BIRLA INSTITUTE OF TECHNOLOGY, PATNA

SUMMER TRANING PROJECT REPORT

Submitted By-

SHUBHAM SOURABH
BTECH-15044-18

CSE-6TH SEMESTER

WORKING UNDER GUIDANCE OF:

MR. S.K CHATTERJEE

Contents:-

Objective:-

To design a full-fledged dynamic web-Music Player.

Motivation:-

I love listening to music since I was a kid . I believe music has the power to unite people, make us feel at peace, make us feel understood; it is something to dance to, bond over, and even listen to when alone. Music is not just sound, it is its own language and it communicates so much; it is a beautiful thing.

Introduction:-

- This Music Player App is a web application that is made using HTML,CSS,JS,Bootstrap for Front-end part and for Backend Django is used.

- The sole purpose for making this app was to provide user to store their own kind or taste of music or songs.

- User can save and store their music through backend page powered by Django.

- Django's makes it so easy to work with a database along with the views (i.e the functional backend of the app) and the template files all connected with the Django MVT (i.e Model View Template) architecture.

Tools & Technologies used:-

1.) **Bootstrap:-** Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. Bootstrap is a HTML, CSS & JS Library that focuses on simplifying the development of informative web pages. The primary purpose of adding it to a web project is to apply Bootstrap's choices of color, size, font and layout to that project.

2.) **Django:-** Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design.Django can be (and has been) used to build almost any type of website — from content management systems and wikis, through to social networks and news sites. It can work with any client-side framework, and can deliver content in almost any format (including HTML, RSS feeds, JSON, XML, etc).It is very secure as Django enables protection against many vulnerabilities by default, including SQL injection, cross-site scripting, cross-site request forgery and clickjacking.

