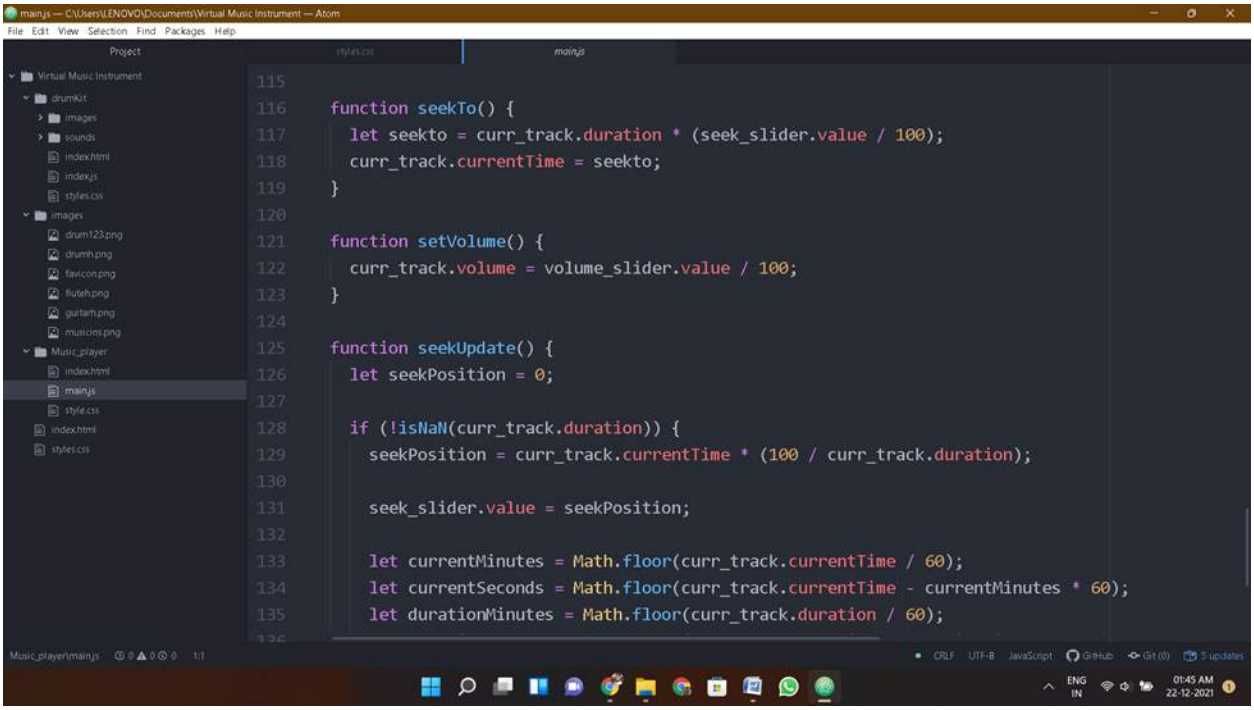
```
let track_list = [
  {
    name: "Night Owl",
    artist: "Broke For Free",
    image: "https://images.pexels.com/photos/2264753/pexels-photo-2264753.jpeg?auto=compress&c
    path: "https://files.freemusicarchive.org/storage-freemusicarchive-org/music/WFMU/Broke_Fc
  },
  {
    name: "Enthusiast",
    artist: "Tours",
    image: "https://images.pexels.com/photos/3100835/pexels-photo-3100835.jpeg?auto=compress&c
    path: "https://files.freemusicarchive.org/storage-freemusicarchive-org/music/no_curator/Tc
  },
  {
    name: "Shipping Lanes",
    artist: "Chad Crouch",
    image: "https://images.pexels.com/photos/1717969/pexels-photo-1717969.jpeg?auto=compress&c
    path: "https://files.freemusicarchive.org/storage-freemusicarchive-org/music/ccCommunity/C
  },
];
```

