MINI PROJECT REPORT

ON

"DEVPOST BLOGGING WEBAPP"

Submitted in

Partial Fulfillment of requirements for the Award of Degree

of

Bachelor of Technology

In

Computer Science and Engineering

By

(Group Number: G-4)

Apoorv Sharma (1816410068)
Aryan Nigam (1816410071)
Aryan Porwal (1816410072)
Ritik Kanotra (1816410220)

Under the supervision of
Kumar Saurabh
(Asst. Professor)

Pranveer Singh Institute of Technology.
Kanpur – Agra – Delhi National Highway – 2
Bhauti –Kanpur – 209305.
(Dr. A.P.J. Abdul Kalam Technical University)

Index

Objective

To offer informative, helpful and educational blog to people who want to become a technology enthusiast more specifically the developer. Development has been the driving force of the IT industry from the ages and will prove to be the same in the times to come. We, through our website, are interested to contribute in the learning journey of developers by enabling all the developers from different parts of the world to come and contribute to the growth of the developer's community by writing and posting informative, suggestive, precise but insightful blogs so as to benefit others in the developer's community.

The blogs may relate to any type of development be it - web, android, software or some helpful content about any trending technology. The blogs will be judged on the basis of upvotes and downvotes to ensure the quality check. We want to continue to generate more and more traffic for learning. Our ultimate goal, of course, is to become more and more successful through blog and how effectively it touches other people. This will translate into direct time savings by avoiding fruitless searches for required information. It is a platform that is used to manage web content, allowing multiple contributors to create, edit and publish their blogs.

Open-source contribution is emerging as a new trending habit of current developers. To encourage others that are interested but lack guidance, this website will prove to be a turning point heading their journey from a beginner to a pro in development pushing themselves to the edge by learning from others' experience who are in the community.

Introduction

Blog is an abbreviated version of weblog which is a term used to describe websites that maintain an ongoing chronicle of information. A blog features diary-type commentary and links to articles on other websites, usually presented as a list of entries in reverse chronological order. Blogs range from the personal to the political, and can focus on one narrow subject or a whole range of subjects. Many blogs focus on a particular topic, such as web design, home staging, sports, or mobile technology. Some are more eclectic, presenting links to all types of other sites. And others are more like personal journals, presenting the author's daily life and thoughts. Blogs tend to have a few things in common: A main content area with articles listed chronologically, newest on top. Often, the articles are organized into categories. An archive of older article, a way for people to leave comments about the articles, a list of links to other related sites, sometimes. A blogger is a person who owns or runs a blog or a person who maintains the blog.

That is, posting articles or new posts, information, sharing the most up-to-date news, opinions and case studies to name but a few. Such entries are known as blog posts. Content is very important for any blogging website. Retail sites feature a catalog of products. University sites contain information about their campuses, curriculum, and faculty. News sites show the latest news stories. For a personal blog, you might have a bunch of observations, or reviews. Without some sort of updated content, there is little reason to visit a website.

On a blog, the content consists of articles (also sometimes called posts or entries) that the author(s) writes. Yes, some blogs have multiple authors, each writing his/her own articles. Typically, blog authors compose their articles in a web-based interface, built into the blogging system itself. Some blogging systems also support the ability to use standalone weblog client software which allows authors to write articles offline and upload them at a later time. Most blogs have a method to allow visitors to leave comments. There are also nifty ways for authors of other blogs to leave comments without even visiting the blog! Called pingbacks or tracebacks they can inform other bloggers whenever they cite an article from another site in their own articles. All this ensures that online conversations can be maintained painlessly among various sites.

The appearance of blogs has changed over time, and these days blogs include a wide variety of items and widgets. However, most blogs still include some standard features and structures. Here are common features that a typical blog will include: Header with the menu or navigation bar, Main content area with highlighted or latest blog posts. Sidebar with social profiles, favourite content, or call-to-action. Footer with relevant links like a disclaimer, privacy policy, contact page, etc. Blogs need frequent updates. Good examples of this include a food blog sharing meal recipes or a company writing about their industry news.

Blogs also promote reader engagement. Readers have a chance to comment and voice their different concerns and thoughts to the community. Blog owners update their site with new blog posts on a regular basis. Key elements that identify a blog post from a static website page include a publishing date, author reference, categories, and tags within a by line. While not all blog posts have all of those by line elements, static website pages do not have any of these items. From a visitor's perspective, the content on a static site will not change from one visit to the next. However, depending on the blog owner's publishing schedule, the content on a blog will offer something new each day, week, or month.