Design & Implementation:-

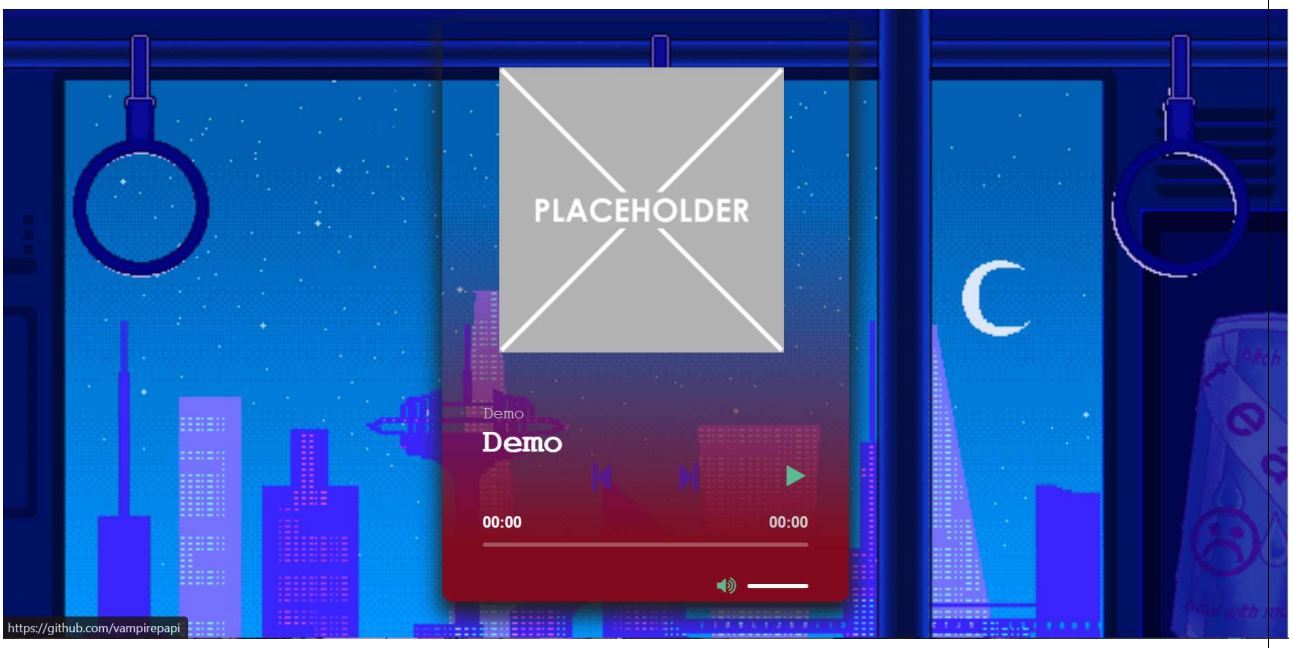


Fig: final design when app launches

Now heading to admin page to add