

## App.py:

```
from flask import Flask, render_template, request, redirect, url_for, session, flash
from flask_session import Session
from werkzeug.security import generate_password_hash, check_password_hash
import json
import re
import time
import os

app = Flask(__name__)
app.config['SESSION_TYPE'] = 'filesystem'
app.config['SECRET_KEY'] = 'your_secret_key'
Session(app)

DATA_FILE = 'data.json'

def load_data():
    if os.path.exists(DATA_FILE):
        with open(DATA_FILE, 'r') as f:
            return json.load(f)
    return {'users': [], 'email_timers': {}}
```

```
def save_data(data):
    with open(DATA_FILE, 'w') as f:
        json.dump(data, f)
```

```
data = load_data()
users = data['users']
email_timers = data['email_timers']
```

```

def email_exists(email_timers, email):

    return any(timer['email'] == email for timer in email_timers)


def get_user_by_email(email):

    for user in users:

        if user['email'] == email:

            return user

    return None


@app.route('/signup', methods=['GET', 'POST'])
def signup():

    if request.method == 'POST':

        email = request.form['email']

        password = request.form['password']


        if get_user_by_email(email) is not None:

            flash('Email already exists..Please log in or use another email..', 'error')

            return redirect(url_for('signup'))


        hashed_password = generate_password_hash(password)

        new_user = {'email': email, 'password': hashed_password}

        users.append(new_user)

        data['users'] = users

        save_data(data)

        flash('Account created successfully..Please log in..', 'success')

        return redirect(url_for('login'))


    return render_template('signup.html')


@app.route('/login', methods=['GET', 'POST'])
def login():

```

```

if request.method == 'POST':

    email = request.form['email']

    password = request.form['password']

    user = get_user_by_email(email)

    if user is None or not check_password_hash(user['password'], password):

        flash('Invalid email or password..Please try again..', 'error')

        return redirect(url_for('login'))

    session['user_email'] = email

    flash('Logged in successfully..', 'success')

    return redirect(url_for('index'))

return render_template('login.html')

@app.route('/logout')
def logout():

    session.pop('user_email', None)

    flash('Logged out successfully..', 'success')

    return redirect(url_for('login'))

@app.route('/', methods=['GET', 'POST'])
def index():

    if 'user_email' not in session:

        return redirect(url_for('login'))

    user_email = session['user_email']

    user_email_timers = email_timers.get(user_email, [])

    if request.method == 'POST':

        email_input = request.form['email']

```

```
emails = re.split(r'[;, \s]\s*', email_input.strip())
```

```
for email in emails:
```

```
    if not email_exists(user_email_timers, email):
```

```
        user_email_timers.append({'email': email, 'status': 'Ready', 'end_time': 0})
```

```
email_timers[user_email] = user_email_timers
```

```
data['email_timers'] = email_timers
```

```
save_data(data)
```

```
return render_template('index.html', email_timers=user_email_timers)
```

```
@app.route('/start_timer/<int:email_timer_index>')
```

```
def start_timer(email_timer_index):
```

```
    if 'user_email' not in session:
```

```
        return redirect(url_for('login'))
```

```
    user_email = session['user_email']
```

```
    user_email_timers = email_timers[user_email]
```

```
    email_timer = user_email_timers[email_timer_index]
```

```
    email_timer['status'] = 'InUse'
```

```
    email_timer['end_time'] = int(time.time()) + 11700
```

```
    email_timers[user_email] = user_email_timers
```

```
    data['email_timers'] = email_timers
```

```
    save_data(data)
```

```
    return redirect(url_for('index'))
```

```
@app.route('/delete_timer/<int:email_timer_index>')
```

```
def delete_timer(email_timer_index):
```

```
if 'user_email' not in session:
    return redirect(url_for('login'))

user_email = session['user_email']
user_email_timers = email_timers[user_email]
del user_email_timers[email_timer_index]
email_timers[user_email] = user_email_timers
data['email_timers'] = email_timers
save_data(data)

return redirect(url_for('index'))

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

### index.html:

```
<!DOCTYPE html>

<html>

<head>

    <title>Email Timer</title>

    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">

    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>

    <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>

    <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>

    <script>

        function formatTime(seconds) {

            let hours = Math.floor(seconds / 3600);

            let minutes = Math.floor((seconds % 3600) / 60);

            let remainingSeconds = seconds % 60;
```

```

        return `${hours.toString().padStart(2, '0')}:${minutes.toString().padStart(2,
'0')}:${seconds.toString().padStart(2, '0')}`;
    }

```

