MEDIA STREAMING WITH IBM CLOUD STREAMING

PHASE – 5

PROJECT DOCUMENTATION AND SUBMISSION

Date	1 November 2023
Team ID	Proj_227254_Team_2
Project Name	Media Streaming With IBM Cloud Streaming

MEDIA STREAMING:

Streaming refers to any media content – live or recorded – delivered to computers and mobile devices via the internet and played back in real time.

- Music: Pandora, Spotify, Soundcloud.
- Video: YouTube, Vimeo, Vine.
- TV and Movies: Hulu, Netflix, Amazon Instant Video.
- Live video: Periscope, Meekat, and YouNow.
- Podcasts: iTunes, various smartphone applications.

Design Thinking:

1. USER RESEARCH:

Begin by understanding the needs and preferences of your target audience. are you targeting movie enthusiasts, independent filmmakers, or a different demographic? conduct surveys, interviews, or user studies to gather insights into what features and functionalities they expect from a virtual cinema platform.

2. ACCESSIBILITY:

Ensure that your platform is accessible to users with disabilities. this includes features like closed captioning, screen reader compatibility, and easy navigation for all users.

3. LEGAL AND LICENSING:

Be aware of legal considerations regarding content licensing, copyright, and intellectual property rights. consult with legal experts if necessary.

4. TESTING:

Rigorously test your platform to identify and fix any bugs or issues. conduct usability testing with real users to gather feedback on the user interface and overall experience.

5. DOCUMENTATION AND SUPPORT:

Create user guides and provide customer support channels to assist users with any questions or problems they may encounter. By empathizing with your target users and thoroughly understanding their needs, you can design and develop a virtual cinema platform that offers a compelling and enjoyable experience for both movie enthusiasts and content creators.

INNOVATION :

In Innovation phase we focus on implementing some of the main frame work and also some of the advanced frame work in media streaming.

- **Use Content Delivery Network (CDN) Integration**:
  A content delivery network (CDN) is a group of geographically distributed services that speed up the delivery of web content by bringing it closer to where users are.

- **Multi-Platform Streaming:**
  Enable simultaneous streaming to multiple social media platforms and streaming services.

- **Analytics and Reporting:**
  Implement analytics tools to gather insights into viewer behavior, helping you make informed decisions and improve content delivery.

- **Content Management:**
  Utilize IBM Cloud's storage and content management solutions to efficiently organize and manage your media assets.

- **Multi-Device Support**:
  Ensure compatibility across various devices, including smartphones, tablets, smart TVs, and desktops

STEP BY STEP INSTRUCTIONS TO IMPLEMENT MEDIA STREAMING BY USING IBM CLOUD:

STEP 1: Content Preparation:

Content preparation is an integral part of the usability equation: it answers the question of what information is needed for effective decision making. Once content preparation has been established, the question "how to present what" can be answered.

Content storage management:

Interfaces to proprietary media creation and consumption devices regardless of interface and storage topology.

Direct integration to any type and format of storage device typically categorized as IT-centric storage devices allowing limitless or near limitless storage expansion

STEP 3:

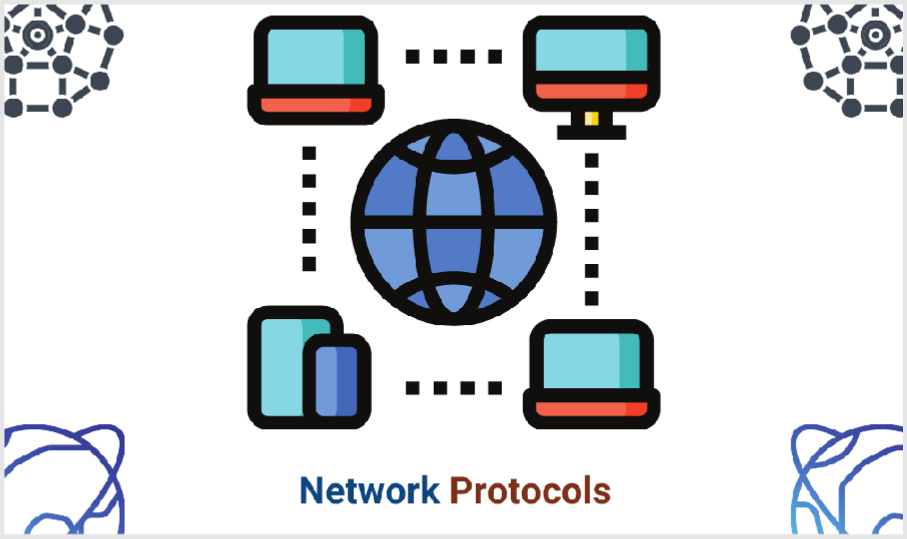
Streaming Protocol Selection:

Choose a streaming protocol that suits your needs. Common protocols include:

HTTP Live Streaming (HLS): Used by Apple devices.

Dynamic Adaptive Streaming over HTTP (DASH): Supports multiple formats and is more platform-agnostic.

Real-Time Messaging Protocol (RTMP): Suitable for live streaming.



STEP 4:

Server Setup:

- Set up a media server or streaming server software that supports your chosen streaming protocol.
  - **NGINX with RTMP module** for RTMP streaming.

STEP 5:

Segment Delivery:

Set up a system to deliver these segments over HTTP(S). This involves creating a directory structure for the segments and updating the playlist files accordingly.

STEP:6

Playback Control:

- Implement playback control features like play, pause, seek, and volume control.

STEP 7:

Testing and Optimization:

Test your streaming setup under various conditions (e.g., different devices, network speeds) to ensure a smooth user experience.
Optimize for performance and scalability, considering CDN integration, load balancing, and content caching

STEP 8:

Monitoring and Analytics:

Implement monitoring and analytics tools to track viewer metrics, diagnose issues, and make data-driven improvements.

STEP 9:

Moderation and Reporting Moderation and Reporting:

Implement moderation features to handle inappropriate content and user reports. Allow users to report abusive or spammy messages.

STEP 10:

Scalability and Compatibility Scalability and Compatibility:

Ensure that your chat system can handle a large number of concurrent users. Optimize the chat interface for both mobile and desktop users, providing a consistent user experience.

STEP 11:

Documentation and User Support:

- Create documentation for users and provide support channels for any issues they encounter.

STREAMING REQUIREMENTS:

Streaming usually requires a reliable, high-speed internet connection because the media files must be retrieved from a remote location and then delivered to a user's local system with minimal lag or latency (delay). A slow connection decreases the speed at which the content is delivered, affecting the user's streaming experience.

INSTALL PYTHON IN OS:

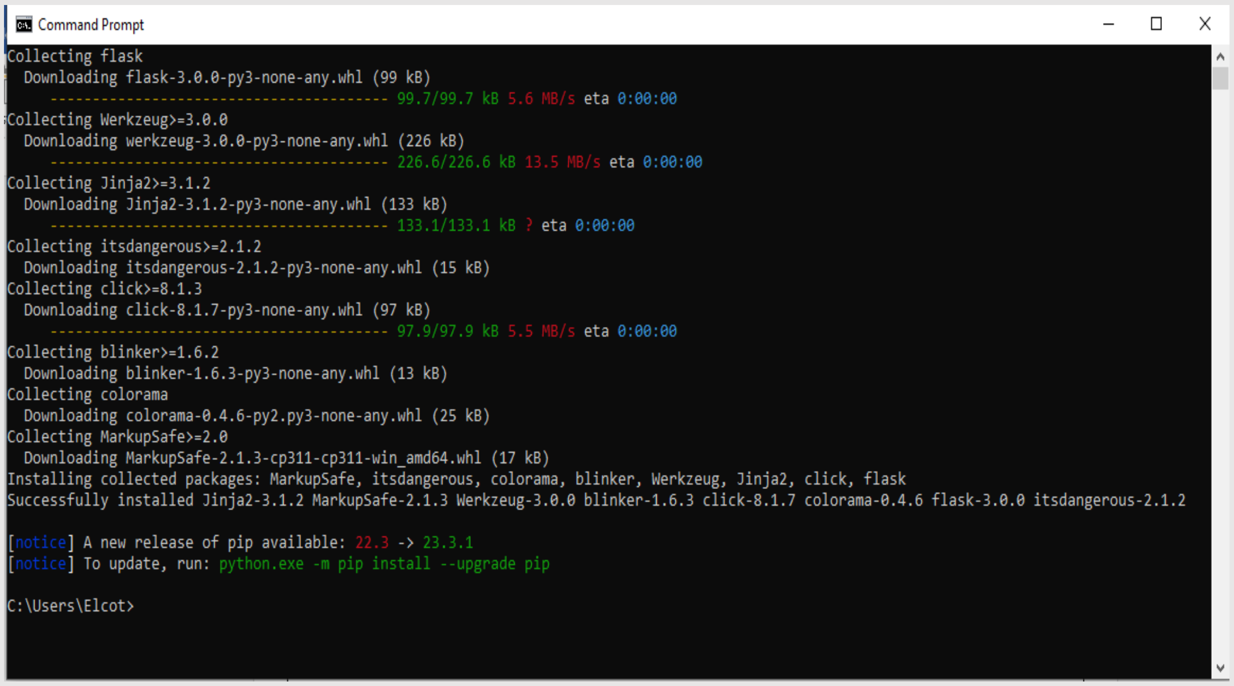
1. **Download PyCharm**:
   Visit the JetBrains website (https://www.jetbrains.com/pycharm/download/) and download the community (free) or professional version of PyCharm, depending on your needs. Make sure to download the Windows version.
2. **Run the Installer**:
   Locate the downloaded installer (usually an .exe file) and double-click it to run the installation.
3. **Choose Installation Type**:
   During the installation, you'll be prompted to select the installation type. You can choose the default settings or customize them according to your preferences. You can also select the option to create associations for .py files, which allows you to open Python scripts with PyCharm by default.
4. **Select the Destination Folder**:
   Choose the folder where you want to install PyCharm, or leave it at the default location.
5. **Create Shortcuts**:
   You can choose whether you want to create desktop shortcuts or add PyCharm to the Start menu.
6. **Complete the Installation**:
   Click the "Install" button to start the installation process. PyCharm will be installed on your system.
7. **Launch PyCharm**:
   Once the installation is complete, you can launch PyCharm by checking the "Run PyCharm" option in the installer or by finding it in your Start menu or desktop shortcuts.

INSTALLING PACKAGES :

1.PACKAGE NAME: Flask

USE : to use flask framework in python

COMMAND TO INSTALL: pip install flask



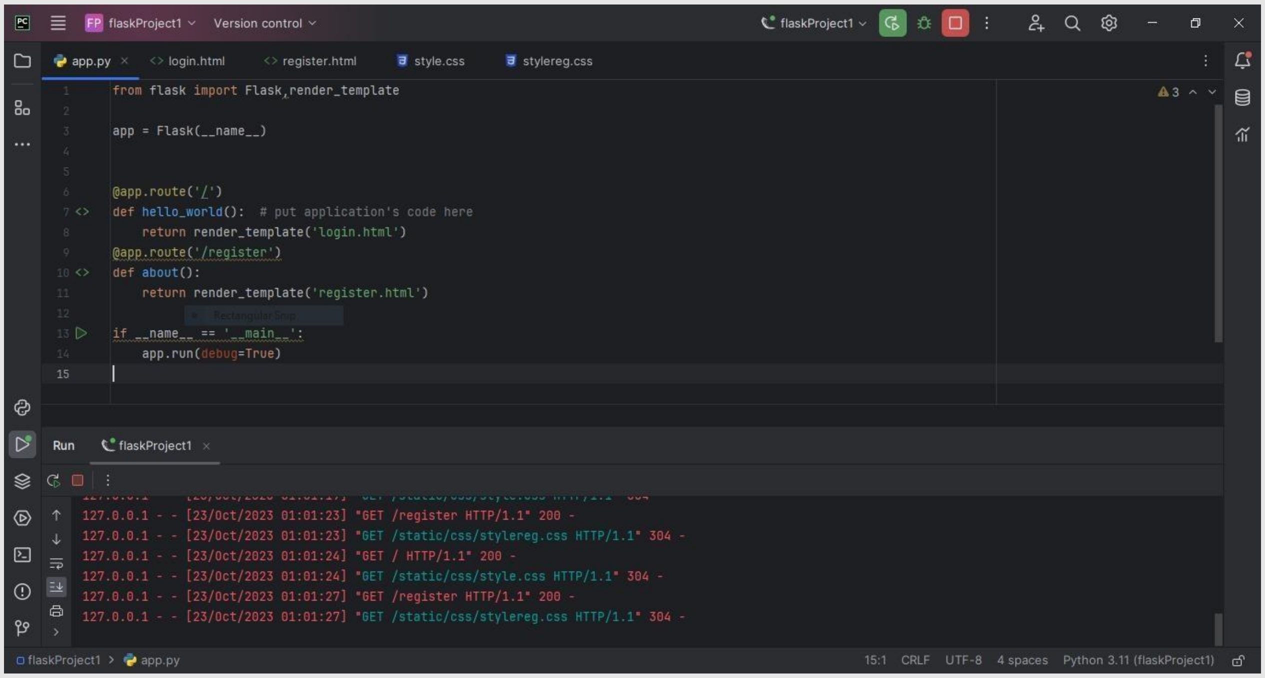
```
Collecting flask
  Downloading flask-3.0.0-py3-none-any.whl (99 kB)
     ---------------------------------------- 99.7/99.7 kB 5.6 MB/s eta 0:00:00
Collecting Werkzeug>=3.0.0
  Downloading werkzeug-3.0.0-py3-none-any.whl (226 kB)
     ---------------------------------------- 226.6/226.6 kB 13.5 MB/s eta 0:00:00
Collecting Jinja2>=3.1.2
  Downloading Jinja2-3.1.2-py3-none-any.whl (133 kB)
     ---------------------------------------- 133.1/133.1 kB ? eta 0:00:00
Collecting itsdangerous>=2.1.2
  Downloading itsdangerous-2.1.2-py3-none-any.whl (15 kB)
Collecting click>=8.1.3
  Downloading click-8.1.7-py3-none-any.whl (97 kB)
     ---------------------------------------- 97.9/97.9 kB 5.5 MB/s eta 0:00:00
Collecting blinker>=1.6.2
  Downloading blinker-1.6.3-py3-none-any.whl (13 kB)
Collecting colorama
  Downloading colorama-0.4.6-py2.py3-none-any.whl (25 kB)
Collecting MarkupSafe>=2.0
  Downloading MarkupSafe-2.1.3-cp311-cp311-win_amd64.whl (17 kB)
Installing collected packages: MarkupSafe, itsdangerous, colorama, blinker, Werkzeug, Jinja2, click, flask
Successfully installed Jinja2-3.1.2 MarkupSafe-2.1.3 Werkzeug-3.0.0 blinker-1.6.3 click-8.1.7 colorama-0.4.6 flask-3.0.0 itsdangerous-2.1.2

[notice] A new release of pip available: 22.3 -> 23.3.1
[notice] To update, run: python.exe -m pip install --upgrade pip

C:\Users\Elcot>
```

CODES WERE USED IN THIS PROJECT:

1. Code this to render a html code in flask

```
from flask import Flask,render_template, request, redirect, url_for, send_from_directory
import os
app = Flask(_name_)
app.config['UPLOAD_FOLDER'] = 'upload'
app.static_folder = 'static'
@app.route('/')
def hello_world():  # put application's code here
    return render_template('login.html')
@app.route('/register')
def about():
    return render_template('register.html')
@app.route('/video')
```

```
def upload_form():
    return  render_template('upload.html')
@app.route('/upload', methods=['POST'])
def upload_video():
    if 'video' not in request.files:
        return redirect(request.url)
    video = request.files['video']
    if video.filename == '':
        return redirect(request.url)
    if video:
        video.save(os.path.join(app.config['UPLOAD_FOLDER'], video.filename))
        return redirect(url_for('view_video', filename=video.filename))
@app.route('/upload/<filename>')
def view_video(filename):
    return render_template('view.html', filename=filename)
@app.route('/upload/<filename>')
def send video(filename):
    return send_from_directory(app.config['UPLOAD_FOLDER'], filename,
mimetype='video/mp4')
@app.route('/home')
def home():
    return render_template('home.html')
@app.route('/feedback')
def feedback():
    return render_template('feedback.html')
@app.route('/profile')
def profile():
    return render_template('profile.html')
@app.route('/subscription')
def subscription():
    return render_template('subscription.html')
if _name_ == '_main_':
    app.run(debug=True)
```



```
from flask import Flask,render_template

app = Flask(__name__)


@app.route('/')
def hello_world():  # put application's code here
    return render_template('login.html')
@app.route('/register')
def about():
    return render_template('register.html')

if __name__ == '__main__':
    app.run(debug=True)
```

```
127.0.0.1 - - [23/Oct/2023 01:01:23] "GET /register HTTP/1.1" 200 -
127.0.0.1 - - [23/Oct/2023 01:01:23] "GET /static/css/stylereg.css HTTP/1.1" 304 -
127.0.0.1 - - [23/Oct/2023 01:01:24] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [23/Oct/2023 01:01:24] "GET /static/css/style.css HTTP/1.1" 304 -
127.0.0.1 - - [23/Oct/2023 01:01:27] "GET /register HTTP/1.1" 200 -
127.0.0.1 - - [23/Oct/2023 01:01:27] "GET /static/css/stylereg.css HTTP/1.1" 304 -
```

2. Code for login page html:

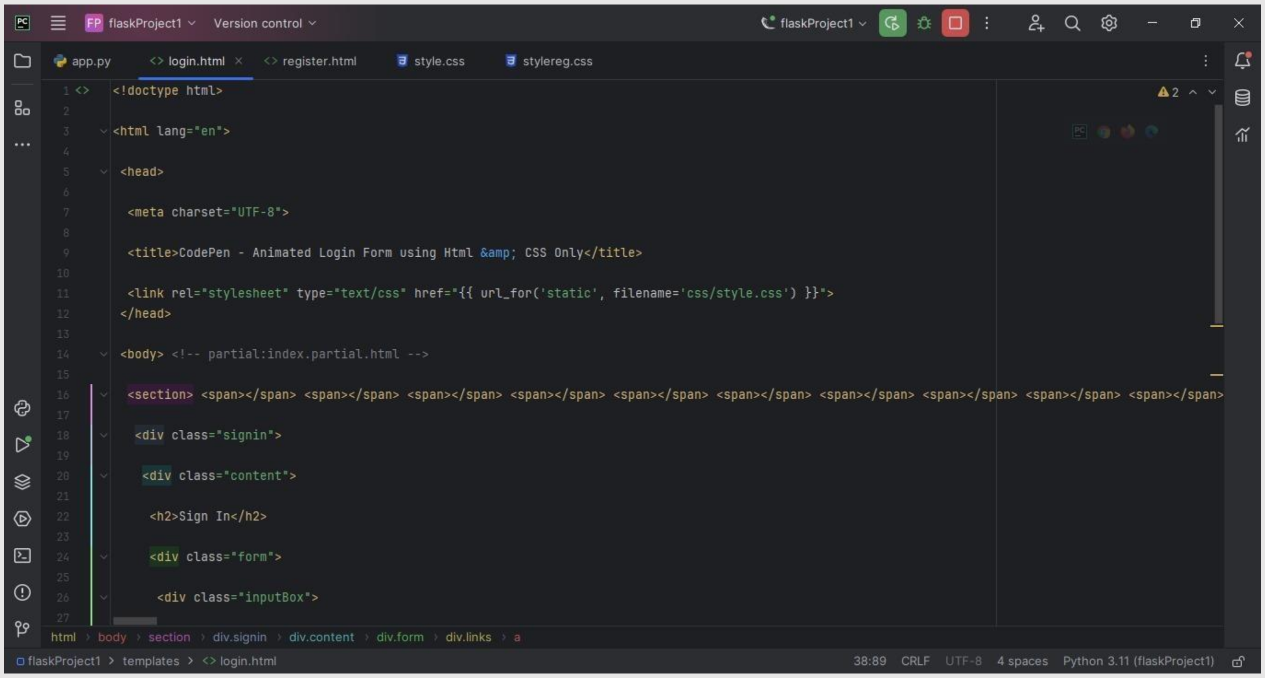
```
<!doctype html>
<html lang="en">
 <head>
  <meta charset="UTF-8">
  <title>CodePen - Animated Login Form using Html &amp; CSS Only</title>
  <link rel="stylesheet" type="text/css" href="{{ url_for('static', filename='css/style.css') }}">
 </head>
 <body> <!-- partial:index.partial.html -->
```

<section> <span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>

```
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
   <div class="signin">
    <div class="content">
     <h2>Sign In</h2>
     <div class="form">
      <div class="inputBox">
       <input type="text" required> <i>Email</i>
      </div>
      <div class="inputBox">
       <input type="password" required> <i>Password</i>
      </div>
      <div class="links"> <a href="#">Forgot Password</a> <a href="/register">Signup</a>
      </div>
      <div class="inputBox">
       <input type="submit" value="Login">
      </div>
     </div>
    </div>
   </div>
  </section> <!-- partial -->
```

</body>

</html>


flaskProject1
Version control
app.py
login.html
register.html
style.css
stylereg.css
html › body › section › div.signin › div.content › div.form › div.links › a
flaskProject1 > templates > login.html
38:89 CRLF UTF-8 4 spaces Python 3.11 (flaskProject1)

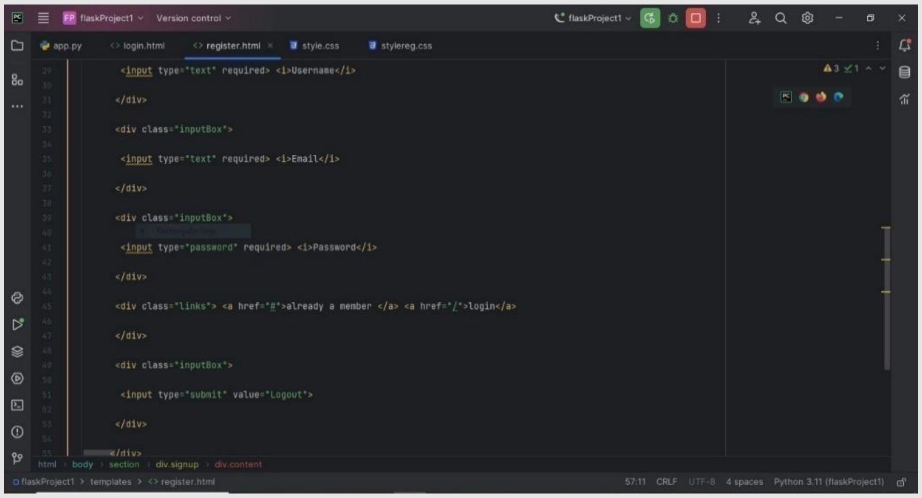
```
<!doctype html>

<html lang="en">

 <head>

  <meta charset="UTF-8">

  <title>CodePen - Animated Login Form using Html &amp; CSS Only</title>

  <link rel="stylesheet" type="text/css" href="{{ url_for('static', filename='css/style.css') }}">
 </head>

 <body> <!-- partial:index.partial.html -->

  <section> <span></span> <span></span> <span></span> <span></span> <span></span> <span></span> <span></span> <span></span> <span></span> <span></span>

   <div class="signin">

    <div class="content">

     <h2>Sign In</h2>

     <div class="form">

      <div class="inputBox">
```

3. Code for registration page html

```
<!doctype html>
<html lang="en">
 <head>
  <meta charset="UTF-8">
  <title>CodePen - Animated Login Form using Html &amp; CSS Only</title>
  <link rel="stylesheet" type="text/css" href="{{ url_for('static',
filename='css/stylereg.css') }}">
 </head>
 <body> <!-- partial:index.partial.html -->
  <section> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
```

<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span> <span></span> <span></span> <span></span> <span></span>

```
<span></span> <span></span> <span></span> <span></span> <span></span>
<span></span>
   <div class="signup">
    <div class="content">
     <h2>Sign Up</h2>
     <div class="form">
      <div class="inputBox">
       <input type="text" required> <i>Username</i>
      </div>
      <div class="inputBox">
       <input type="number" required> <i>Mobile No</i>
      </div>
      <div class="inputBox">
       <input type="text" required> <i>Email</i>
      </div>
      <div class="inputBox">
       <input type="password" required> <i>Password</i>
      </div>
      <div class="links"> <a href="#">already a member </a> <a href="/">login</a>
      </div>
      <div class="inputBox">
       <input type="submit" value="Create">
      </div>
     </div>
    </div>
   </div>
  </section> <!-- partial -->
 </body>
</html>
```



```
 <input type="text" required> <i>Username</i>

</div>

<div class="inputBox">

 <input type="text" required> <i>Email</i>

</div>

<div class="inputBox">

 <input type="password" required> <i>Password</i>

</div>

<div class="links"> <a href="#">already a member </a> <a href="/">login</a>

</div>

<div class="inputBox">

 <input type="submit" value="Logout">

</div>

</div>
```

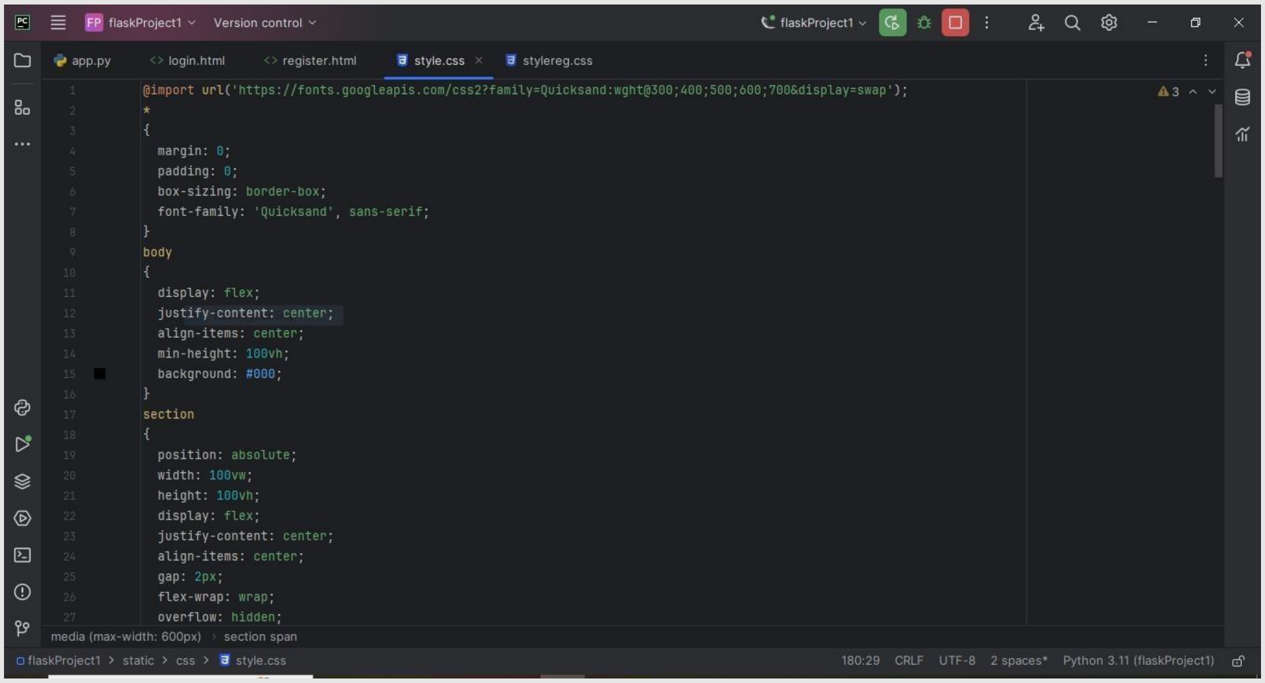
html › body › section › div.signup › div.content

flaskProject1 > templates > register.html   57:11 CRLF UTF-8 4 spaces Python 3.11 (flaskProject1)

4. SYNCHRONIZING A CSS FILE WITH LOGIN HTML FOR DESIGNING PURPOSE:

```
@import
url('https://fonts.googleapis.com/css2?family=Quicksand:wght@300;400;500;600;700&disp
lay=swap');

*

{

  margin: 0;

  padding: 0;

  box-sizing: border-box;

  font-family: 'Quicksand', sans-serif;

}

body

{

  display: flex;

  justify-content: center;

  align-items: center;

  min-height: 100vh;

  background: #000;

}

section

{

  position: absolute;

  width: 100vw;

  height: 100vh;

  display: flex;

  justify-content: center;

  align-items: center;
  gap: 2px;
  flex-wrap: wrap;
  overflow: hidden;
}
section::before
{
  content: '';
  position: absolute;
  width: 100%;
  height: 100%;
  background: linear-gradient(#000,#0f0,#000);
  animation: animate 5s linear infinite;
}
@keyframes animate
{
  0%
  {
    transform: translateY(-100%);
  }
  100%
  {
    transform: translateY(100%);
  }
}
section span
{
  position: relative;
  display: block;
  width: calc(6.25vw - 2px);
  height: calc(6.25vw - 2px);
  background: #181818;
  z-index: 2;
  transition: 1.5s;
}
section span:hover
{
  background: #0f0;
  transition: 0s;
}
section .signin
{
  position: absolute;
  width: 400px;
  background: #222;
  z-index: 1000;
  display: flex;
  justify-content: center;
  align-items: center;
  padding: 40px;
  border-radius: 4px;
  box-shadow: 0 15px 35px rgba(0,0,0,9);
}
section .signin .content
{
  position: relative;
  width: 100%;
  display: flex;
  justify-content: center;
  align-items: center;
  flex-direction: column;
  gap: 40px;
}
section .signin .content h2
{
  font-size: 2em;
  color: #0f0;
  text-transform: uppercase;
}
section .signin .content .form
{
  width: 100%;
  display: flex;
  flex-direction: column;
  gap: 25px;
}
section .signin .content .form .inputBox
{
  position: relative;
  width: 100%;
}
section .signin .content .form .inputBox input
```

```
{
  position: relative;
  width: 100%;
  background: #333;
  border: none;
  outline: none;
  padding: 25px 10px 7.5px;
  border-radius: 4px;
  color: #fff;
  font-weight: 500;
  font-size: 1em;
}
section .signin .content .form .inputBox i
{
  position: absolute;
  left: 0;
  padding: 15px 10px;
  font-style: normal;
  color: #aaa;
  transition: 0.5s;
  pointer-events: none;
}
.signin .content .form .inputBox input:focus ~ i,
.signin .content .form .inputBox input:valid ~ i
{
  transform: translateY(-7.5px);
  font-size: 0.8em;
  color: #fff;
}
.signin .content .form .links
{
  position: relative;
  width: 100%;
  display: flex;
  justify-content: space-between;
}
.signin .content .form .links a
{
  color: #fff;
  text-decoration: none;
}
.signin .content .form .links a:nth-child(2)
{
  color: #0f0;
  font-weight: 600;
}
.signin .content .form .inputBox input[type="submit"]
{
  padding: 10px;
  background: #0f0;
  color: #000;
  font-weight: 600;
  font-size: 1.35em;
  letter-spacing: 0.05em;
  cursor: pointer;
}
input[type="submit"]:active
{
  opacity: 0.6;
}
@media (max-width: 900px)
{
  section span
  {
    width: calc(10vw - 2px);
    height: calc(10vw - 2px);
  }
}
@media (max-width: 600px)
{
  section span
  {
    width: calc(20vw - 2px);
    height: calc(20vw - 2px);
  }
}
```


flaskProject1
Version control
flaskProject1
app.py
login.html
register.html
style.css
stylereg.css
media (max-width: 600px) › section span
flaskProject1 › static › css › style.css
180:29 CRLF UTF-8 2 spaces* Python 3.11 (flaskProject1)

```
@import url('https://fonts.googleapis.com/css2?family=Quicksand:wght@300;400;500;600;700&display=swap');
*
{
  margin: 0;
  padding: 0;
  box-sizing: border-box;
  font-family: 'Quicksand', sans-serif;
}
body
{
  display: flex;
  justify-content: center;
  align-items: center;
  min-height: 100vh;
  background: #000;
}
section
{
  position: absolute;
  width: 100vw;
  height: 100vh;
  display: flex;
  justify-content: center;
  align-items: center;
  gap: 2px;
  flex-wrap: wrap;
  overflow: hidden;
```