our song in the database.

For login Django authentication system is used.

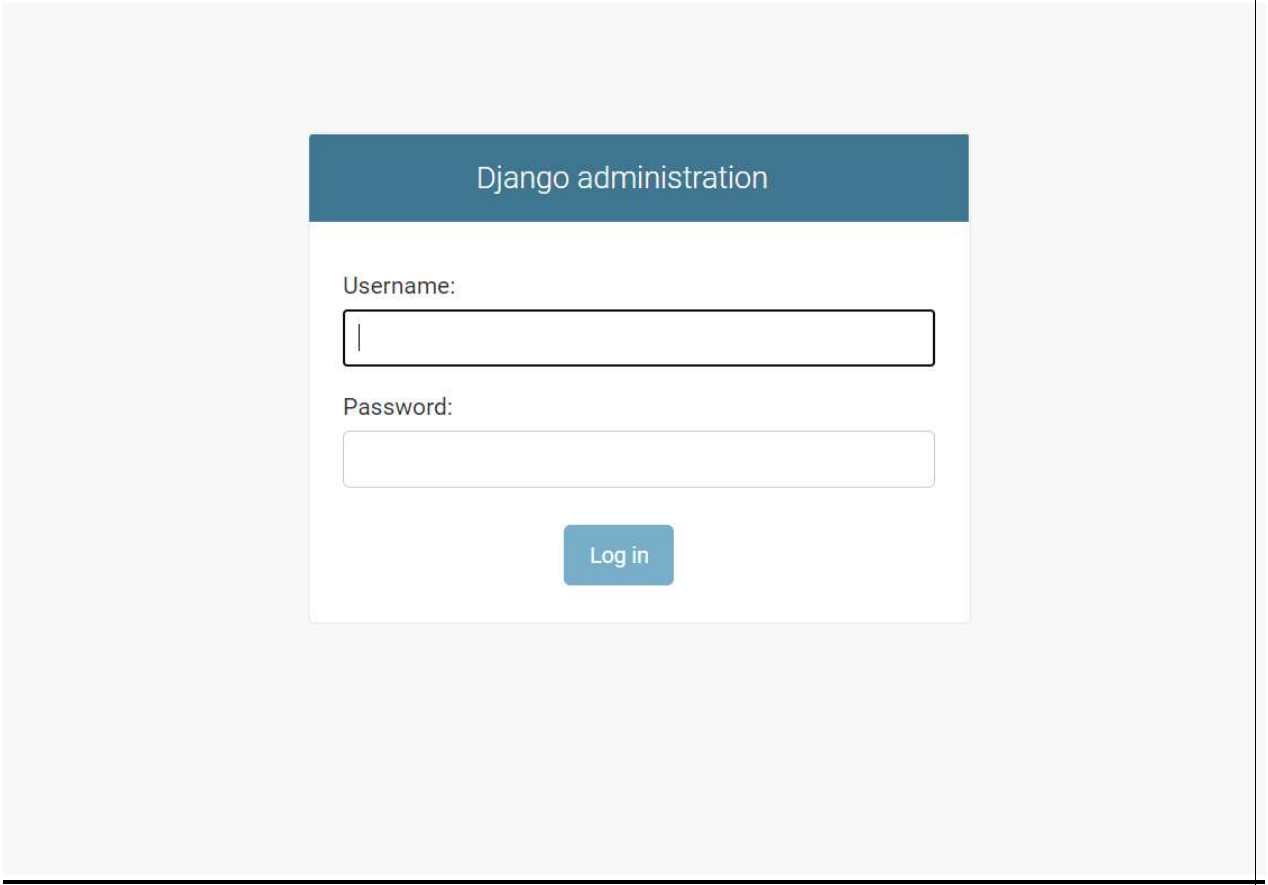


Fig: Sign in page.

