

Sprint 1

Date	29 October 2022
Team ID	PNT2022TMID25532
Project Name	Emerging Methods for Early Detection of Forest Fires

app.py

```
# Import Necessary libraries
from datetime import timedelta
from flask import Flask, render_template, request, redirect, session
from flask_sqlalchemy import SQLAlchemy
from flask_session import Session
import os

app = Flask(__name__)

UPLOAD_FOLDER = os.path.join('static', 'uploads')
app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER
app.secret_key = 'admin'

# Database connection
app.config['SQLALCHEMY_DATABASE_URI'] = 'sqlite:///user.db'
app.config['SQLALCHEMY_TRACK_MODIFICATIONS'] = False
db = SQLAlchemy(app)

# user cache

app.permanent_session_lifetime = timedelta(minutes=5)
app.config['SESSION_TYPE'] = 'filesystem'
Session(app)

with app.app_context():
    db.create_all()

# Database Model
class Registration(db.Model):
    _id = db.Column('id', db.Integer, primary_key=True)
    name = db.Column(db.String(25), nullable=False)
    email = db.Column(db.String(120), nullable=False)
    password = db.Column(db.String(16), nullable=False)

    def __repr__(self):
        return f"Name : {self.name} Mail :{self.email}"

# Home Page
@app.route('/', methods=['GET'])
```

```

def home():
    if session.get('name'):
        return render_template('home.html', link_image_video = True)
    else:
        return redirect('/signup')

# signup page
@app.route('/signup', methods=['POST', 'GET'])
def signup():
    if request.method == 'POST':
        name = request.form['uname']
        email = request.form['email']
        pwd = request.form['pwd']
        user = Registration.query.filter_by(name=name).first()
        if user is not None:
            return render_template('index.html', msg='Already
Registered')
        user = Registration(name=name, email=email, password=pwd)
        db.session.add(user)
        db.session.commit()
        return render_template('index.html', msg='Successfully
Registered')
        return render_template('index.html')

# login page
@app.route('/login', methods=['POST'])
def login():
    if request.method == 'POST':
        email = request.form['email']
        pwd = request.form['pwd']
        user = Registration.query.filter_by(email=email).first()
        if user is None:
            return render_template('index.html', msg="Invalid user!")
        if user.password != pwd:
            return render_template('index.html', msg="Invalid
Password!")
        session['name'] = user.name
        return redirect('/')
        return redirect('/signup')

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

```

index.html

```

<!DOCTYPE html>
<html>
<head>
    <title>Signup</title>

```

```

    <link rel="stylesheet" type="text/css" href="{{ url_for('static', filename
= 'css/style.css') }}">
    <link
href="https://fonts.googleapis.com/css2?family=Jost:wght@500&display=swap"
rel="stylesheet">
</head>
<body>
    <div>
        <h1 id = 'header'>Forest Fire Detection <br>and Prevention</h1>
    </div>
    <div class="main">
        <input type="checkbox" id="chk" aria-hidden="true">

        <div class="signup">
            <form action="/signup" method="post">
                <label for="chk" aria-hidden="true">Sign up</label>
                <h3 class="msg">{{ msg }}</h3>
                <input type="text" name="uname" placeholder="User name"
required>
                <input type="email" name="email" placeholder="Email"
required>
                <input type="password" name="pwd" placeholder="Password"
required>
                <button type = "submit">Sign up</button>
            </form>
        </div>

        <div class="login">
            <form action="/login" method="post">
                <label for="chk" aria-hidden="true">Login</label>
                <input type="email" name="email" placeholder="Email"
required>
                <input type="password" name="pwd" placeholder="Password"
required>
                <button type="submit">Login</button>
            </form>
        </div>
    </div>
</body>
</html>

```

style.css

```

body{
    margin: 0;
    padding: 0;
    display: flex;
    justify-content: center;

```

```

    align-items: center;
    min-height: 100vh;
    font-family: 'Jost', sans-serif;
    background: linear-gradient(to bottom, #0f0c29, #302b63, #24243e);
}
.main{
    width: 350px;
    height: 500px;
    background: red;
    overflow: hidden;
    background: url("https://doc-08-2c-
docs.googleusercontent.com/docs/securesc/68c90smiglihng9534mvqmq1946dmis5/fo0p
icsp1nhiucmc0l25s29respgpr4j/1631524275000/03522360960922298374/03522360960922
298374/1Sx0jhdpEpnNIydS4rnN4kHSJtU1EyWka?e=view&authuser=0&nonce=gcrocepgbb17m
&user=03522360960922298374&hash=tfhgbs86ka6divo3llbvp93mg4csvb38") no-repeat
center/ cover;
    border-radius: 10px;
    box-shadow: 5px 20px 50px #000;
}
#chk{
    display: none;
}
.signup{
    position: relative;
    width: 100%;
    height: 100%;
}
label{
    color: #fff;
    font-size: 2.3em;
    justify-content: center;
    display: flex;
    margin: 50px;
    font-weight: bold;
    cursor: pointer;
    transition: .5s ease-in-out;
}
input{
    width: 60%;
    height: 20px;
    background: #e0dede;
    justify-content: center;
    display: flex;
    margin: 20px auto;
    padding: 10px;
    border: none;
    outline: none;
    border-radius: 5px;
}

```

```
}
button{
  width: 60%;
  height: 40px;
  margin: 10px auto;
  justify-content: center;
  display: block;
  color: #fff;
  background: #573b8a;
  font-size: 1em;
  font-weight: bold;
  margin-top: 20px;
  outline: none;
  border: none;
  border-radius: 5px;
  transition: .2s ease-in;
  cursor: pointer;
}
button:hover{
  background: #6d44b8;
}
.login{
  height: 460px;
  background: #eee;
  border-radius: 60% / 10%;
  transform: translateY(-180px);
  transition: .8s ease-in-out;
}
.login label{
  color: #573b8a;
  transform: scale(.6);
}
#chk:checked ~ .login{
  transform: translateY(-500px);
}
#chk:checked ~ .login label{
  transform: scale(1);
}
#chk:checked ~ .signup label{
  transform: scale(.6);
}
#header{
  padding: 0.5em;
  margin : 0.2em;
  font-size : 4em;
  color: #fff;
  text-align:center;
}
```

```
}  
.msg{  
  text-align: center;  
  color: #fff;  
  margin-bottom: 1.5em;  
}
```