```

function updateTimers() {
    $('.timer').each(function() {
        let endTime = parseInt($(this).data('end-time'));
        let currentTime = Math.floor(Date.now() / 1000);
        let timeLeft = endTime - currentTime;
        if (timeLeft > 0) {
            $(this).text(formatTime(timeLeft));
        } else {
            $(this).text('00:00:00');
            $(this).siblings('.status').text('Ready');
            $(this).siblings('.actions').find('.start-timer').show();
        }
    });
}

```

```

$(document).ready(function() {
    setInterval(updateTimers, 1000);
});

```

```

</script>

```

```

</head>

```

```

<body>

```

```

<div class="container">
    {% with messages = get_flashed_messages(with_categories=true) %}
    {% if messages %}
    {% for category, message in messages %}
    <div class="alert alert-{{ category }}">{{ message }}</div>
    {% endfor %}

```

```

{% endif %}

{% endwith %}

<a href="{{ url_for('logout') }}" class="btn btn-secondary">Logout</a>

<form method="POST" action="{{ url_for('index') }}">

    <div class="form-group">

        <label for="email">Enter email addresses (separated by commas, semicolons, or
spaces):</label>

        <textarea class="form-control" name="email" rows="3" required></textarea>

    </div>

    <button type="submit" class="btn btn-primary">Add Email</button>

</form>

<table class="table">

    <thead>

        <tr>

            <th>Email</th>

            <th>Status</th>

            <th>Time Left</th>

            <th>Actions</th>

        </tr>

    </thead>

    <tbody>

        {% for email_timer in email_timers %}

        <tr>

            <td>{{ email_timer.email }}</td>

            <td class="status">{{ email_timer.status }}</td>

            <td class="timer" data-end-time="{{ email_timer.end_time }}"></td>

            <td class="actions">

                {% if email_timer.status == 'Ready' %}

                <a href="{{ url_for('start_timer', email_timer_index=loop.index0) }}" class="btn btn-
primary start-timer">Start Timer</a>

                {% else %}

```

```

        <a href="{{ url_for('start_timer', email_timer_index=loop.index0) }}" class="btn btn-
primary start-timer" style="display:none;">Start Timer</a>

        {% endif %}

        <a href="{{ url_for('delete_timer', email_timer_index=loop.index0) }}" class="btn btn-
danger">Delete</a>

    </td>

</tr>

{% endfor %}

</tbody>

</table>

</div>

</body>

</html>

```

## signup.html:

```

<!DOCTYPE html>
<html>
<head>
    <title>Sign Up</title>
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
</head>
<body>
    <div class="container">
        {% with messages = get_flashed_messages(with_categories=true) %}
        {% if messages %}
        {% for category, message in messages %}
        <div class="alert alert-{{ category }}">{{ message }}</div>
        {% endfor %}
        {% endif %}
        {% endwith %}
        <h1>Sign Up</h1>
        <form method="POST" action="{{ url_for('signup') }}">
            <div class="form-group">
                <label for="email">Email:</label>
                <input type="email" class="form-control" name="email" required>
            </div>
            <div class="form-group">
                <label for="password">Password:</label>
                <input type="password" class="form-control" name="password" required>
            </div>
            <button type="submit" class="btn btn-primary">Sign Up</button>

```



```

        </form>
        <p>Already have an account? <a href="{{ url_for('login') }}">Login</a></p>
    </div>
</body>
</html>

```

## login.html:

```

<!DOCTYPE html>

<html>

<head>

    <title>Login</title>

    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">

</head>

<body>

    <div class="container">

        {% with messages = get_flashed_messages(with_categories=true) %}

        {% if messages %}

        {% for category, message in messages %}

        <div class="alert alert-{{ category }}">{{ message }}</div>

        {% endfor %}

        {% endif %}

        {% endwith %}

        <h1>Login</h1>

        <form method="POST" action="{{ url_for('login') }}">

            <div class="form-group">

                <label for="email">Email:</label>

                <input type="email" class="form-control" name="email" required>

            </div>

            <div class="form-group">

                <label for="password">Password:</label>

                <input type="password" class="form-control" name="password" required>

            </div>

```

```
<button type="submit" class="btn btn-primary">Login</button>
```

```
</form>
```

```
<p>Don't have an account? <a href="{{ url_for('signup') }}">Sign up</a></p>
```

```
</div>
```

```
</body>
```

```
</html>
```