In the models.py file above, we defined our Song model which represents a data table in our database to store our songs. The attributes of the class define the fields of the “Song” table in our database.

Code

```
from django.db import models

# Create your models here.

class Song(models.Model):
```

```
    title= models.TextField()
    artist= models.TextField()
    image= models.ImageField()
    audio_file = models.FileField(blank=True,null=True)
    #audio_link = models.CharField(max_length=200,blank=True,null=True)
    paginate_by = 2
    def __str__(self):
        return self.title
```

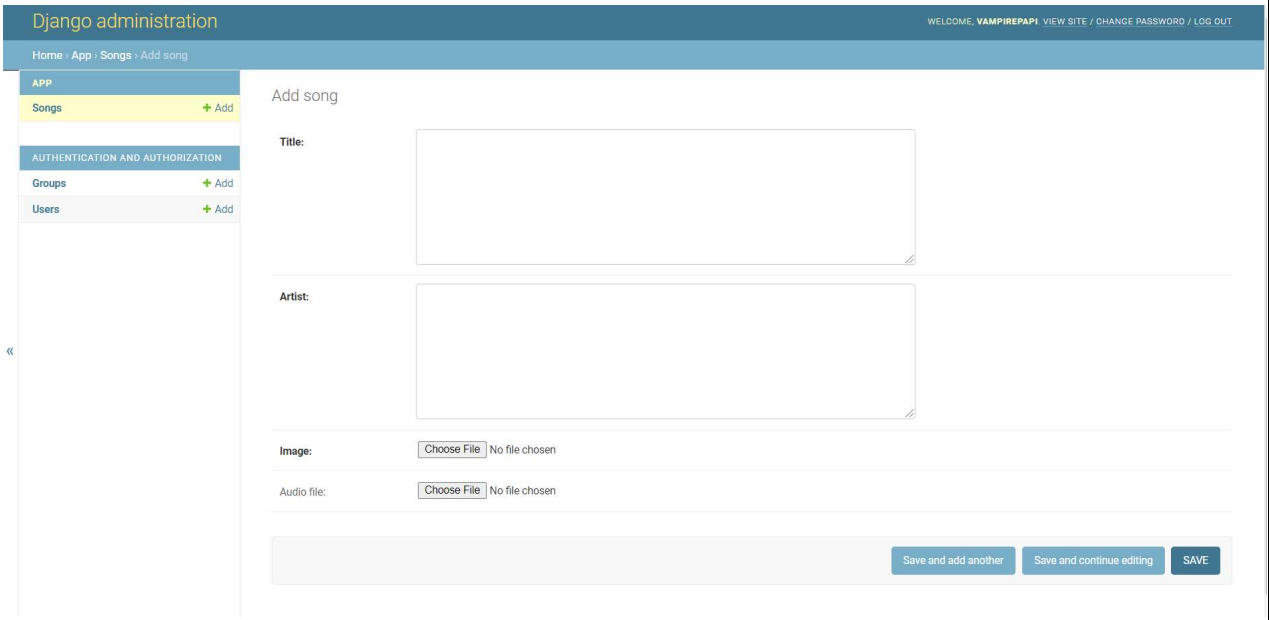


Fig: Adding our song through backend

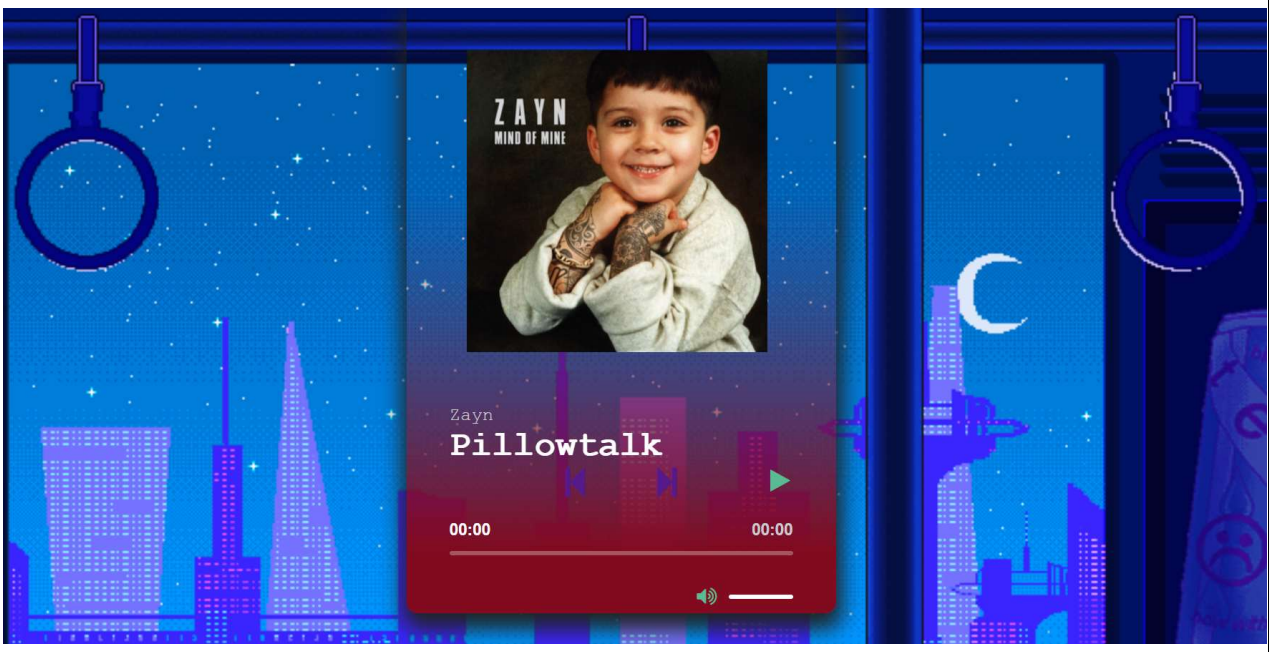


Fig: UI after adding a Song.