This project is a blogging web application made on Flask. Using this application people can explore blogs based on variety of categories for free. This web application also provides a platform for bloggers to post their blogs. The content is checked by a system of upvotes and downvotes to ensure quality check for the blogs which are posted.

Feasibility Study

This case study will try and assess the feasibility for a developer blogging website. We will analyze the costs and benefits associated with the implementation of a blogging website for a membership society.

Technical Feasibility - The technology we have used is 'Flask'. Flask framework is lightweight, trusted and used by many big Web-based projects across the globe. It is easy to implement and flask app can be hosted as a website. SQLite database is open-source, secure and a trusted platform to store user data and is available for free.

Economic Feasibility - All the tools and technologies required to make the project are easily available, mostly open-source and accessible over the internet so it is economically feasible. Also, nowadays, there are many trusted platforms which offers free services to publish a Python-based web application

Legal Feasibility - All technologies used in the project are open source and available on the internet so it is legally feasible.

Technology Used

FRONT END

HTML (version 5)

CSS (version 3)

JAVASCRIPT

JQUERY (version 3.5.1)

BOOTSTRAP (version 3 & 5)

BACK END

FLASK 2.0 (Python based Web Framework)

JINJA 2 (Templating engine for Python)

SQLITE Database

Summernote - WYSIWYG Editor

PYTHON (version 3.8)

HEROKU

Hardware/Software Required

- A Web Browser (Google Chrome / Mozilla Firefox / Internet Explorer)
- A Text Editor (Sublime Text / Notepad++ / VS Code)
- Flask Server
- Python 3
- Python editor and compiler
- Flask and related libraries

Coding

Imports (app.py) ->

```
from flask import Flask, render_template, request, redirect, session
from flask_session import Session
import sqlite3
from cs50 import SQL
from werkzeug.security import check_password_hash, generate_password_hash
from helpers import login_required
import os
from werkzeug.utils import secure_filename
from base64 import b64decode
import uuid
import magic
import shutil
from datetime import datetime
```

Register (app.py) ->

```
@app.route('/register', methods=['POST'])
def register():
    session.clear()

    if request.method == "POST":

        name = request.form.get('name')
        email = request.form.get('email')
        password_hash = generate_password_hash(request.form.get('password'))

        rows = mydb.execute("SELECT * FROM users WHERE email=:email;",
        email=email)
```

```
        if len(rows) != 0:
            return render_template("apology.html", msg="Email address already
            registered!", back="/register")

        mydb.execute("INSERT INTO users (name, email, password_hash)
        VALUES(?, ?, ?);", name, email, password_hash)

        return redirect("/")
```

Login (app.py) ->

```
@app.route('/login', methods=['GET', 'POST'])
def login():
    if request.method == 'GET':
        return render_template("login.html")
    elif request.method == 'POST':
        email = request.form.get('email')
        password = request.form.get('password')

        rows = mydb.execute("SELECT * FROM users WHERE email=:email", email =
        email)

        if len(rows) == 0:
            return render_template("apology.html", msg="You are not
            registered!", back="/login")

        print(check_password_hash(rows[0]["password_hash"], password))

        if check_password_hash(rows[0]["password_hash"], password) == False:
            return render_template("apology.html", msg="Incorrect password!
            Try again.", back="/login")

        session["user_id"] = rows[0]["id"]

        return redirect("/")
```

Like/Dislike (app.py) ->

```
@app.route('/update_like_dislike/<blog_title>/<liked>/<disliked>')
def update_like_dislike(blog_title, liked, disliked):
    print("update_like_dislike called")
    print("like: {}, dislike: {}".format(liked, disliked))
    if liked == '1':
        print('yes')
        mydb.execute("UPDATE blogs SET likes=likes+1 WHERE title=:title",
        title=blog_title)

    if disliked == '1':
        mydb.execute("UPDATE blogs SET dislikes=dislikes+1 WHERE
        title=:title", title=blog_title)

    return "1"
```