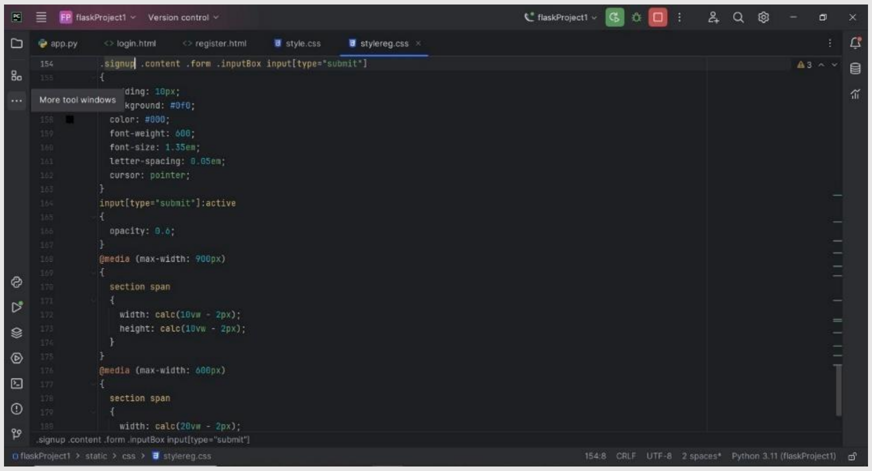
5. SYNCHRONIZING A CSS FILE WITH REGISTRATION HTML FOR DESIGNING PURPOSE:

```
@import
url('https://fonts.googleapis.com/css2?family=Quicksand:wght@300;400;500;600;700&display=swap');

*

{

  margin: 0;

  padding: 0;

  box-sizing: border-box;

  font-family: 'Quicksand', sans-serif;

}

body

{

  display: flex;

  justify-content: center;

  align-items: center;
  min-height: 100vh;
  background: #000;
}
section
{
  position: absolute;
  width: 100vw;
  height: 100vh;
  display: flex;
  justify-content: center;
  align-items: center;
  gap: 2px;
  flex-wrap: wrap;
  overflow: hidden;
}
section::before
{
  content: '';
  position: absolute;
  width: 100%;
  height: 100%;
  background: linear-gradient(#000,#0f0,#000);
  animation: animate 5s linear infinite;
}
@keyframes animate
{
  0%
  {
    transform: translateY(-100%);
  }
  100%
  {
    transform: translateY(100%);
  }
}
section span
{
  position: relative;
  display: block;
  width: calc(6.25vw - 2px);
  height: calc(6.25vw - 2px);
  background: #181818;
  z-index: 2;
  transition: 1.5s;
}
section span:hover
{
  background: #0f0;
  transition: 0s;
}

section .signup
{
  position: absolute;
  width: 400px;
  background: #222;
  z-index: 1000;
  display: flex;
  justify-content: center;
  align-items: center;
  padding: 40px;
  border-radius: 4px;
  box-shadow: 0 15px 35px rgba(0,0,0,9);
}

section .signup .content
{
  position: relative;
  width: 100%;
  display: flex;
  justify-content: center;
  align-items: center;
  flex-direction: column;
  gap: 40px;
}
section .signup .content h2
{
  font-size: 2em;
  color: #0f0;
  text-transform: uppercase;
}
```

```
section .signup .content .form
{
  width: 100%;
  display: flex;
  flex-direction: column;
  gap: 25px;
}
section .signup .content .form .inputBox
{
  position: relative;
  width: 100%;
}
section .signup .content .form .inputBox input
{
  position: relative;
  width: 100%;
  background: #333;
  border: none;
  outline: none;
  padding: 25px 10px 7.5px;
  border-radius: 4px;
  color: #fff;
  font-weight: 500;
  font-size: 1em;
}
section .signup .content .form .inputBox i
{
  position: absolute;
  left: 0;
  padding: 15px 10px;
  font-style: normal;
  color: #aaa;
  transition: 0.5s;
  pointer-events: none;
}
.signup .content .form .inputBox input:focus ~ i,
.signup .content .form .inputBox input:valid ~ i
{
  transform: translateY(-7.5px);
  font-size: 0.8em;
  color: #fff;
}
.signup .content .form .links
{
  position: relative;
  width: 100%;
  display: flex;
  justify-content: space-between;
}
.signup .content .form .links a
{
  color: #fff;
  text-decoration: none;
}
```

```
.signup .content .form .links a:nth-child(2)
{
  color: #0f0;
  font-weight: 600;
}
.signup .content .form .inputBox input[type="submit"]
{
  padding: 10px;
  background: #0f0;
  color: #000;
  font-weight: 600;
  font-size: 1.35em;
  letter-spacing: 0.05em;
  cursor: pointer;
}
input[type="submit"]:active
{
  opacity: 0.6;
}
@media (max-width: 900px)
{
  section span
  {
    width: calc(10vw - 2px);
    height: calc(10vw - 2px);
  }
}
```

```
@media (max-width: 600px)

{

  section span

  {

    width: calc(20vw - 2px);

    height: calc(20vw - 2px);

  }

}
```



flaskProject1
Version control
app.py
login.html
register.html
style.css
stylereg.css
More tool windows

```
.signup .content .form .inputBox input[type="submit"]
{
  ding: 10px;
  kground: #0f0;
  color: #000;
  font-weight: 600;
  font-size: 1.35em;
  letter-spacing: 0.05em;
  cursor: pointer;
}
input[type="submit"]:active
{
  opacity: 0.6;
}
@media (max-width: 900px)
{
  section span
  {
    width: calc(10vw - 2px);
    height: calc(10vw - 2px);
  }
}
@media (max-width: 600px)
{
  section span
  {
    width: calc(20vw - 2px);
```

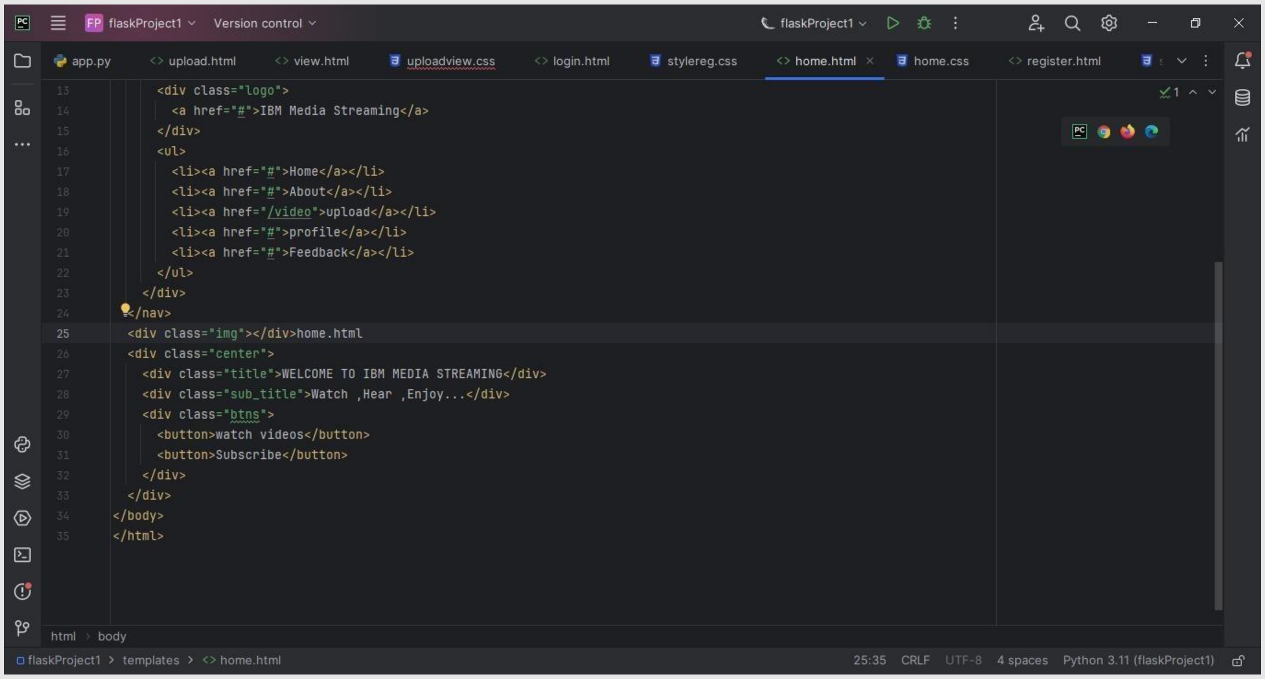
.signup .content .form .inputBox input[type="submit"]
flaskProject1 > static > css > stylereg.css
154:8 CRLF UTF-8 2 spaces* Python 3.11 (flaskProject1)

6. CREATING A SQL TO STORE THE DATA's IN THE DATABASE:

```
CREATE DATABASE IF NOT EXISTS UserDatabase;

USE UserDatabase;

CREATE TABLE IF NOT EXISTS Users (
    UserID INT AUTO_INCREMENT PRIMARY KEY,
    Username VARCHAR(50) NOT NULL,
    MobileNo VARCHAR(15) NOT NULL,
    EmailID VARCHAR(100) NOT NULL,
    Password VARCHAR(255) NOT NULL
);
```

7. HTML CODE TO CREATE A HOMEPAGE:

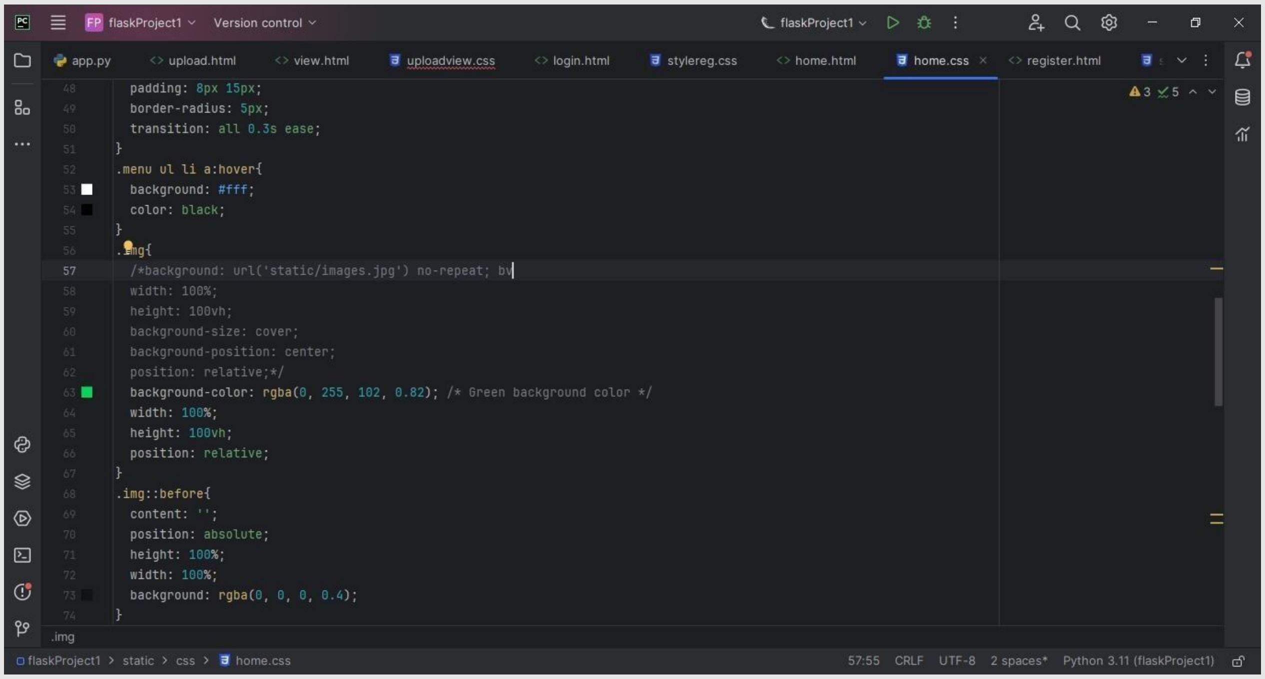
```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title> Website Layout | CodingLab</title>
    <!link rel="stylesheet" href="css/home.css">
    <link rel="stylesheet" type="text/css" href="{{ url_for('static',
filename='css/home.css') }}">
   </head>
<body>
  <nav>
    <div class="menu">
      <div class="logo">
        <a href="#">IBM Media Streaming</a>
      </div>
      <ul>
        <li><a href="#">Home</a></li>
        <li><a href="#">About</a></li>
        <li><a href="/video">upload</a></li>
        <li><a href="/profile">profile</a></li>
        <li><a href="/feedback">Feedback</a></li>
      </ul>
    </div>
  </nav>
  <div class="img"></div>
  <div class="center">
    <div class="title">WELCOME TO IBM MEDIA STREAMING</div>
    <div class="sub_title">Watch ,Hear ,Enjoy...</div>
    <div class="btns">
      <button>watch videos</button>
      <button>Subscribe</button>
    </div>
  </div>
</body>
</html>
```



```
        <div class="logo">
          <a href="#">IBM Media Streaming</a>
        </div>
        <ul>
          <li><a href="#">Home</a></li>
          <li><a href="#">About</a></li>
          <li><a href="/video">upload</a></li>
          <li><a href="#">profile</a></li>
          <li><a href="#">Feedback</a></li>
        </ul>
      </div>
    </nav>
    <div class="img"></div>home.html
    <div class="center">
      <div class="title">WELCOME TO IBM MEDIA STREAMING</div>
      <div class="sub_title">Watch ,Hear ,Enjoy...</div>
      <div class="btns">
        <button>watch videos</button>
        <button>Subscribe</button>
      </div>
    </div>
  </body>
  </html>
```