Working:-

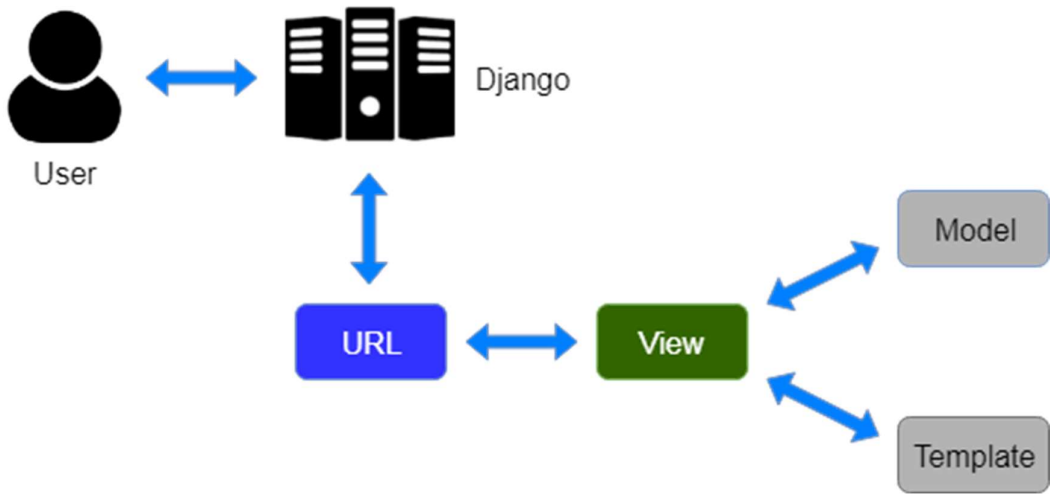
DJANGO MVT Architecture

MVT (Model View Template) is a software program layout sample that's a group of 3 elements: Model, View, and Template. The Model allows dealing with the database. It is a data access layer that handles the information in the database.

The Template is a presentation layer that handles all the User Interface parts. The View executes the logic and interact with the model to carry data and renders the template.

Although Django follows the MVC pattern, it still continues its conventions, so control is taken care of through the framework itself.

Here is an easy diagram that indicates the MVT structure in Django:



In the MVC architecture:-

- a user sends a request for a resource to Django, Django works as a controller and checks for the available resource in the URL.

- If a URL is mapped, a view is called that interact with the model and template, it renders a template.

- Django responds to the user and sends a template as a response.

Coding:-

URls.py(Backend)

Code

```
from django.conf import settings
from django.conf.urls.static import static
from django.urls import path, include
from . import views

app_name = "App"

urlpatterns = [
    path("", views.index, name="index"),
]
```

Views.py(Backend)
Code

```
# Create your views here.
from django.shortcuts import render, redirect

# imported our models
from django.core.paginator import Paginator
from . models import Song

def index(request):
    paginator= Paginator(Song.objects.all(),1)
    page_number = request.GET.get('page')
    page_obj = paginator.get_page(page_number)
    context={"page_obj":page_obj}
    return render(request,"index.html",context)
```

To play the media file follow script.js is used:-
Code

```
var audio = {
    init: function() {
    var $that = this;
        $(function() {
            $that.components.media();
        });
    },
    components: {
        media: function(target) {
            var media = $('audio.fc-media', (target !== undefined) ? target : 'body');
            if (media.length) {
                media.mediaelementplayer({
                    audioHeight: 40,
                    features : ['playpause', 'current', 'duration', 'progress', 'volume',
'tracks', 'fullscreen'],
                    alwaysShowControls      : true,
                    timeAndDurationSeparator: '<span></span>',
                    iPadUseNativeControls: true,
                    iPhoneUseNativeControls: true,
```

```
                AndroidUseNativeControls: true
            });
        }
    },

  },
};

audio.init();
```

Frontend:-

Index.html

Code

```
{% load static %}

<!DOCTYPE html>
<html lang="en">
 <head>
  <meta charset="utf-8"/>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>
   zaynify-vampirepapi
  </title>
  <link href="https://netdna.bootstrapcdn.com/font-awesome/4.0.3/css/font-awesome.css" rel="stylesheet"/>
  <link rel="icon" type="image/x-icon" href="{% static 'images/apple-music.png' %}">
  <link href="https://cdnjs.cloudflare.com/ajax/libs/mediaelement/4.2.7/mediaelementplayer.min.css" rel="stylesheet"/>
  <link rel="stylesheet" type="text/css" href="{% static 'style.css' %}">
 </head>
 <body>
  <!-- partial:index.partial.html -->
  <html>
   <head>
    <meta charset="utf-8"/>
   </head>
   <body>
    <div class="contain">
     <div class="container">
```

```
      <div class="music-player">
       {% for item in page_obj %}
       <div class="cover">
        <span title="loving it!? follow me on github:-vampirepapi">
        <a href="https://github.com/vampirepapi" target="_blank" rel="noopener
noreferrer">
        <img alt="" src="{{item.image.url}}"/>
        </a>
        </span>
       </div>
       <div class="titre">
        <h3>
         {{item.artist}}
        </h3>
        <h1>
         {{item.title}}
        </h1>
       </div>
       <center><a href="{% if page_obj.has_previous %}?page={{
page_obj.previous_page_number }}{% endif %}"><i class="fa fa-step-backward
fa-2x"></i></a>       <a href="{% if page_obj.has_next
%}?page={{ page_obj.next_page_number }} {% endif %}"><i class="fa fa-step-
forward fa-2x"></i></a></center>
       <div class="lecteur">
        <audio class="fc-media" style="width: 100%;">
         <source src="{% if item.audio_file %}{{item.audio_file.url}} {% else %}
{{item.audio_link}} {% endif %}" type="audio/mp3"/>
        </audio>

       </div>
       {% endfor %}
      </div>
     </div>
    </div>
   </body>
  </html>
  <!-- partial -->
  <script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.2.1/jquery.min.js">
  </script>
  <script
src="https://cdnjs.cloudflare.com/ajax/libs/mediaelement/4.2.7/mediaelement-
and-player.min.js">
  </script>
  <script src="{% static 'script.js' %}"></script>
  </script>
 </body>
```

```
</html>
```