Logout (app.py) ->

```
@app.route("/logout")
def logout():
    session.clear()
    return redirect("/")
```

html/css for each blog card (index.html) (using Jinja) ->

```
{% for row in rows %}

  <div class="card" style="width: 22rem; margin-bottom: 30px;">
     <img class="card-img-top" width="50px" height="200px"
      src="/static/assets/images/user_cache/{{ row['coverimage'] }}"
```

```
      alt="Card image cap">
         <div class="card-body">
            <h5 class="card-title" style="height: 100px; overflow: hidden;
             text-overflow: ellipsis;">{{ row['title'] }}</h5>
               <a href="/blogs/{{ row['title'] }}" class="btn btn-
                primary">Read more</a>
         </div>
  </div>

{% endfor %}
```

Like/Dislike feature (JQuery/JS) ->

```
var blog_title = $('#blog_title').text()

var likes = parseInt($('#likes').text())
var dislikes = parseInt($('#dislikes').text())

var liked = 0;
var disliked = 0;

setIcon();

$('#like_img').click(function(e) {
  liked = 1;
  disliked = 0;
  likes = likes + 1;

  e.preventDefault()
    $.getJSON('/update_like_dislike/' + blog_title + '/1/0',
        function(liked, disliked) {

    });

  setIcon();
```

```
});


$('#dislike_img').click(function(e) {
  liked = 0;
  disliked = 1;
  dislikes = dislikes + 1;


  e.preventDefault()
    $.getJSON('/update_like_dislike/' + blog_title + '/0/1',
        function(liked, disliked) {

    });
  setIcon();
});

function setIcon() {

  updateLikeDislike()


  if (liked == 1) {
    $('#like_img').attr('src', "{{ url_for('static',
    filename='assets/images/liked_ico.png') }}");
    $('#like_img').off('click');
    $('#dislike_img').attr('src', "{{ url_for('static',
    filename='assets/images/dislike_ico.png') }}");
    $('#dislike_img').click(function(e) {
      dislikes = dislikes + 1;


      liked = 0;
      disliked = 1;
      e.preventDefault()
        $.getJSON('/update_like_dislike/' + blog_title + '/0/1',
            function(liked, disliked) {


        });
      setIcon();
    });
  }


  if (disliked == 1) {
```

```
    $('#dislike_img').attr('src', "{{ url_for('static',
    filename='assets/images/disliked_ico.png') }}");
    $('#dislike_img').off('click');
    $('#like_img').attr('src', "{{ url_for('static',
    filename='assets/images/like_ico.png') }}");
    $('#like_img').click(function(e) {

      likes = likes + 1;

      liked = 1;
      disliked = 0;
      e.preventDefault()
        $.getJSON('/update_like_dislike/' + blog_title + '/1/0',
            function(liked, disliked) {
          //do nothing
        });
      setIcon();
    });
  }
}

function updateLikeDislike() {
  $('#likes').text(likes.toString());
  $('#dislikes').text(dislikes.toString());
}
```

cards.css ->

```
.card {
  box-shadow: 0 4px 16px 0 rgba(0, 0, 0, 0.2);
  transition: 0.3s;
  padding: 20px;
  text-align: center;
  margin-left: 15px;
  margin-right: 15px;
  background-color: white;
  border-radius: 3px;
  border-width: 1px;
  margin-bottom: 60px;
}

.card-container .card img {
  height: 200px;
  width: 100px;
}

tr {
  padding: 30px;
}

.container {
  overflow-x: auto;
}

.card-container {
  display: grid;
  padding: 2rem 18rem;
  grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));
  grid-gap: 1rem;
}
```

new_blog.html ->

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <title>Devpost-New Blog</title>
    <link rel="shortcut icon" type="image/png" href="{{ url_for('static',
    filename='assets/images/devpost_logo_image.png') }}"/>

    <!-- include libraries(jQuery, bootstrap) -->
    <link
href="https://stackpath.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.cs
     s" rel="stylesheet">
    <script src="https://code.jquery.com/jquery-3.5.1.min.js"></script>
    <script
src="https://stackpath.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js">
</script>

    <!-- include summernote css/js -->
    <link
href="https://cdn.jsdelivr.net/npm/summernote@0.8.18/dist/summernote.min.css"
rel="stylesheet">
    <script
src="https://cdn.jsdelivr.net/npm/summernote@0.8.18/dist/summernote.min.js"><
/script>

    <link rel="stylesheet" href="{{ url_for('static',
filename='styles/cards.css') }}">
    <link rel="stylesheet" href="{{ url_for('static',
filename='styles/new_blog.css') }}">