8. CSS CODE TO CREATE A HOMEPAGE:

```
@import
url('https://fonts.googleapis.com/css2?family=Poppins:wght@200;300;400;500;600;700
&display=swap');
*{
  margin: 0;
  padding: 0;
  box-sizing: border-box;
  font-family: 'Poppins',sans-serif;
}
::selection{
  color: #000;
  background: #fff;
}
nav{
  position: fixed;
  background: #1b1b1b;
  width: 100%;
  padding: 10px 0;
  z-index: 12;
}
nav .menu{
  max-width: 1250px;
  margin: auto;
  display: flex;
  align-items: center;
  justify-content: space-between;
  padding: 0 20px;
}
.menu .logo a{
  text-decoration: none;
  color: #fff;
  font-size: 35px;
  font-weight: 600;
}
.menu ul{
  display: inline-flex;
}
.menu ul li{
  list-style: none;
  margin-left: 7px;
}
.menu ul li:first-child{
  margin-left: 0px;
}
.menu ul li a{
  text-decoration: none;
  color: #fff;
  font-size: 18px;
  font-weight: 500;
  padding: 8px 15px;
  border-radius: 5px;
  transition: all 0.3s ease;
}
.menu ul li a:hover{
  background: #fff;
  color: black;
}
.img{
  /*background: url('static/images.jpg') no-repeat; bv
  width: 100%;
  height: 100vh;
  background-size: cover;
  background-position: center;
  position: relative;*/
  background-color: rgba(0, 255, 102, 0.82); /* Green background color */
  width: 100%;
  height: 100vh;
  position: relative;
}
.img::before{
  content: '';
  position: absolute;
  height: 100%;
  width: 100%;
  background: rgba(0, 0, 0, 0.4);
}
.center{
  position: absolute;
  top: 52%;
  left: 50%;
  transform: translate(-50%, -50%);
  width: 100%;
  padding: 0 20px;
  text-align: center;
}
.center .title{
  color: #fff;
  font-size: 55px;
  font-weight: 600;
}
.center .sub_title{
  color: #fff;
  font-size: 52px;
  font-weight: 600;
}
.center .btns{
  margin-top: 20px;
}
.center .btns button{
  height: 55px;
  width: 170px;
  border-radius: 5px;
  border: none;
  margin: 0 10px;
  border: 2px solid white;
  font-size: 20px;
  font-weight: 500;
  padding: 0 10px;
  cursor: pointer;
  outline: none;
  transition: all 0.3s ease;
}
.center .btns button:first-child{
  color: #fff;
  background: none;
}
.btns button:first-child:hover{
  background: white;
  color: black;
}
.center .btns button:last-child{
  background: white;
  color: black;
}
```

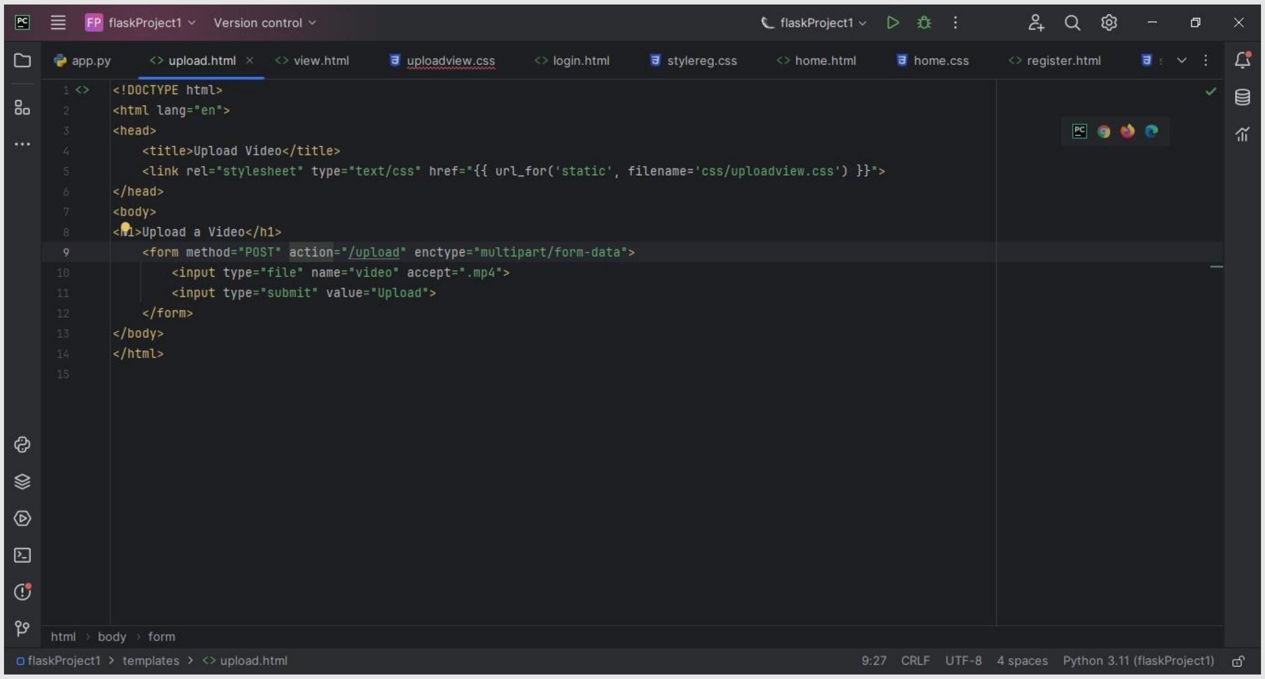


```
  padding: 8px 15px;
  border-radius: 5px;
  transition: all 0.3s ease;
}
.menu ul li a:hover{
  background: #fff;
  color: black;
}
.img{
  /*background: url('static/images.jpg') no-repeat; bv
  width: 100%;
  height: 100vh;
  background-size: cover;
  background-position: center;
  position: relative;*/
  background-color: rgba(0, 255, 102, 0.82); /* Green background color */
  width: 100%;
  height: 100vh;
  position: relative;
}
.img::before{
  content: '';
  position: absolute;
  height: 100%;
  width: 100%;
  background: rgba(0, 0, 0, 0.4);
}
```

.img
flaskProject1 > static > css > home.css
57:55 CRLF UTF-8 2 spaces* Python 3.11 (flaskProject1)

9. HTML CODE TO UPLOAD A DATA IN STREAMING SITE:

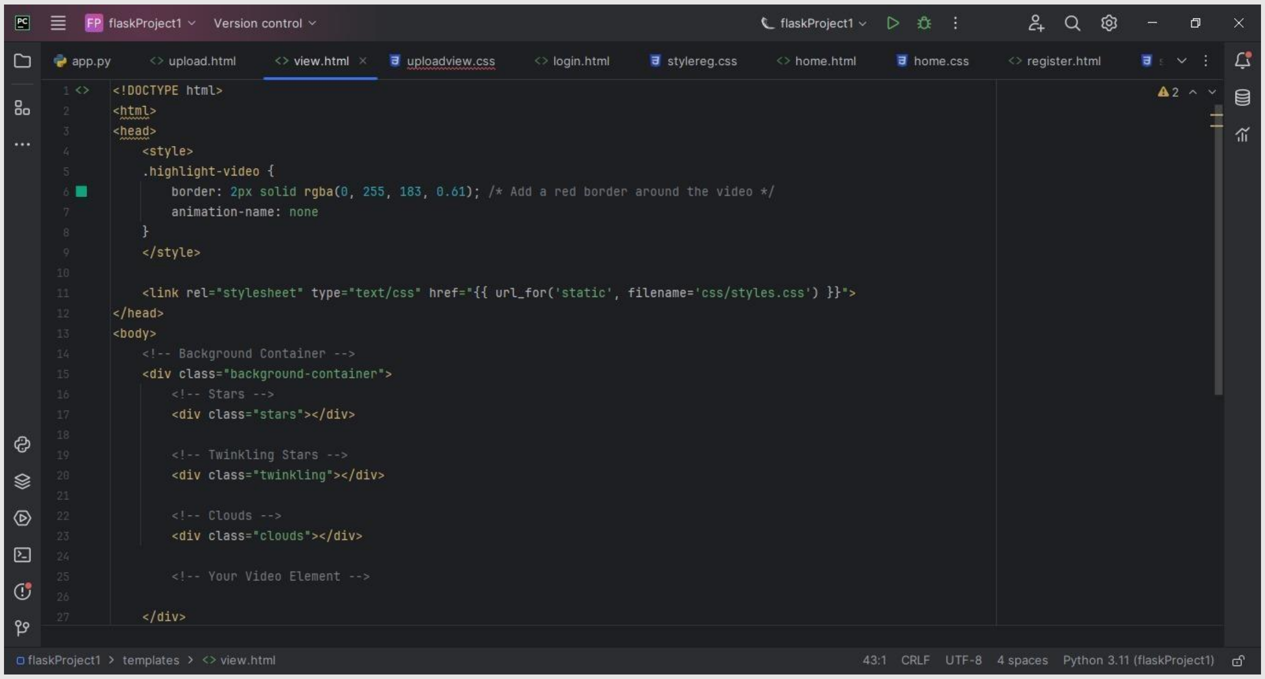
```
<!DOCTYPE html>
<html lang="en">
<head>
   <title>Upload Video</title>
   <link rel="stylesheet" type="text/css" href="{{ url_for('static',
filename='css/uploadview.css') }}">
</head>
<body>
<h1>Upload a Video</h1>
   <form method="POST" action="/upload" enctype="multipart/form-data">
      <input type="file" name="video" accept=".mp4">
      <input type="submit" value="Upload">
   </form>
</body>
</html>
```



```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Upload Video</title>
    <link rel="stylesheet" type="text/css" href="{{ url_for('static', filename='css/uploadview.css') }}">
</head>
<body>
<h1>Upload a Video</h1>
    <form method="POST" action="/upload" enctype="multipart/form-data">
        <input type="file" name="video" accept=".mp4">
        <input type="submit" value="Upload">
    </form>
</body>
</html>
```

10.HTML TO STREAM A DATA IN STREAMING SITE:

```
<!DOCTYPE html>
<html>
<head>
    <style>
    .highlight-video {
        border: 2px solid rgba(0, 255, 183, 0.61); /* Add a red border around the video */
        animation-name: none
    }
    </style>

    <link rel="stylesheet" type="text/css" href="{{ url_for('static', filename='css/styles.css')
}}">
</head>
<body>
    <!-- Background Container -->
    <div class="background-container">
        <!-- Stars -->
        <div class="stars"></div>
        <!-- Twinkling Stars -->
        <div class="twinkling"></div>

        <!-- Clouds -->
        <div class="clouds"></div>

        <!-- Your Video Element -->

    </div>

    <div class="video-container">
        <video class="highlight-video" width="640" height="480" controls >
            <source src="{{ url_for('send_video', filename=filename) }}" type="video/mp4">
        </video>
    </div>
    <!-- Your regular page content goes here -->
    <div class="page-content">
        <!-- More content, other sections, etc. -->
    </div>

</body>
</html>
```



```
<!DOCTYPE html>
<html>
<head>
    <style>
    .highlight-video {
        border: 2px solid rgba(0, 255, 183, 0.61); /* Add a red border around the video */
        animation-name: none
    }
    </style>

    <link rel="stylesheet" type="text/css" href="{{ url_for('static', filename='css/styles.css') }}">
</head>
<body>
    <!-- Background Container -->
    <div class="background-container">
        <!-- Stars -->
        <div class="stars"></div>

        <!-- Twinkling Stars -->
        <div class="twinkling"></div>

        <!-- Clouds -->
        <div class="clouds"></div>

        <!-- Your Video Element -->

    </div>
```

11.CSS CODE FOR BOTH UPLOAD AND STREAMING A VIDEO:

```
@keyframes move-background {
  from {
    -webkit-transform: translate3d(0px, 0px, 0px);
  }
  to {
    -webkit-transform: translate3d(1000px, 0px, 0px);
  }
}
@-webkit-keyframes move-background {
  from {
    -webkit-transform: translate3d(0px, 0px, 0px);
  }
  to {
    -webkit-transform: translate3d(1000px, 0px, 0px);
  }
}
@-moz-keyframes move-background {
  from {
    -webkit-transform: translate3d(0px, 0px, 0px);
  }
  to {
    -webkit-transform: translate3d(1000px, 0px, 0px);
  }
}
.background-container {
  position: fixed;
  top: 0;
  left: 0;
  bottom: 0;
  right: 0;
}
.stars {
  background: black url(https://s3-us-west-
2.amazonaws.com/s.cdpn.io/1231630/stars.png) repeat;
  position: absolute;
  top: 0;
  bottom: 0;
  left: 0;
  right: 0;
  display: block;
  z-index: 0;
}
.twinkling {
  width: 10000px;
  height: 100%;
  background: transparent url("https://s3-us-west-
2.amazonaws.com/s.cdpn.io/1231630/twinkling.png") repeat;
  background-size: 1000px 1000px;
  position: absolute;
  right: 0;
  top: 0;
  bottom: 0;
  z-index: 2;
  -moz-animation: move-background 70s linear infinite;
  -ms-animation: move-background 70s linear infinite;
  -o-animation: move-background 70s linear infinite;
  -webkit-animation: move-background 70s linear infinite;
  animation: move-background 70s linear infinite;
}
.clouds {
  width: 10000px;
  height: 100%;
  background: transparent url("https://s3-us-west-
2.amazonaws.com/s.cdpn.io/1231630/clouds_repeat.png") repeat;
  background-size: 1000px 1000px;
  position: absolute;
  right: 0;
  top: 0;
  bottom: 0;
  z-index: 3;

  -moz-animation: move-background 150s linear infinite;
  -ms-animation: move-background 150s linear infinite;
  -o-animation: move-background 150s linear infinite;
  -webkit-animation: move-background 150s linear infinite;
  animation: move-background 150s linear infinite;
}
.video-container {
    position: relative;
```

```
}
/* CSS rule for highlighting the video */
.highlight-video {
    border: 2px solid red;

video {
  height: 70vh;
  width: 70vh;
  position: absolute;
  z-index: 3;
  right: 20px;
}
```