style.css

Code

```
*:focus{
    outline: none;
}
html {
    background: #f2f2f2;
}
body {
    margin: 0;
    font-family: "Courier New",sans-serif;
}
h1 {
    margin: 0;
    font-size: 33px;
    color: #fff;
    padding: 0 10%;
}
h3 {
    margin: 0;
    font-size: 17px;
    font-weight: 500;
    color: #ccc;
    padding: 0 10%;
}
.container {
    display: block;
    width: 100%;
    height: 620px;
    margin: auto;
    overflow: hidden;
    background-image:
url("https://64.media.tumblr.com/b881f0e9323a401b3c0fadf89486258d/tumblr_ot
psp0uqBB1sqlugvo1_1280.gif");
    background-repeat: repeat;
    background-size: cover;
}
.music-player {
    display: block;
```

```
    position: relative;
    width: 400px;
    height: 570px;
    margin: auto;
    margin-top: 1%;
    border-radius: 0 0 10px 10px;
    background: transparent linear-gradient(to bottom,rgba(130,11,31,0)
30%,rgb(130,11,31) 90%) repeat scroll 0 0;
    box-shadow: 1px 10px 20px 5px #222;
}
.cover {
    float: left;
    width: 100%;
    height: 66%;
}
.cover img {
    display: block;
    position: absolute;
    top: 8%;
    left: 14%;
    width: 70%;
    margin: auto;
    text-align: center;
}
.titre {
    float: left;
    width: 100%;
}
.lecteur {
    width: 100%;
    display: block;
    height: auto;
    position: relative;
    float: left;
}
.mejs__button>button:focus {
    outline: 0px dotted #999;
}
.mejs__container {
    position: relative;
    background-color: transparent;
    min-width: auto !important;
}
.mejs__controls {
    padding: 0 10%;
    background: transparent !important;
```

```
    display: block;
    position: relative;
}
.mejs__controls div {
    display: block;
    float: left;
    position: relative;
}
.mejs__controls .mejs__playpause-button {
    position: absolute !important;
    right: 8%;
    bottom: 95%;
    width: 40px;
}
.mejs__controls .mejs__playpause-button button {
    display: block;
    width: 40px;
    height: 40px;
    padding: 0;
    border: 0;
    font-family: FontAwesome;
    font-size: 23px;
    color: #5bbb95;
    background: transparent;
    padding: 0;
    margin: 0;
}
.mejs__controls .mejs__play button:before{
    content:"\f04b";
}
.mejs__controls .mejs__pause button:before{
    content:"\f04c";
}
.mejs__controls .mejs__volume-button button {
    display: block;
    width: 40px;
    height: 40px;
    padding: 0;
    border: 0;
    font-family: FontAwesome;
    font-size: 20px;
    color: #5bbb95;
    background: transparent;
    margin: 0;
    padding: 0;
}
```

```
.mejs__controls .mejs__mute button:before {
    content: "\f028";
}
.mejs__controls .mejs__unmute button:before {
    content: "\f026";
}
.mejs__controls .mejs__time {
    width: 100%;
    margin-top: 7%;
    margin-bottom: 3%;
    color: #fff;
    height: auto;
    padding: 0;
    overflow: visible;
    min-width: 100%;
}
.mejs__controls .mejs__time span {
    font-size: 15px;
}
.mejs__controls span.mejs__duration {
    float: right;
    text-align: right;
    color: #ccc;
}
.mejs__controls span.mejs__currenttime {
    font-weight: 700;
    float: left;
}