    <link rel="preconnect" href="https://fonts.gstatic.com">
    <link
href="https://fonts.googleapis.com/css2?family=Righteous&display=swap"
rel="stylesheet">

    <link rel="stylesheet" href="{{ url_for('static',
```

```
filename='styles/footer.css') }}">
  </head>
  <body>

    <div class="card" style="width: 100%; height: 400px; position: relative;
top: 0px; margin-left: 0px; padding: 0px;">

      <img height="400px" width="100%" src="{{ url_for('static',
filename='assets/images/bg_new_blog_2.png') }}">

    </div>

    <h1 style="position: absolute; top: 60px; margin-left: 10%; text-align:
left; font-family: 'Righteous', cursive; margin-bottom: 20px; font-size:
80px;">Create a new blog □</h1>

    <div style="position: absolute; top: 230px; width: 100%;">

      <div class="card editor-block">

        <form action="/new_blog" method="post" enctype="multipart/form-data">
          <input type="text" class="form-control"
            name="title"
            placeholder="Title* (maximum 50 characters)"
            aria-describedby="basic-addon1"
            onkeypress="return event.keyCode != 13;"
            style="font-size: 20px; height: 40px; margin-bottom: 30px;
margin-top: 20px;" required>

          <!-- <form action="/new_blog" method="POST"> -->
            <div class="form-group">
              <label for="exampleFormControlFile1">Upload cover
image.</label>
              <input type="file" accept="image/*" class="form-control-file"
id="input_cover_image" name="input_cover_image" required>
            </div>
          <!-- </form> -->
```

```
          <img src="{{ url_for('static',
filename='assets/images/preview_image.png') }}" width="50%" style="margin-
bottom: 20px;" id="preview_cover_image" name="preview_cover_image">

          <textarea id="summernote" name="editordata"></textarea>
          <!-- <input type="submit" value="Post"> -->
          <button type="submit" class="btn btn-primary">Submit</button>
        </form>

      </div>

      <footer>
        <div class="footer" id="footer">
          <div class="container">
            <div class="row">
              <div class="col-lg-6 col-md-6 col-sm-6 col-xs-6">

                <h4><img style="height: 60px" src="{{ url_for('static',
            filename='assets/images/devpost_logo_image.png') }}" alt="logo">
            &nbspdevpost.herokuapp.com </h4>
                <p>
                    We ,through our website Devpost, are interested to
   contribute in the learning journey of developers by enabling all the
   developers from different parts of the world to come and contribute to the
   growth of the developers community by writing and posting blogs.
                </p>
              </div>
              <div class="col-lg-3 col-sm-2 col-xs-3">
                <h3> Contact </h3>
                <ul>
                  <li><a class="email" href="#"> devpost.app@gmail.com
</a></li>
                  <br />
                  <li>
                    <p> Computer Science and Engineering Department PSIT </p>
                  </li>
                  <li>
                    <p> Bhauti, Kanpur-209305 </p>
```

```
                  </li>
                </ul>
              </div>
              <div class="col-lg-3 col-sm-2 col-xs-3">
                <ul>
                  <li>
                    <h6> <a href="/about" style="margin-top: 5em"> ABOUT
US</a>
                    </h6>
                  </li>
                  <li>
                    <h6><a href="mailto:devpost.app@gmail.com"> CONTACT US
</a>
                      </h6>
                  </li>
                  <li>
                    <h6><a href="/login"> REGISTER </a>
                      </h6>
                  </li>
                  <li>
                    <h6><a href="/login"> LOGIN </a>
                      </h6>
                  </li>
                </ul>
              </div>
            </div>
          </div>

          <div class="footer-bottom">
            <div class="container">
              <p class="pull-left copyright"> Copyright © Devpost 2021. All
right reserved. </p>

            </div>
          </div>
          </div>
      </footer>
```

```
    </div>

    <script src="{{ url_for('static', filename='js/new_blog.js')
}}"></script>

  </body>
</html>
```

login.html ->

```
<!DOCTYPE html>
<html lang="en">
<head>
      <meta charset="utf-8">
      <meta name="viewport" content="width=device-width,initial-
scale=1,shrink-to-fit=no">
      <meta http-equiv="x-ua-compatible" content="ie=edge">
    <link rel="stylesheet" href="{{ url_for('static',
filename='styles/style.css') }}">
    <title>Devpost-Login</title>
    <link rel="shortcut icon" type="image/png" href="{{ url_for('static',
filename='assets/images/devpost_logo_image.png') }}"/>
</head>