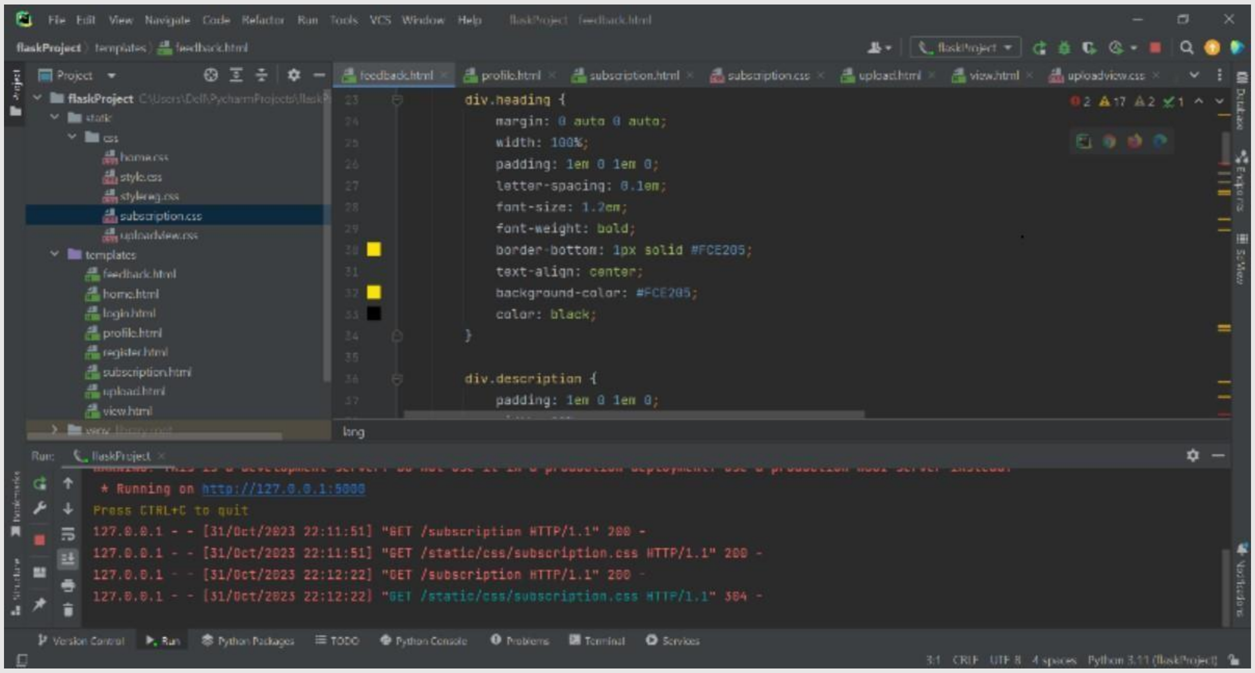
12. HTML Code for FEEDBACK From:

```
<!DOCTYPE html>
<lang html>
<head>
    <title>RESPONSIVE FEEDBACK FORM </title>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width,initial-scale=1,minimum-
scale=1.0,maximum-scale=1.0,user-scalable=no" />
    <style>
        body {
            margin: 0 auto;
            background-color: black;
        }
        div.formContainer {
            margin: 4em auto 4em auto;
            width: 80%;
            color: #FCE205;
            border: 1px solid #FCE205;
            border-radius: 2px;
        }
        div.heading {
            margin: 0 auto 0 auto;
            width: 100%;
            padding: 1em 0 1em 0;
            letter-spacing: 0.1em;
            font-size: 1.2em;
            font-weight: bold;
            border-bottom: 1px solid #FCE205;
            text-align: center;
            background-color: #FCE205;
            color: black;
        }
        div.description {
            padding: 1em 0 1em 0;
            width: 80%;
            margin: 0 auto 0 auto;
            text-align: center;
        }
        div.form {
            margin: 0 auto 0 auto;
            width: 100%;
        }
        div.form form {
            margin: 0 auto 0 auto;
            width: 80%;
        }
        div.form label {
            display: block;
            width: 80%;
            margin: 1em auto 1em auto;
            outline: none;
            color: #FCE205;
            font-weight: bolder;
            letter-spacing: 0.1;
        }
        div.form input {
            display: block;
            width: 80%;
            margin: 1em auto 1em auto;
            outline: none;
            color: #FCE205;
            padding: 1.2em 0 1em 1.2em;
            background-color: black;
            border: 0.5px solid #FCE205;
        }
        div.form textarea {
            display: block;
            width: 80%;
            margin: 1em auto 1em auto;
            outline: none;
            color: #FCE205;
            padding: 1.2em 0 1em 1.2em;
            background-color: black;
            border: 0.5px solid #FCE205;
            height: 8em;
            resize: none;
        }
        div.form button {
            outline: none;
            margin: 2em auto 2em auto;
            padding: 2em;
            cursor: pointer;
            border: none;
            display: block;
            width: 60%;
            text-align: center;
            border: 1px solid #FCE205;
        }
        div.form  input:active,
        div.form   input:focus,
        div.form select:active,
        div.form select:focus,
        div.form textarea:active,
        div.form textarea:focus,
        div.form button:active,
        div.form button:hover {
            box-shadow: 0 0 2px 2px white;
        }
        div.form button {
            background-color: black;
            color: #FCE205;
            font-weight: bolder;
            transition: all 0.2s linear;
        }
        @keyframes status {
            from {
                transform: scale(0);
            }
            to {
                transform: scale(1);
            }
        }
        div.statusActive {
            display:  block;
            width: 80%;
            margin: 1em auto 1em auto;
            padding: 1.2em 0 1em 1.2em;
            background-color: #FCE205;
            color: black;
            text-align: center;
            animation-name: status;
            animation-duration: 0.3s;
            transition: all 0.2s linear;
        }
        div.loadingActive {
            width: 90%;
            margin: 4em auto 4em auto;
            grid-template-columns: 33.3% 33.3% 33.3%;
            display: grid;
            transition: all linear 0.2s;
        }
        div.loadingActive div {
            padding: 0.5em;
            background-color: #FCE205;
            animation-name: rotatingDiv;
            animation-duration: 1s;
            animation-iteration-count: infinite;
        }
        @keyframes rotatingDiv {
            from {
                transform: rotate(-180deg);
            }
            to {
                transform: rotate(360deg);
            }
        }
        @media only screen and (min-width: 1224px) {
            div.formContainer {
```

```
                width: 500px;
            }
        }
        @media only screen and (min-width: 1824px) {
            div.formContainer {
                width: 500px;
            }
        }
        @media only screen and (min-device-width: 320px) and (max-device-width: 480px)
{
            div.formContainer {
                width: 80%;
            }
        }
    </style>
</head>
<body>
    <div class="formContainer">
        <div class="heading">
            FEEDBACK FORM
        </div>
        <div class="description">
            <p> Please notify your queries to get better experience.</p>
        </div>
        <div class="form">
            <form id="form">
                <label for="name">Name</label>
                <input name="name" type="text" id="name" />
                <label for="email">Email</label>
                <input name="email" type="email" id="email" />
                <label for="designation">Designation</label>
                <input name="designation" type="text" id="designation" />
                <label for="feedback">Feedback</label>
                <textarea name="feedback" id="feedback"></textarea>
                <button type="submit">SUBMIT</button>
            </form>
        </div>
        <div id="loading">
            <div>
            </div>
            <div>
        </div>
        <div>
        </div>
    </div>
    <div id="status">
    </div>
</div>
<script>
    class Feedback {
        constructor() {
            this.formEle = document.getElementById("form");
            this.emailEle = document.getElementById("email");
            this.designationEle = document.getElementById("designation");
            this.nameEle = document.getElementById("name");
            this.feedbackEle = document.getElementById("feedback");
            this.statusEle = document.getElementById("status");
            this.loading = document.getElementById("loading");
        }
        setStatus(content) {
            this.removeLoading();
            this.statusEle.classList.add("statusActive");
            this.statusEle.innerHTML = content;
        }
        removeStatus() {
            this.statusEle.classList.remove("statusActive");
            this.statusEle.innerHTML = ``;
        }
        addLoading() {
            this.loading.classList.add("loadingActive");
        }
        removeLoading() {
            this.loading.classList.remove("loadingActive");
        }
        isEmpty(value) {
            return (value.trim().length == 0) ? true : false;
        }
        invalidLength(value, len) {
            return (value.length > len) ? true : false;
        }
        validateEmail(email) {
```

```
        let re = /^(([^<>()[\]\\.,;:\s@\"]+(\.[^<>()[\]\\.,;:\s@\"]+)*)|(\".+\"))@((\[[0-
9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\])|(([a-zA-Z\-0-9]+\.)+[a-zA-Z]{2,}))$/;
        return re.test(email);
      }
      initialize() {
        this.formEle.onsubmit = (e) => {
          e.preventDefault();
          this.removeStatus();
          this.addLoading();
          this.validate();
        }
      }

      validate() {
        let email = this.emailEle.value.trim();
        let designation = this.designationEle.value.trim();
        let feedback = this.feedbackEle.value.trim();
        let name = this.nameEle.value.trim();
        let hasErrors = false;
        let errors = <h>ERROR</h>;
        if (this.isEmpty(name)) {
          errors = ${errors} <p> We need your name pal !!! </p>;
          hasErrors = true;
        }
        if (this.invalidLength(name, 59)) {
          errors = ${errors} <p>Name cant exceed 60 characters</p>;
          hasErrors = true;
        }
        if (!this.validateEmail(email)) {
          errors = ${errors} <p> Is that a valid email ? ? </p>;
          hasErrors = true;
        }
        if (this.isEmpty(email)) {
          errors = ${errors} <p> Email can not be empty </p>;
          hasErrors = true;
        }
        if (this.invalidLength(email, 319)) {
          errors = ${errors} <p>Email can not exceed 320 characters</p>;
          hasErrors = true;
        }
        if (this.isEmpty(designation)) {
                errors = ${errors} <p> Designation cant be empty </p>;
                hasErrors = true;
            }
            if (this.invalidLength(designation, 59)) {
                errors = ${errors} <p>Please fit your designation within 60 characters</p>;
                hasErrors = true;
            }
            if (this.isEmpty(feedback)) {
                errors = ${errors} <p> Irony .. !feedback cant be empty</p>;
                hasErrors = true;
            }
            if (this.invalidLength(feedback, 5999)) {
                errors = ${errors} <p>Thats a long feedback,sorry feedback can not exceed
6000 characters</p>;
                hasErrors = true;
            }
            if (hasErrors) {
                this.setStatus(errors);
            } else {
                let data = {
                        data: {
                            email: email,
                            designation: designation,
                            name: name,
                            feedback: feedback
                        }
                    }
                    // this.sendData(JSON.stringify(data));
            }
        }
        sendData(jsonData) {
            const url = "./php/AddFeedback.php";
            let xhttp = new XMLHttpRequest();
            let that = this;
            xhttp.onreadystatechange = function() {
                if (this.readyState == 4 && this.status == 200) {
                    if (this.responseText) {
                        that.setStatus("Thank you ! We got your feedback, we look forward to
more from you.");
                        that.formEle.reset();
                    } else {
```

```
                              that.setStatus("Are you sure you entered valid data? Dont worry there
might also be problem with our server. Why not give it another shot?");
                          }
                      }
                  };
                  xhttp.open("POST", url, true);
                  xhttp.setRequestHeader("Content-type", "application/x-www-form-
urlencoded");
                  xhttp.send(data=${jsonData});
              }

          }
          const fb = new Feedback();
          fb.initialize();
      </script>
  </body>
  </lang html>
```



```
div.heading {
    margin: 0 auto 0 auto;
    width: 100%;
    padding: 1em 0 1em 0;
    letter-spacing: 0.1em;
    font-size: 1.2em;
    font-weight: bold;
    border-bottom: 1px solid #FCE205;
    text-align: center;
    background-color: #FCE205;
    color: black;
}

div.description {
    padding: 1em 0 1em 0;
```

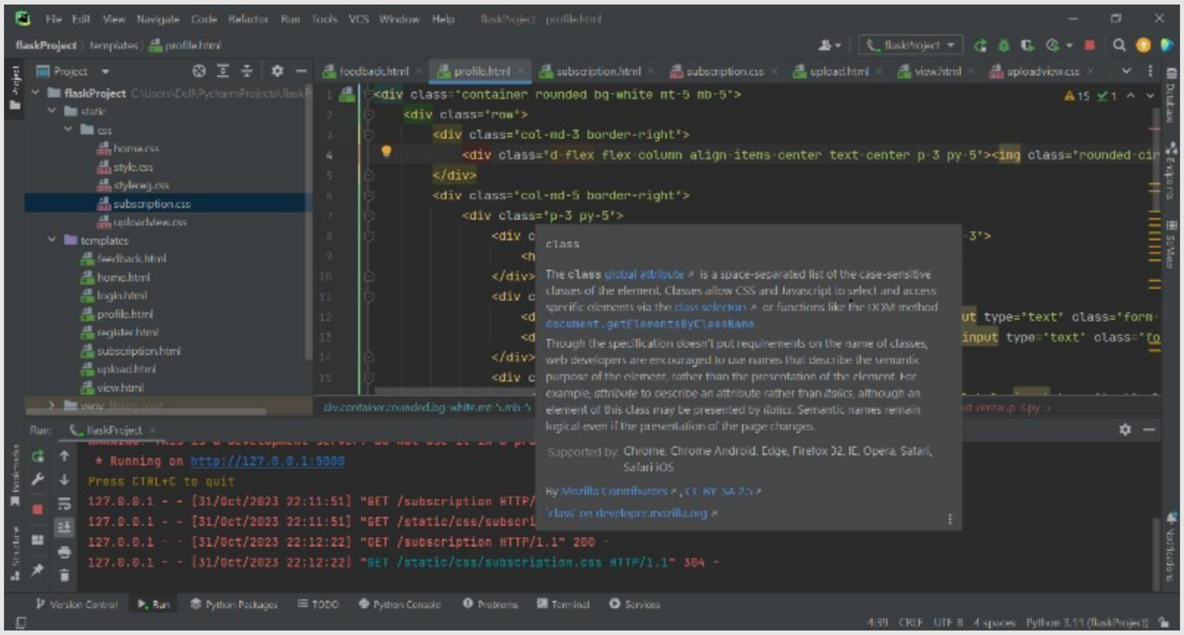
```
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
127.0.0.1 - - [31/Oct/2023 22:11:51] "GET /subscription HTTP/1.1" 200 -
127.0.0.1 - - [31/Oct/2023 22:11:51] "GET /static/css/subscription.css HTTP/1.1" 200 -
127.0.0.1 - - [31/Oct/2023 22:12:22] "GET /subscription HTTP/1.1" 200 -
127.0.0.1 - - [31/Oct/2023 22:12:22] "GET /static/css/subscription.css HTTP/1.1" 304 -
```

13. HTML code to create the Profile page:

```
<div class="container rounded bg-white mt-5 mb-5">

  <div class="row">

    <div class="col-md-3 border-right">
```

```
                        <div class="d-flex flex-column align-items-center text-center p-3 py-5"><img
class="rounded-circle mt-5" width="150px"
src="https://st3.depositphotos.com/15648834/17930/v/600/depositphotos_179308454-stock-
illustration-unknown-person-silhouette-glasses-profile.jpg"><span class="font-weight-
bold">Edogaru</span><span class="text-black-50">edogaru@mail.com.my</span><span>
</span></div>

                    </div>

                    <div class="col-md-5 border-right">

                        <div class="p-3 py-5">

                            <div class="d-flex justify-content-between align-items-center mb-3">

                                <h4 class="text-right">Profile Settings</h4>

                            </div>

                            <div class="row mt-2">

                                <div class="col-md-6"><label class="labels">Name</label><input type="text"
class="form-control" placeholder="first name" value=""></div>

                                <div class="col-md-6"><label class="labels">Surname</label><input
type="text" class="form-control" value="" placeholder="surname"></div>

                            </div>

                            <div class="row mt-3">

                                <div class="col-md-12"><label class="labels">Mobile Number</label><input
type="text" class="form-control" placeholder="enter phone number" value=""></div>

                                <div class="col-md-12"><label class="labels">Address Line 1</label><input
type="text" class="form-control" placeholder="enter address line 1" value=""></div>

                                <div class="col-md-12"><label class="labels">Address Line 2</label><input
type="text" class="form-control" placeholder="enter address line 2" value=""></div>

                                <div class="col-md-12"><label class="labels">Postcode</label><input
type="text" class="form-control" placeholder="enter address line 2" value=""></div>

                                <div class="col-md-12"><label class="labels">State</label><input
type="text" class="form-control" placeholder="enter address line 2" value=""></div>

                                <div class="col-md-12"><label class="labels">Area</label><input type="text"
class="form-control" placeholder="enter address line 2" value=""></div>
```

```
                        <div class="col-md-12"><label class="labels">Email ID</label><input
type="text" class="form-control" placeholder="enter email id" value=""></div>

                        <div class="col-md-12"><label class="labels">Education</label><input
type="text" class="form-control" placeholder="education" value=""></div>

                    </div>

                    <div class="row mt-3">

                        <div class="col-md-6"><label class="labels">Country</label><input
type="text" class="form-control" placeholder="country" value=""></div>

                        <div class="col-md-6"><label class="labels">State/Region</label><input
type="text" class="form-control" value="" placeholder="state"></div>

                    </div>

                    <div class="mt-5 text-center"><button class="btn btn-primary profile-button"
type="button">Save Profile</button></div>

                </div>

            </div>

        </div>

    </div>

    </div>

    </div>
```



```
<div class="container rounded bg-white mt-5 mb-5">
    <div class="row">
        <div class="col-md-3 border-right">
            <div class="d-flex flex-column align-items-center text-center p-3 py-5"><img class="rounded-cir
        </div>
        <div class="col-md-5 border-right">
            <div class="p-3 py-5">
                <div c                                        -3">
                    <h
                </div>
                <div c
                    <d                                        ut type="text" class="form-
                    <d                                        input type="text" class="fo
                </div>
                <div c
```

class

The class global attribute is a space-separated list of the case-sensitive classes of the element. Classes allow CSS and Javascript to select and access specific elements via the class selectors or functions like the DOM method document.getElementsByClassName.

