# flask\_app/app/\_\_init\_\_.py

from flask import Flask

from flask\_sqlalchemy import SQLAlchemy

from flask\_migrate import Migrate

from flask\_login import LoginManager

app = Flask(\_\_name\_\_)

app.config['SECRET\_KEY'] = 'your\_secret\_key'

app.config['SQLALCHEMY\_DATABASE\_URI'] = 'sqlite:///site.db'

db = SQLAlchemy(app)

migrate = Migrate(app, db)

login\_manager = LoginManager(app)

login\_manager.login\_view = 'login'

from app import routes

# flask\_app/app/models.py

from datetime import datetime

from flask\_login import UserMixin

from app import db, login\_manager

@login\_manager.user\_loader

def load\_user(user\_id):

return User.query.get(int(user\_id))

class User(db.Model, UserMixin):

id = db.Column(db.Integer, primary\_key=True)

username = db.Column(db.String(20), unique=True, nullable=False)

email = db.Column(db.String(120), unique=True, nullable=False)

image\_file = db.Column(db.String(20), nullable=False, default='default.jpg')

password = db.Column(db.String(60), nullable=False)

posts = db.relationship('Post', backref='author', lazy=True)

class Post(db.Model):

id = db.Column(db.Integer, primary\_key=True)

title = db.Column(db.String(100), nullable=False)

date\_posted = db.Column(db.DateTime, nullable=False, default=datetime.utcnow)

content = db.Column(db.Text, nullable=False)

user\_id = db.Column(db.Integer, db.ForeignKey('user.id'), nullable=False)

# flask\_app/app/forms.py

from flask\_wtf import FlaskForm

from wtforms import StringField, PasswordField, SubmitField, BooleanField

from wtforms.validators import DataRequired, Length, Email, EqualTo, ValidationError

from app.models import User

class RegistrationForm(FlaskForm):

username = StringField('Username', validators=[DataRequired(), Length(min=2, max=20)])

email = StringField('Email', validators=[DataRequired(), Email()])

password = PasswordField('Password', validators=[DataRequired()])

confirm\_password = PasswordField('Confirm Password', validators=[DataRequired(), EqualTo('password')])

submit = SubmitField('Sign Up')

def validate\_username(self, username):

user = User.query.filter\_by(username=username.data).first()

if user:

raise ValidationError('That username is taken. Please choose a different one.')

def validate\_email(self, email):

user = User.query.filter\_by(email=email.data).first()

if user:

raise ValidationError('That email is taken. Please choose a different one.')

class LoginForm(FlaskForm):

email = StringField('Email', validators=[DataRequired(), Email()])

password = PasswordField('Password', validators=[DataRequired()])

remember = BooleanField('Remember Me')

submit = SubmitField('Login')

# flask\_app/app/routes.py

from flask import render\_template, url\_for, flash, redirect

from app import app, db

from app.forms import RegistrationForm, LoginForm

from app.models import User, Post

from flask\_login import login\_user, current\_user, logout\_user, login\_required

@app.route("/")

@app.route("/home")

def home():

posts = Post.query.all()

return render\_template('home.html', posts=posts)

@app.route("/register", methods=['GET', 'POST'])

def register():

if current\_user.is\_authenticated:

return redirect(url\_for('home'))

form = RegistrationForm()

if form.validate\_on\_submit():

user = User(username=form.username.data, email=form.email.data, password=form.password.data)

db.session.add(user)

db.session.commit()

flash('Your account has been created! You are now able to log in', 'success')

return redirect(url\_for('login'))

return render\_template('register.html', title='Register', form=form)

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

def login():

if current\_user.is\_authenticated:

return redirect(url\_for('home'))

form = LoginForm()

if form.validate\_on\_submit():

user = User.query.filter\_by(email=form.email.data).first()

if user and user.password == form.password.data:

login\_user(user, remember=form.remember.data)

return redirect(url\_for('home'))

else:

flash('Login Unsuccessful. Please check email and password', 'danger')

return render\_template('login.html', title='Login', form=form)

@app.route("/logout")

def logout():

logout\_user()

return redirect(url\_for('home'))

@app.route("/profile")

@login\_required

def profile():

return render\_template('profile.html', title='Profile')

# flask\_app/config.py

import os

class Config:

SECRET\_KEY = os.environ.get('SECRET\_KEY') or 'a\_hard\_to\_guess\_string'

SQLALCHEMY\_DATABASE\_URI = os.environ.get('DATABASE\_URL') or 'sqlite:///site.db'

# flask\_app/run.py

from app import app

if \_\_name\_\_ == '\_\_main\_\_':

app.run(debug=True)

# flask\_app/templates/base.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>{% block title %}{% endblock %} - Flask App</title>

<link rel="stylesheet" href="{{ url\_for('static', filename='css/style.css') }}">

</head>

<body>

<nav>

<ul>

<li><a href="{{ url\_for('home') }}">Home</a></li>

{% if current\_user.is\_authenticated %}

<li><a href="{{ url\_for('profile') }}">Profile</a></li>

<li><a href="{{ url\_for('logout') }}">Logout</a></li>

{% else %}

<li><a href="{{ url\_for('login') }}">Login</a></li>

<li><a href="{{ url\_for('register') }}">Register</a></li>

{% endif %}

</ul>

</nav>

<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 %}

{% block content %}{% endblock %}

</div>

</body>

</html>

# flask\_app/templates/home.html

{% extends "base.html" %}

{% block title %}Home{% endblock %}

{% block content %}

<h1>Welcome to the Home Page</h1>

{% for post in posts %}

<article>

<h2>{{ post.title }}</h2>

<p>{{ post.content }}</p>

</article>

{% endfor %}

{% endblock %}

# flask\_app/templates/register.html

{% extends "base.html" %}

{% block title %}Register{% endblock %}

{% block content %}

<h2>Register</h2>

<form method="POST" action="">

{{ form.hidden\_tag() }}

<div>

{{ form.username.label }} {{ form.username(size=32) }}

</div>

<div>

{{ form.email.label }} {{ form.email(size=32) }}

</div>

<div>

{{ form.password.label }} {{ form.password(size=32) }}

</div>

<div>

{{ form.confirm\_password.label }} {{ form.confirm\_password(size=32) }}

</div>

<div>

{{ form.submit() }}

</div>

</form>

{% endblock %}

# flask\_app/templates/login.html

{% extends "base.html" %}

{% block title %}Login{% endblock %}

{% block content %}

<h2>Login</h2>

<form method="POST" action="">

{{ form.hidden\_tag() }}

<div>

{{ form.email.label }} {{ form.email(size=32) }}

</div>

<div>

{{ form.password.label }} {{ form.password(size=32) }}

</div>

<div>

{{ form.remember }} {{ form.remember.label }}

</div>

<div>

{{ form.submit() }}

</div>

</form>

{% endblock %}

# flask\_app/templates/profile.html

{% extends "base.html" %}

{% block title %}Profile{% endblock %}

{% block content %}

<h2>Profile</h2>

<p>Welcome, {{ current\_user.username }}!</p>

{% endblock %}