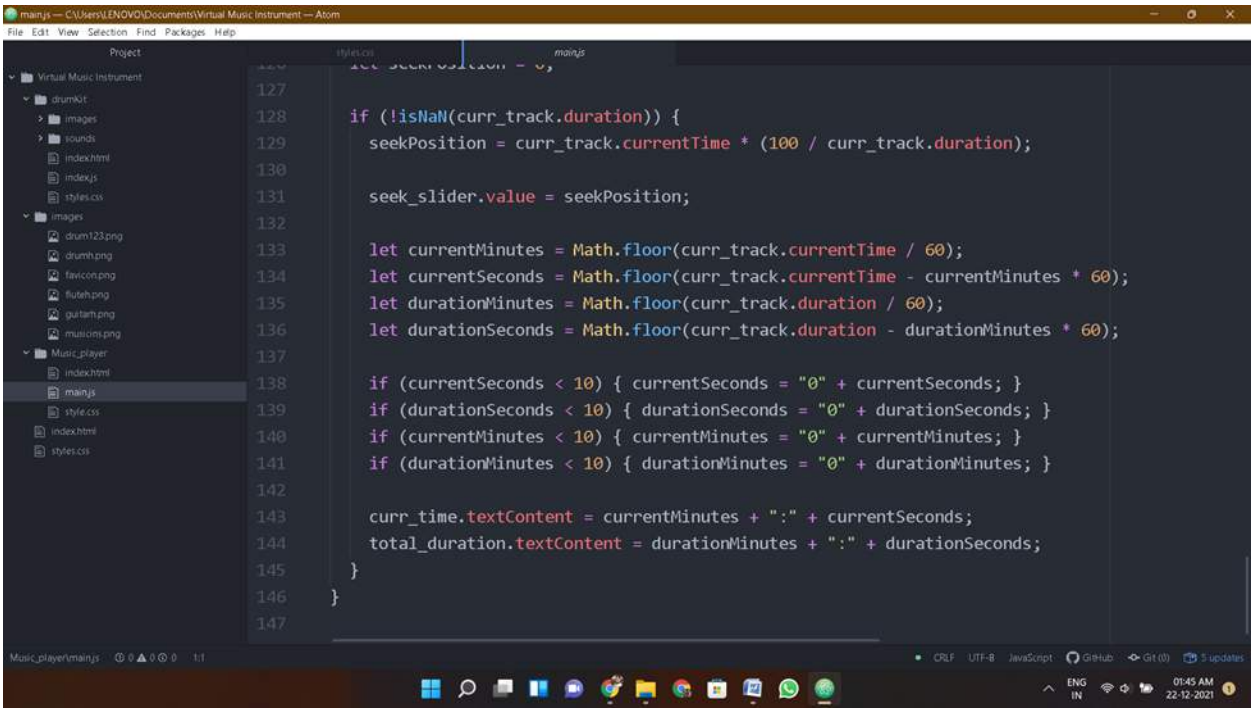


```
function random_bg_color() {

  // Get a number between 64 to 256 (for getting lighter colors)
  let red = Math.floor(Math.random() * 256) + 64;
  let green = Math.floor(Math.random() * 256) + 64;
  let blue = Math.floor(Math.random() * 256) + 64;

  // Construct a color withe the given values
  let bgColor = "rgb(" + red + "," + green + "," + blue + ")";

  // Set the background to that color
  document.body.style.background = bgColor;
}

function loadTrack(track_index) {
  clearInterval(updateTimer);
  resetValues();
  curr_track.src = track_list[track_index].path;
  curr_track.load();

  track_art.style.backgroundImage = "url(" + track_list[track_index].image + ")";
```


main.js — C:\Users\LENOVO\Documents\Virtual Music Instrument — Atom
File Edit View Selection Find Packages Help
Project
Virtual Music Instrument
drumKit
images
sounds
index.html
index.js
styles.css
images
drum123.png
drumh.png
favicon.png
fluteh.png
guitarh.png
musicins.png
Music_player
index.html
main.js
style.css
index.html
styles.css
styles.css
main.js
track_name.textContent = track_list[track_index].name;
track_artist.textContent = track_list[track_index].artist;
now_playing.textContent = "PLAYING " + (track_index + 1) + " OF " + track_list.length;
updateTimer = setInterval(seekUpdate, 1000);
curr_track.addEventListener("ended", nextTrack);
random_bg_color();
}
function resetValues() {
curr_time.textContent = "00:00";
total_duration.textContent = "00:00";
seek_slider.value = 0;
}
// Load the first track in the tracklist
loadTrack(track_index);
function playpauseTrack() {
if (!isPlaying) playTrack();
else pauseTrack();
Music_player\main.js
CRLF UTF-8 JavaScript GitHub Git (0) 5 updates
ENG IN
01:45 AM 22-12-2021


main.js — C:\Users\LENOVO\Documents\Virtual Music Instrument — Atom
File Edit View Selection Find Packages Help
Project
Virtual Music Instrument
drumKit
images
sounds
index.html
index.js
styles.css
images
drum123.png
drumh.png
favicon.png
fluteh.png
guitarh.png
musicins.png
Music_player
index.html
main.js
style.css
index.html
styles.css
styles.css
main.js
function playTrack() {
curr_track.play();
isPlaying = true;
playpause_btn.innerHTML = '';
}
function pauseTrack() {
curr_track.pause();
isPlaying = false;
playpause_btn.innerHTML = '';;
}
function nextTrack() {
if (track_index < track_list.length - 1)
track_index += 1;
else track_index = 0;
loadTrack(track_index);
playTrack();
}
function prevTrack() {
Music_player\main.js
CRLF UTF-8 JavaScript GitHub Git (0) 5 updates
ENG IN
01:45 AM 22-12-2021