Though the specification doesn't put requirements on the name of classes, web developers are encouraged to use names that describe the semantic purpose of the element, rather than the presentation of the element. For example, attribute to describe an attribute rather than italics, although an element of this class may be presented by italics. Semantic names remain logical even if the presentation of the page changes.

Supported by: Chrome, Chrome Android, Edge, Firefox 32, IE, Opera, Safari, Safari iOS

By Mozilla Contributors, CC BY-SA 2.5

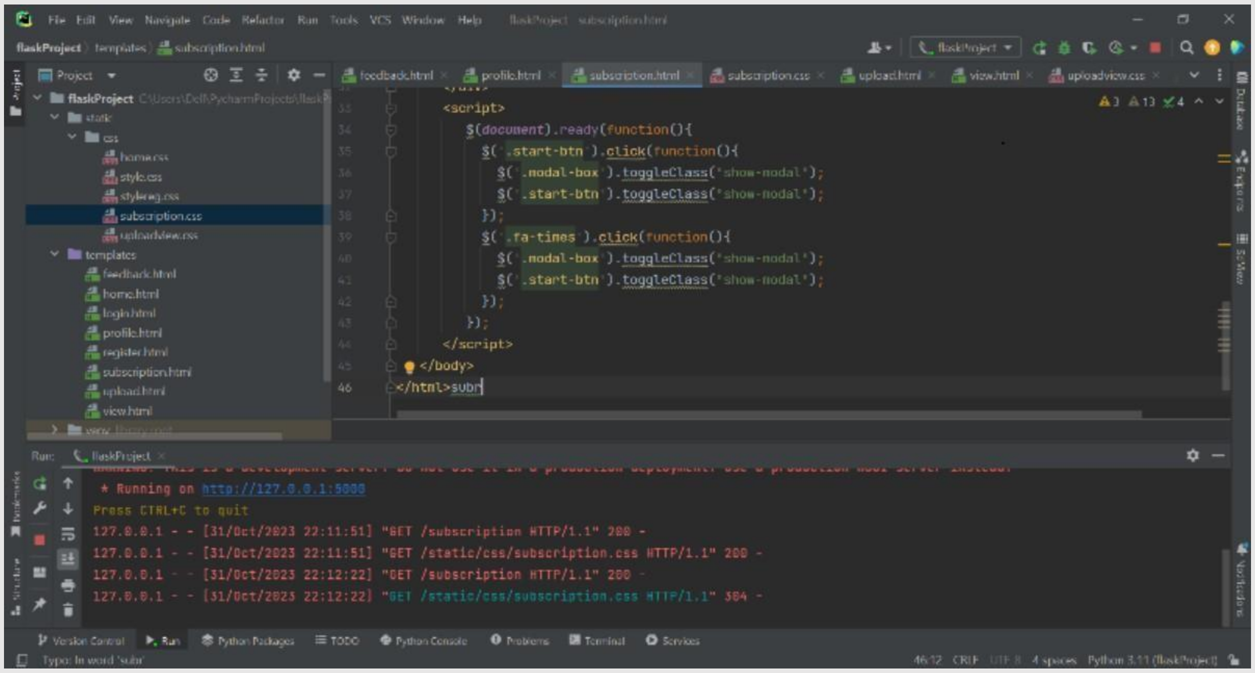
'class' on developer.mozilla.org

```
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
127.0.0.1 - - [31/Oct/2023 22:11:51] "GET /subscription HTTP/
127.0.0.1 - - [31/Oct/2023 22:11:51] "GET /static/css/subscri
127.0.0.1 - - [31/Oct/2023 22:12:22] "GET /subscription HTTP/1.1" 200 -
127.0.0.1 - - [31/Oct/2023 22:12:22] "GET /static/css/subscription.css HTTP/1.1" 304 -
```

14. HTML code to create subscription page:

```
<!DOCTYPE html>
<!-- Created By CodingNepal -->
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title>Sliding Modal Box Style</title>
    <link rel="stylesheet" type="text/css" href="{{ url_for('static',
filename='css/subscription.css') }}">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/5.15.3/css/all.min.css"/>
    <script src="https://code.jquery.com/jquery-3.4.1.js"></script>
  <body>
    <div class="start-btn">
      <a href="#">View Modal</a>
    </div>
    <div class="center modal-box">
      <div class="fas fa-times"></div>
      <div class="fas fa-envelope icon1"></div>
      <header>Don't miss updates from us!</header>
      <p>
        Lorem ipsum dolor sit amet, consectetur adipisicing elit. Odit, suscipit, pers
      </p>
      <form action="">
        <div class="fas fa-envelope icon2"></div>
        <input type="email" required placeholder="abc@example.com">
```

```
          <button>Subscribe</button>
        </form>
        <div class="icons">
          <i class="fab fa-facebook-f"></i>
          <i class="fab fa-twitter"></i>
          <i class="fab fa-instagram"></i>
          <i class="fab fa-youtube"></i>
        </div>
      </div>
      <script>
        $(document).ready(function(){
          $('.start-btn').click(function(){
            $('.modal-box').toggleClass("show-modal");
            $('.start-btn').toggleClass("show-modal");
          });
          $('.fa-times').click(function(){
            $('.modal-box').toggleClass("show-modal");
            $('.start-btn').toggleClass("show-modal");
          });
        });
      </script>
    </body>
</html>
```



```
        <script>
            $(document).ready(function(){
              $('.start-btn').click(function(){
                $('.modal-box').toggleClass('show-modal');
                $('.start-btn').toggleClass('show-modal');
              });
              $('.fa-times').click(function(){
                $('.modal-box').toggleClass('show-modal');
                $('.start-btn').toggleClass('show-modal');
              });
            });
        </script>
    </body>
</html>subr
```

```
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
127.0.0.1 - - [31/Oct/2023 22:11:51] "GET /subscription HTTP/1.1" 200 -
127.0.0.1 - - [31/Oct/2023 22:11:51] "GET /static/css/subscription.css HTTP/1.1" 200 -
127.0.0.1 - - [31/Oct/2023 22:12:22] "GET /subscription HTTP/1.1" 200 -
127.0.0.1 - - [31/Oct/2023 22:12:22] "GET /static/css/subscription.css HTTP/1.1" 304 -
```

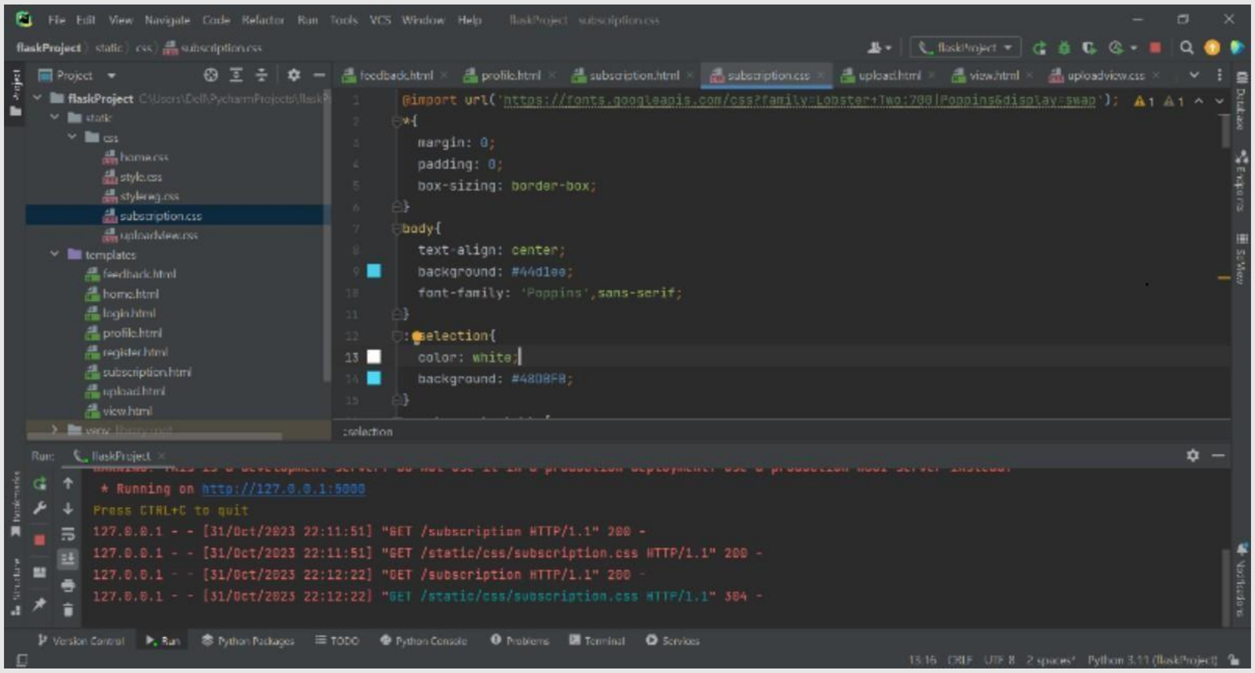
15. synchronizing subscription with css code:

```
@import
url('https://fonts.googleapis.com/css?family=Lobster+Two:700|Poppins&display=swap');

*{
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}
body{
  text-align: center;
  background: #44d1ee;
  font-family: 'Poppins',sans-serif;
}
::selection{
  color: white;
  background: #48DBFB;
```

```
}
.center,.start-btn{
  position: absolute;
  top: 50%;
  left: 50%;
  transform: translate(-50%, -50%);
}
.start-btn a{
  font-size: 25px;
  background: white;
  color: #48DBFB;
  padding: 10px 15px;
  border-radius: 5px;
  text-decoration: none;
  text-transform: uppercase;
  font-weight: 600;
  letter-spacing: 1px;
  box-shadow: 5px 5px 15px rgba(0,0,0,.1);
}
.modal-box{
  top: 40%;
  opacity: 0;
  visibility: hidden;
  background: white;
  height: auto;
  width: 400px;
  padding: 30px;
  border-radius: 5px;
  border: 1px solid #50dcfb;
  box-shadow: 5px 5px 30px rgba(0,0,0,.2);
}
.start-btn.show-modal{
  opacity: 0;
  visibility: hidden;
}
.modal-box.show-modal{
  top: 50%;
  opacity: 1;
  visibility: visible;
  transition: .4s;
}
.modal-box .fa-times{
  position: absolute;
  top: 0;
  right: 0;
  background: #48DBFB;
  height: 45px;
  width: 50px;
  line-height: 40px;
  color: white;
  font-size: 18px;
  border-radius: 0 5px 0 50px;
  padding-left: 13px;
  cursor: pointer;
```

```
}
.fa-times:hover{
  font-size: 22px;
}
.modal-box .icon1{
  font-size: 60px;
  background: #48DBFB;
  height: 120px;
  width: 120px;
  color: white;
  border-radius: 50%;
  line-height: 120px;
  text-align: center;
  margin-bottom: 10px;
}
.modal-box header{
  font-size: 31px;
  font-family: 'Lobster Two';
  margin-bottom: 10px;
}
.modal-box p{
  /* margin-bottom: 10px; */
  line-height: 20px;
  color: grey;
}
form input, form button{
  height: 50px;
  width: 95%;
  border-radius: 3px;
}
form .icon2{
  position: absolute;
  height: 50px;
  width: 50px;
  margin-top: 15px;
  background: #48DBFB;
  color: white;
  line-height: 50px;
  font-size: 24px;
  border-radius: 5px 0 0 5px;
}
form input{
  margin-top: 15px;
  padding: 0 65px;
  font-size: 18px;
  outline: none;
  border: 2px solid #1ed2fa;
  caret-color: #48DBFB;
}
input::placeholder{
  color: #8c8c8c;
}
form input:focus{
  box-shadow: 0 0 15px #82e6fc,
              0 0 25px #b4f0fd,
              0 0 35px #ffffff;
}
form button{
  margin-top: 15px;
  background: #48DBFB;
  color: white;
  font-size: 25px;
  border: 1px solid #1ed2fa;
  letter-spacing: 1px;
  cursor: pointer;
  outline: none;
  transition: .3s;
}
form button:hover{
  background: #1ed2fa;
  border: 1px solid #05cdfa;
  letter-spacing: 2px;
}
.modal-box .icons{
  margin-top: 25px;
}
.icons i{
  font-size: 22px;
  margin: 0 7px;
  color: #48DBFB;
  cursor: pointer;
```

```
}
.icons i:hover{
  transform: scale(1.1);
  color: #05cdfa;
}
```