<body class="bg">

  <img class="ill_img" src="{{ url_for('static',
filename='assets/images/login_illustration.svg') }}">

  <img class="logo_img" src="{{ url_for('static',
filename='assets/images/devpost_logo_image.png') }}">

  <img class="logo_img" src="{{ url_for('static',
filename='assets/images/devpost_logo_written_trans.png') }}">
```

```
<div class="form">
  <div class="tab-header">
    <div class="active">Sign Up</div>
    <div>Sign In</div>
  </div>
  <div class="tab-content">
    <div class="tab-body active">
      <form action="/register" method="post">
        <div class="form-element">
          <input name="name" type="text" placeholder="Name" required>
        </div>
        <div class="form-element">
          <input name="email" type="email" placeholder="Email" required>
        </div>
        <div class="form-element">
          <input name="password" type="password" placeholder="Password"
required>
        </div>
        <div class="form-element">
          <button type="submit">Sign Up</button>
        </div>
      </form>
    </div>

    <div class="tab-body">
      <form action="/login" method="post">
        <div class="form-element">
          <input name="email" type="email" placeholder="Email" required>
        </div>
        <div class="form-element">
          <input name="password" type="password" placeholder="Password"
required>
        </div>
        <div class="form-element">
          <button type="submit">Sign In</button>
        </div>
      </form>
```

```
    </div>
  </div>
</div>
<script src="{{ url_for('static', filename='js/main.js') }}"></script>
</body>
</html>
```

Conclusion

The project “DEVPOST BLOGGING WEBAPP” succeeds in developing a website for blogging which enables the growth of learning and supports the community imparting knowledge and the much needed breakthroughs to complex problems.

While developing the project a conscious effort has been made to create and develop the package, making the use of available tools, techniques and resources – that would generate a proper system for use.

It is ensured that the project is as user – friendly as such one may hope that the system will be acceptable to any user and will adequately meet his/her needs.

Future Scope

This project in future will work on the following functionalities:

- Apart from the written textual blogs the website will also provide the functionality of video explanations of the textual blogs so as to give a crisp and clear understanding of the blogs.

- Also, an additional functionality which will classify the blog content as positive and negative before posting it into our website. This will ensure another layer of quality check and hence enhance the user experience.

- This project also looks forward to translate the blog content into different languages of various origins.

- We are also planning to add a feature of a search engine to increase the productivity of the content and also a chatbot assistance.

References

- https://flask.palletsprojects.com/en/1.1.x/ - > Flask Documentation
- https://jinja.palletsprojects.com/en/2.11.x/ -> Jinja (Templating language for Python) Documentation
- https://docs.python.org/3/ -> Python (v3) Documentation
- https://stackoverflow.com/questions/25286176/how-to-use-python-magic-5-19-1 -> For usage of Python "Magic" module
- https://api.jquery.com/ -> Official jQuery Documentation
- https://www.w3schools.com/bootstrap/bootstrap_navbar.asp -> Bootstrap Navbar
- https://getbootstrap.com/docs/5.0/components/navbar/ -> Bootstrap Navbar (Official Documentation)
- https://www.w3schools.com/bootstrap/bootstrap_forms.asp -> Bootstrap Forms
- https://www.w3schools.com/bootstrap/bootstrap_buttons.asp -> Bootstrap Buttons
- https://cs50.harvard.edu/web/2018/notes/2/ -> Python and Flask -> Documentation by Harvard University CS50 Course.
- https://devcenter.heroku.com/articles/getting-started-with-python -> Official Heroku Documentation (For deploying a python application for free)
- https://stackabuse.com/deploying-a-flask-application-to-heroku/
- https://www.geeksforgeeks.org/deploy-python-flask-app-on-heroku/
- https://medium.com/@gitaumoses4/deploying-a-flask-application-on-heroku-e509e5c76524
- https://icons8.com/ -> Icons8 (for various icons)
- https://www.flaticon.com/ Flaticon (for various icons)
- https://fonts.google.com/ - > Google Fonts
- https://summernote.org/getting-started/ -> Summernote Official
- https://summernote.org/deep-dive/ -> Summernote Advanced