main.js — C:\Users\LENOVO\Documents\Virtual Music Instrument — Atom
File Edit View Selection Find Packages Help
Project
styles.css
main.js
Virtual Music Instrument
drumKit
images
sounds
index.html
index.js
styles.css
images
drum123.png
drumh.png
favicon.png
fluteh.png
guitarh.png
musicins.png
Music_player
index.html
main.js
style.css
index.html
styles.css
115
116 function seekTo() {
117 let seekto = curr_track.duration * (seek_slider.value / 100);
118 curr_track.currentTime = seekto;
119 }
120
121 function setVolume() {
122 curr_track.volume = volume_slider.value / 100;
123 }
124
125 function seekUpdate() {
126 let seekPosition = 0;
127
128 if (!isNaN(curr_track.duration)) {
129 seekPosition = curr_track.currentTime * (100 / curr_track.duration);
130
131 seek_slider.value = seekPosition;
132
133 let currentMinutes = Math.floor(curr_track.currentTime / 60);
134 let currentSeconds = Math.floor(curr_track.currentTime - currentMinutes * 60);
135 let durationMinutes = Math.floor(curr_track.duration / 60);
Music_player\main.js
CRLF UTF-8 JavaScript GitHub Git (0) 5 updates
ENG IN
01:45 AM 22-12-2021


main.js — C:\Users\LENOVO\Documents\Virtual Music Instrument — Atom
File Edit View Selection Find Packages Help
Project
styles.css
main.js
Virtual Music Instrument
drumKit
images
sounds
index.html
index.js
styles.css
images
drum123.png
drumh.png
favicon.png
fluteh.png
guitarh.png
musicins.png
Music_player
index.html
main.js
style.css
index.html
styles.css
127
128 if (!isNaN(curr_track.duration)) {
129 seekPosition = curr_track.currentTime * (100 / curr_track.duration);
130
131 seek_slider.value = seekPosition;
132
133 let currentMinutes = Math.floor(curr_track.currentTime / 60);
134 let currentSeconds = Math.floor(curr_track.currentTime - currentMinutes * 60);
135 let durationMinutes = Math.floor(curr_track.duration / 60);
136 let durationSeconds = Math.floor(curr_track.duration - durationMinutes * 60);
137
138 if (currentSeconds < 10) { currentSeconds = "0" + currentSeconds; }
139 if (durationSeconds < 10) { durationSeconds = "0" + durationSeconds; }
140 if (currentMinutes < 10) { currentMinutes = "0" + currentMinutes; }
141 if (durationMinutes < 10) { durationMinutes = "0" + durationMinutes; }
142
143 curr_time.textContent = currentMinutes + ":" + currentSeconds;
144 total_duration.textContent = durationMinutes + ":" + durationSeconds;
145 }
146 }
147
Music_player\main.js
CRLF UTF-8 JavaScript GitHub Git (0) 5 updates
ENG IN
01:45 AM 22-12-2021

Chapter-5 Conclusion/ Future work:

Conclusion

We have designed, initiated, and explored a virtual musical instrument system. Our system is low-cost and the user needs to just visit the webpage and play each and every instrument he/she wishes to play.The latency of the process is small, and the sound output is real-time, though there is still room for improvement.

This website can also be very useful for the teachers who teach music to the students virtually so the beginner students can practice the music through it as well.

So we are working on that to improve the quality of sound produced by different musical instruments and also add more songs to the musical players also.

Future work:

In future this can be enhanced and various new features can be added to it. Different varieties of musical instruments can also be added such as the instruments that were famous of their times and became extinct after new instruments were introduced in the music industry. Some instruments that were used in old times such as Mayuri,Nagfeni,Ektara,Tenor cornett,chime bells,copper serpent and many more the sound of these musical instruments can be researched and added to the website so that people can gain and hear the sound of all the musical instruments that existed in the world. In music players, more number of songs with different artists can be added and by using various machine learning algorithms such as the recommender algorithm by detecting the type of song and music heard by the user many times the similar kind of music can be suggested to the user.

References:

1.Complete Web developer bootcamp by udemy

2.HTML CSS javascript course by coursera.

3.HTML and CSS:Design and build website by john Duckett.

4.Learning Web design:A beginner guide to HTML CSS and Javascript and web graphics by Jennifer Niderst Robbins.