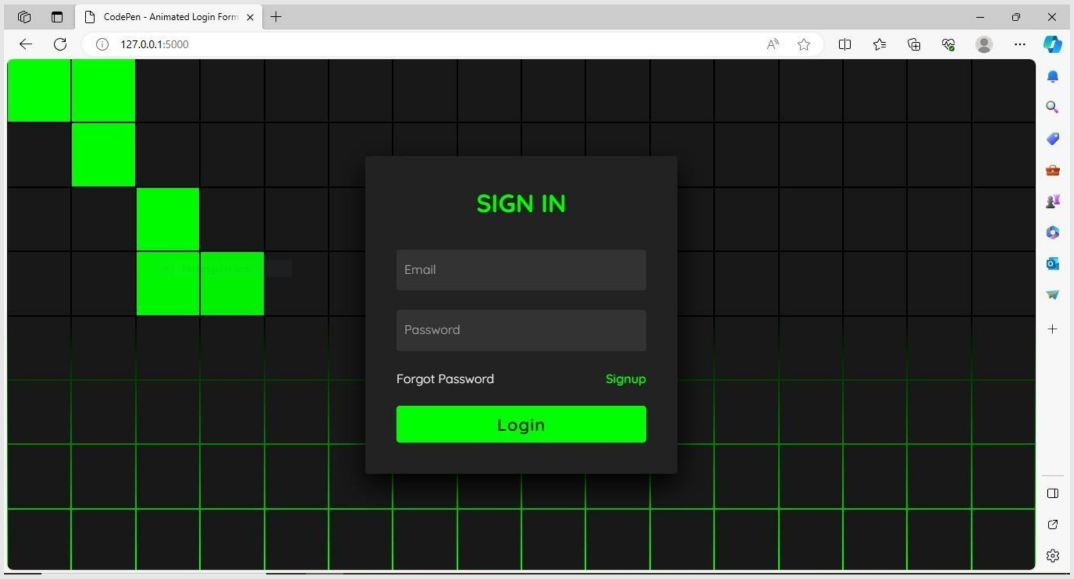


```
@import url('https://fonts.googleapis.com/css?family=Lobster+Two:700|Poppins&display=swap');
*{
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}
body{
  text-align: center;
  background: #44d1ee;
  font-family: 'Poppins',sans-serif;
}
::selection{
  color: white;
  background: #4808F8;
}
```

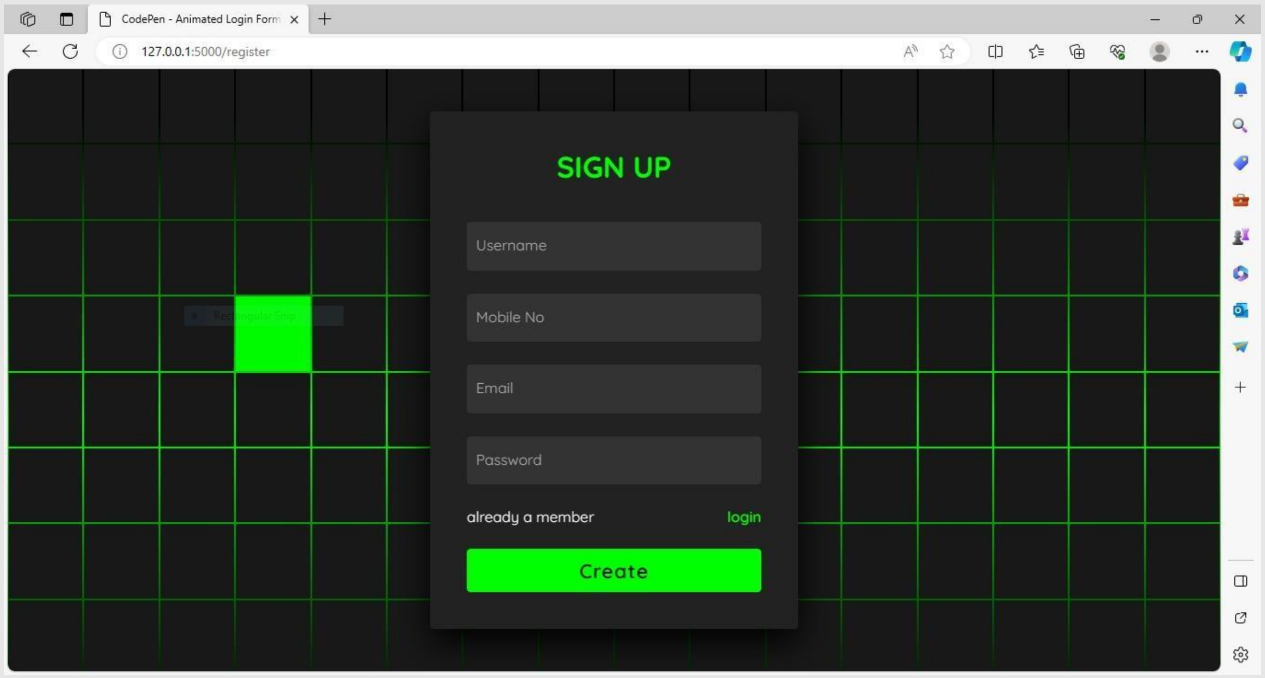
```
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
127.0.0.1 - - [31/Oct/2023 22:11:51] "GET /subscription HTTP/1.1" 200 -
127.0.0.1 - - [31/Oct/2023 22:11:51] "GET /static/css/subscription.css HTTP/1.1" 200 -
127.0.0.1 - - [31/Oct/2023 22:12:22] "GET /subscription HTTP/1.1" 200 -
127.0.0.1 - - [31/Oct/2023 22:12:22] "GET /static/css/subscription.css HTTP/1.1" 304 -
```

FINAL OUTPUT FOR ALL CODE:

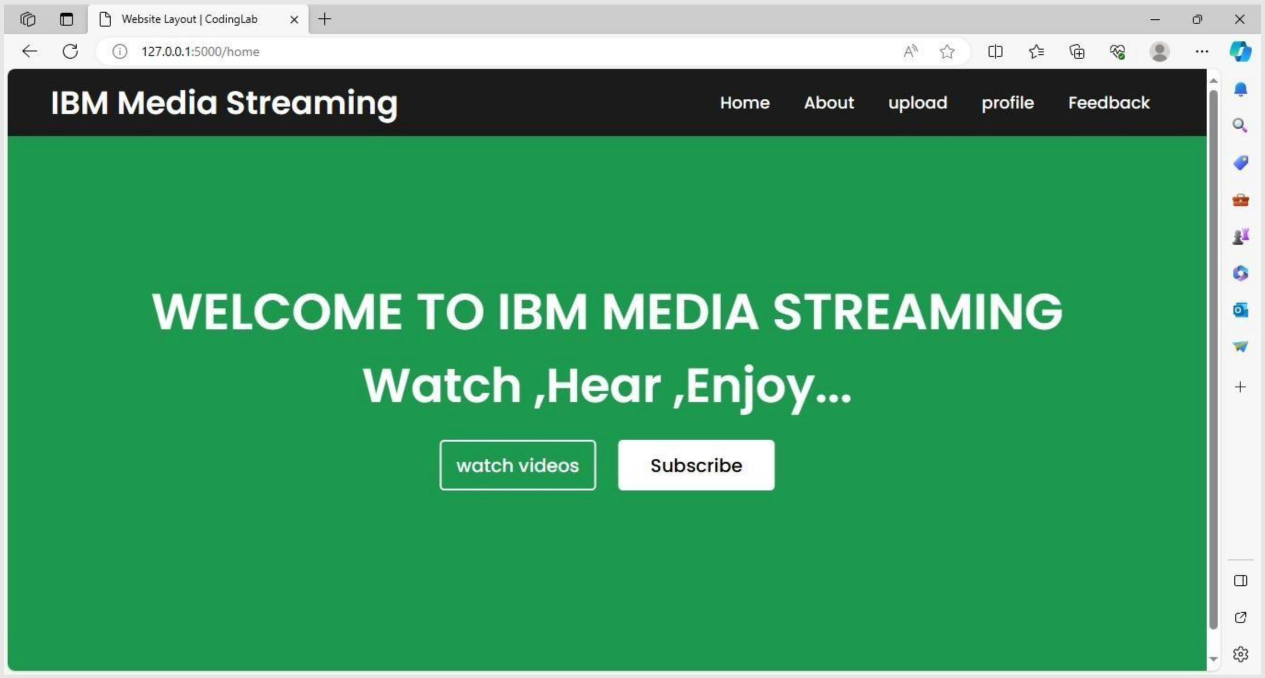
1. LOGIN PAGE TO ENTER INTO THE HOME PAGE:



2. SIGN UP PAGE TO CREATE AN ACCOUNT TO BE LOGGED IN:

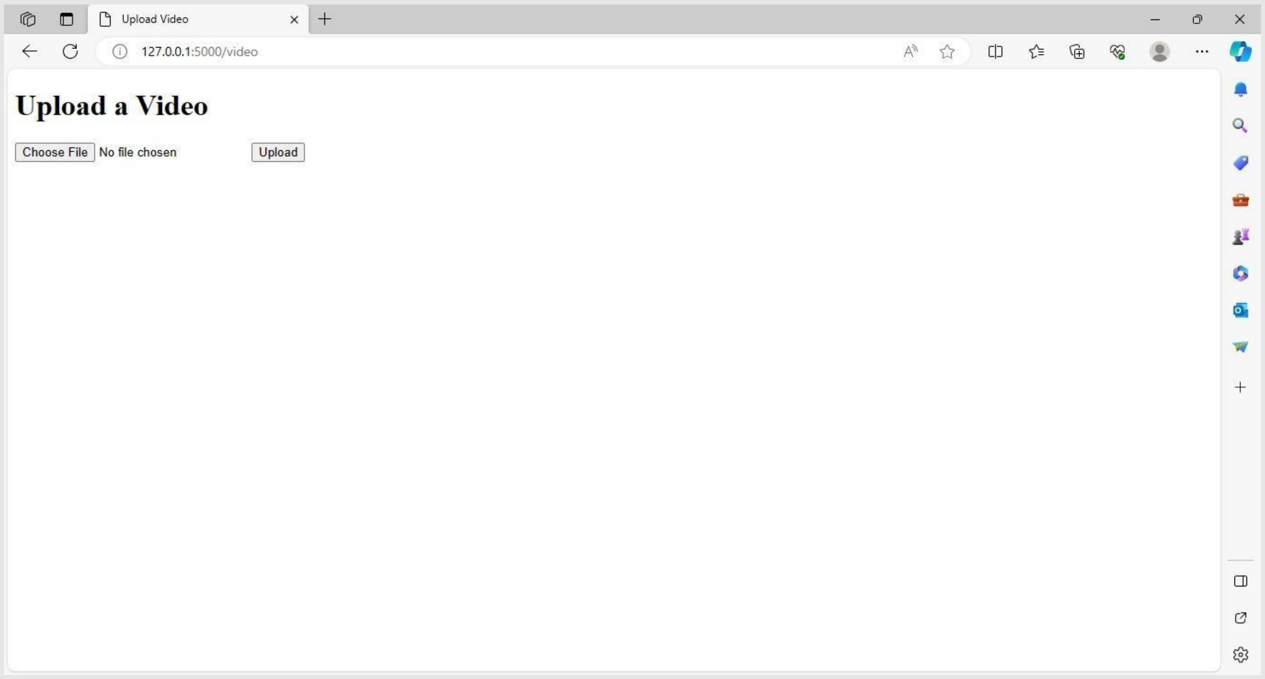


3. HOME PAGE INTERFACE FOR MEDIA STREAMING:

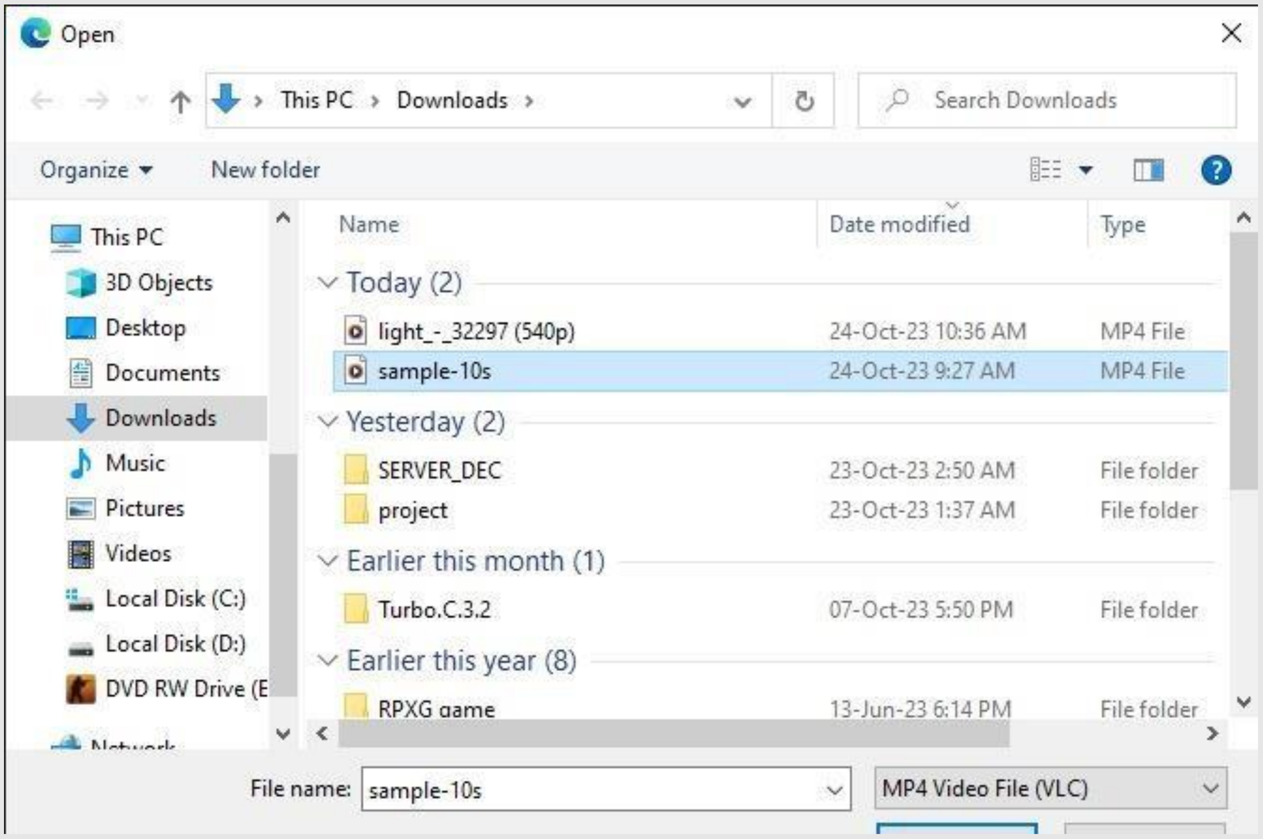


4. STEPS TO UPLOAD A FILE:

4.1) FIRST CLICK CHOOSE FILE CHECK BOX:



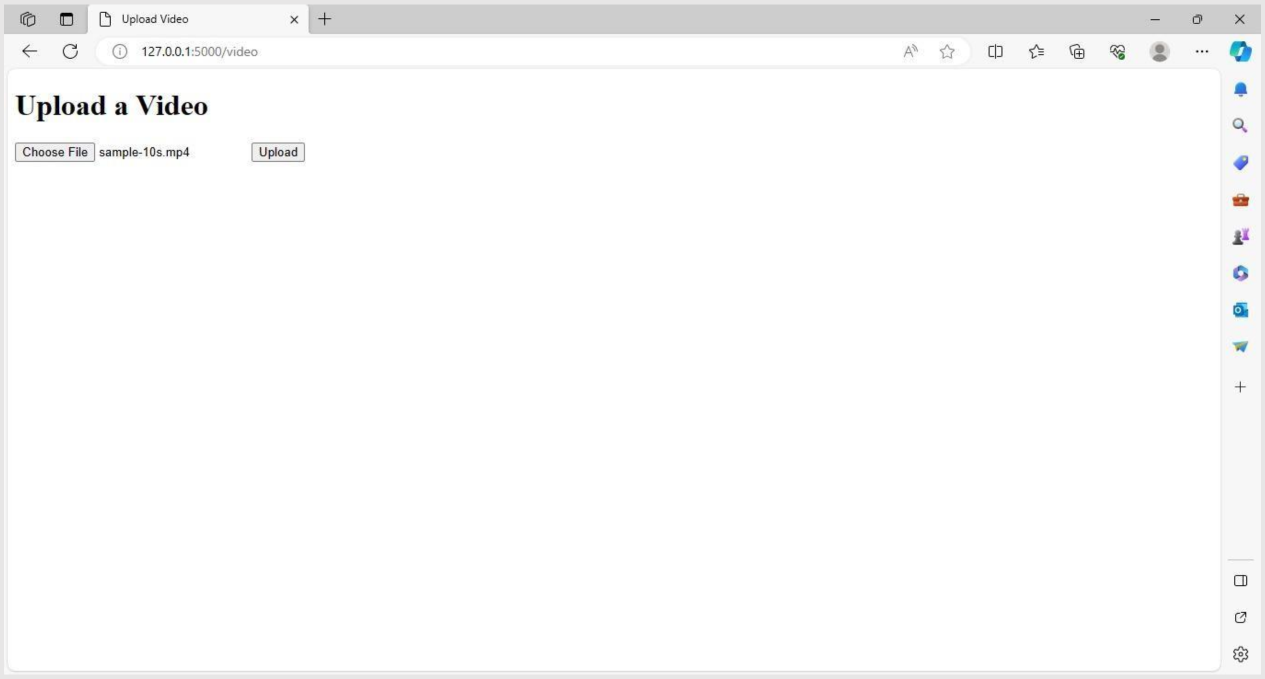
4.2)CHOOSE THE FILE AND CLICK OPEN TO CONTINUE:



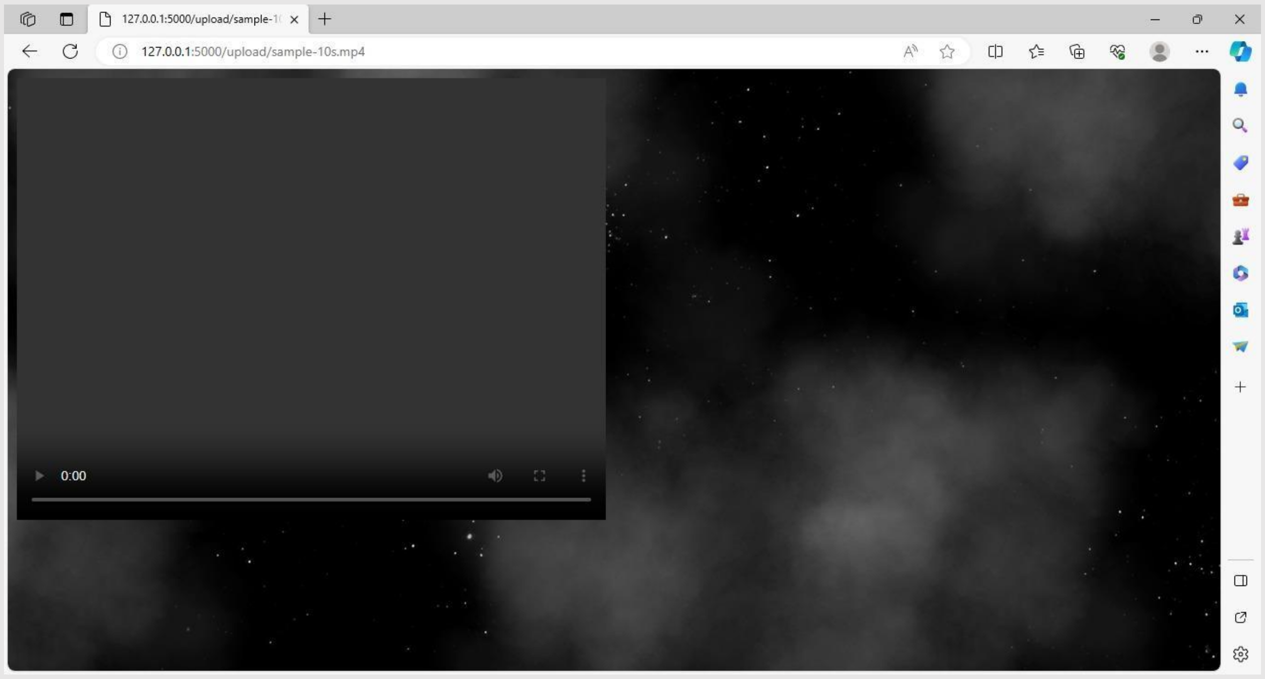
Name	Date modified	Type
Today (2)		
light_-_32297 (540p)	24-Oct-23 10:36 AM	MP4 File
sample-10s	24-Oct-23 9:27 AM	MP4 File
Yesterday (2)		
SERVER_DEC	23-Oct-23 2:50 AM	File folder
project	23-Oct-23 1:37 AM	File folder
Earlier this month (1)		
Turbo.C.3.2	07-Oct-23 5:50 PM	File folder
Earlier this year (8)		
RPXG game	13-Jun-23 6:14 PM	File folder

File name: sample-10s

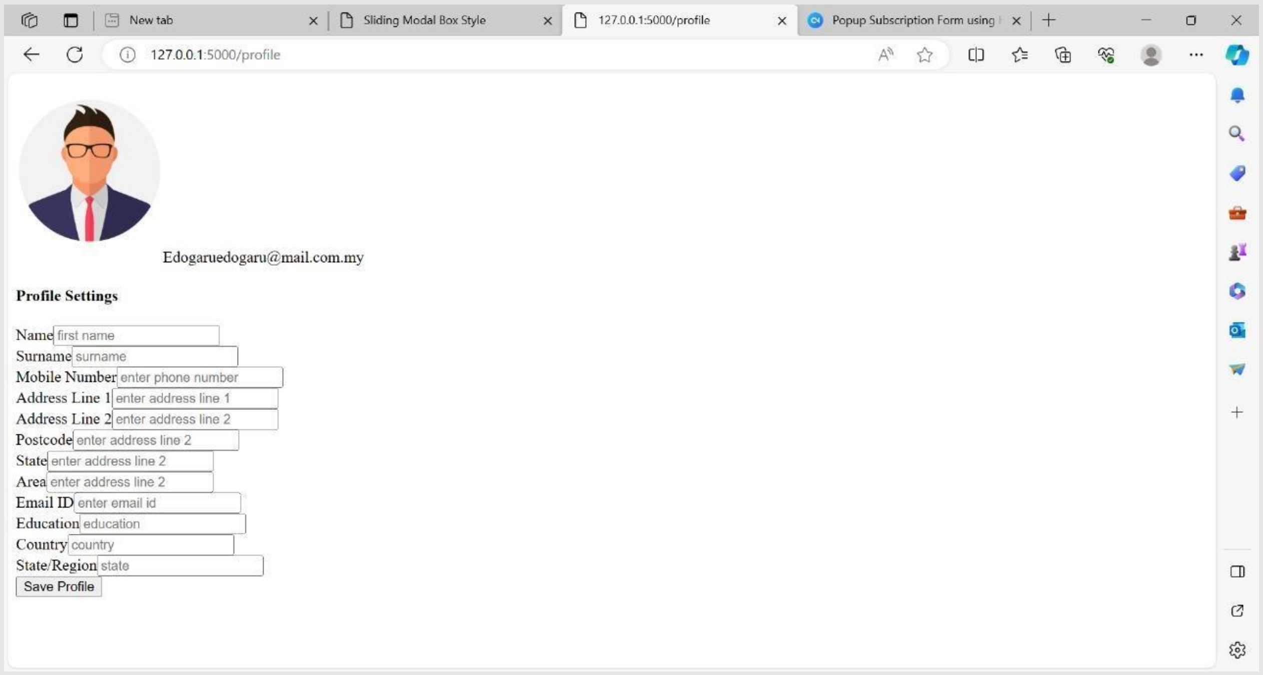
4.3) CLICK THE UPLOAD CHECK BOX TO UPLOAD A VIDEO TO BE STREAMED:



5. STREAMING MEDIA INTERFACE:



6. PROFILE PAGE FOR USERS:



127.0.0.1:5000/profile

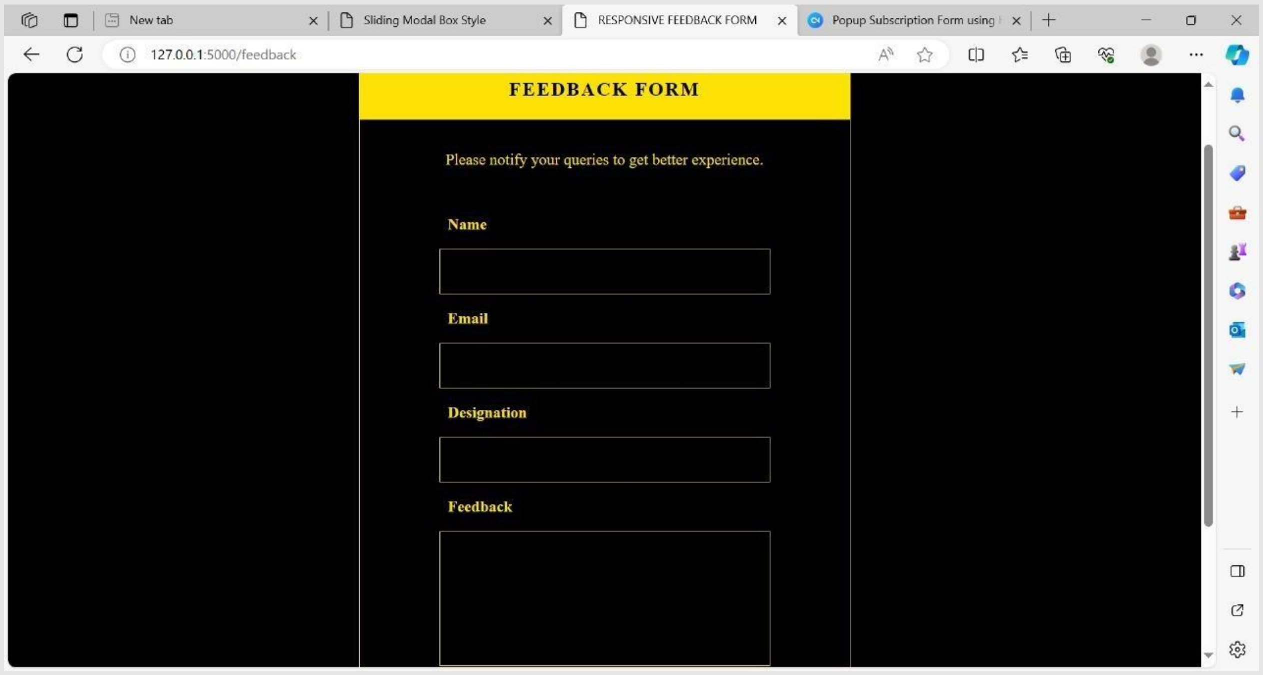
Edogaruedogaru@mail.com.my

Profile Settings

Name	first name
Surname	surname
Mobile Number	enter phone number
Address Line 1	enter address line 1
Address Line 2	enter address line 2
Postcode	enter address line 2
State	enter address line 2
Area	enter address line 2
Email ID	enter email id
Education	education
Country	country
State/Region	state

Save Profile

7. FEEDBACK PAGE FOR USERS:



127.0.0.1:5000/feedback

FEEDBACK FORM

Please notify your queries to get better experience.

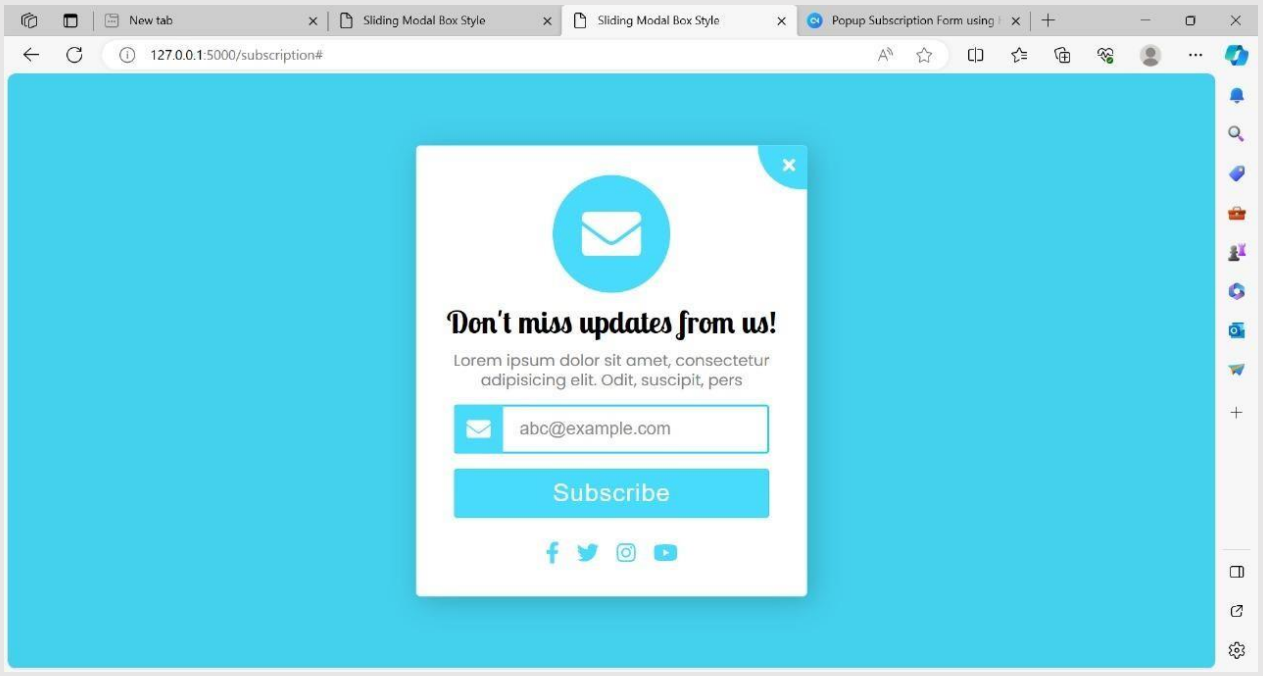
Name

Email

Designation

Feedback

8. SUBSCRIPTION PAGE FOR USERS:


New tab
Sliding Modal Box Style
Sliding Modal Box Style
Popup Subscription Form using
127.0.0.1:5000/subscription#
Don't miss updates from us!
Lorem ipsum dolor sit amet, consectetur adipisicing elit. Odit, suscipit, pers
abc@example.com
Subscribe

CONCLUSION:

Media streaming is a technology that allows the continuous delivery of multimedia content, such as video and audio, over the internet. It involves the preparation, storage, and efficient delivery of media files using various streaming protocols. Media streaming enables real-time or on-demand access to content, and it plays a significant role in modern entertainment, communication, and information distribution, making it a fundamental aspect of the digital age.

----------------X--------------x--------------x------------------