.mejs__controls .mejs__time-rail {
    width: 100%;
    margin: 0;
}
.mejs__controls .mejs__time-rail span {
    position: absolute;
    top: 0;
    width: 100%;
    height: 4px;
    border-radius: 50px;
    cursor: pointer;
}
.mejs__controls .mejs__time-rail .mejs__time-loaded {
    background: rgba(255,255,255,0.2);
}
.mejs__controls .mejs__time-rail .mejs__time-float {
    display: none;
    top: -40px;
```

```
    width: 40px;
    height: 25px;
    margin-left: 0px;
    text-align: center;
    font-size: 10px;
    background: #fff;
    border: 0;
}
.mejs__controls .mejs__time-rail .mejs__time-float-current {
    display: block;
    position: relative;
    top: 0;
    margin: 0;
    line-height: 26px;
    color: #100d28;
}
.mejs__controls .mejs__time-rail .mejs__time-float-corner {
    top: auto;
    bottom: -9px;
    left: 50%;
    width: 0;
    height: 0;
    border-top: 6px solid #fff;
    border-right: 6px solid transparent;
    border-left: 6px solid transparent;
}
.mejs__controls .mejs__time-rail .mejs__time-current {
    background: #5BBB95 none repeat scroll 0 0;
}
.mejs__controls .mejs__time-handle {
    display: none;
}
.mejs__controls .mejs__volume-button {
    position: relative;
    position: absolute !important;
    top: 70px;
    right: 25%;
    width: 40px;
    height: 40px;
}
.mejs__controls .mejs__horizontal-volume-slider {
    display: block;
    position: absolute !important;
    position: relative;
    top: 70px;
    right: 10%;
```

```
    width: 60px;
    height: 4px;
    margin-top: 18px;
    border-radius: 50px;
    line-height: 11px;
}
.mejs__controls .mejs__horizontal-volume-slider .mejs__horizontal-volume-total,
.mejs__controls .mejs__horizontal-volume-slider .mejs__horizontal-volume-
current {
    position: absolute;
    top: 0;
    left: 0;
    width: 100%;
    height: 100%;
    background: rgba(255,255,255,0.1);
}
.mejs__controls .mejs__horizontal-volume-slider .mejs__horizontal-volume-
current {
    background: #fff;
}
```

Installation:-

1.) Download this project Zaynify

2.) Create a Virtual Environment if you are using PyCharm no need to create.

3.) Install Dependencies

```
pip install -r requirements.txt
```

4.) Move to project `cd Zaynify/`

5.) Execute `python manage.py runserver`

Deployment:-

As this project is made using Django it can be deployed on various platform eg. To know more about visit Deploying Python and Django Apps on Heroku

Conclusion:-

- Upon the completion of this project , i got the thorough knowlegde of Python, Frameworks, Django, Bootstrap etc.
- There were certain difficulties while intergrating backend to frontend,scalability issues, and many small issues as well but after the completion of the application, i not only got to know how to come over those problems but also how to never give up.
- I too learned how to use CLi's like CMD,Powershell.
- Creating Virtualenv and activating them and installing packages into Enviroment etc

References:-

For Python :- https://www.python.org/

For Django:- https://www.djangoproject.com/

ForBootstrap:-https://getbootstrap.com/docs/3.4/getting-started/