****

**PROJECT ON**

**CODE REFACTORING AND BUG REPORT**

**By Sanhita Thangallapalli**

**ABOUT ME:**

I’m Sanhita Thangallapalli, and I have studied my master’s from JNTU College of Engineering Hyderabad where I specialized in Electrical Power Engineering. However, I have a passion for data analytics that ignited a new path for me and started gaining knowledge on skills used for data analytics such as python and its libraries like Numpy and Pandas to analyze data. I’ve also have good handle on MYSQL for managing databases.

One of the things I love most is digging into data and uncovering hidden insights. Whether it’s spotting trends or making sense of complex datasets, I thrive on the challenge of exploratory data analysis. And visualizing data with tools like Matplotlib and Seaborn for impressive charts and graphs.

I’m skilled at using Power BI to create interactive dashboards that tell compelling stories with data. And if you need a web application to showcase your findings, I’ve got you covered with my knowledge of stream lit and Request libraries.

In essence, I’m curious and passionate individual with a knack for turning data into insights. I’m always eager to learn and grow, and I’m constantly seeking out new opportunities to expand my horizons in the world of data analytics.

Github : https://github.com/Sanhita0220?tab=repositories

LinkedIn: https://www.linkedin.com/in/sanhita-thangallapalli/

Overview:

The Note Taking Application is a Web-based tool developed using Python, Flask, and HTML. It purposes is to allow users to add notes and view them in a list format on the same page. This documentation outlines the functionality of the application, how it’s built, identified bugs, and the approach used to fix them.

How it Works:

* Adding Notes:

You can types something in a box and press a button to save it. Your note will then show up in a list below

* Empty Note Warning:

If you try to save a note without writing anything, the app will tell you it can’t save empty notes.

* Viewing Notes:

All the notes you have saved are shown in a list under the text box.

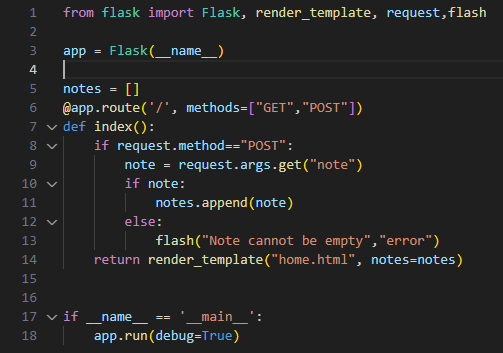
Code Explanation (app.py):

* Imports: We bring in the necessary parts of Flask and other tools.
* Flask: We create the app and set a secret key for security.
* Keeping Track of Notes: We start with an empty list to hold all the notes.
* Handling Requests: We tell the app what to do when someone visits the site. For getting the page, we show the page with the notes and for adding a note, type checking and adding the note to the list if not empty.
* Starting the App: We start the Flask app so people can use it.
* from flask import Flask, render\_template, request,flash
* app = Flask(\_\_name\_\_)
* notes = []
* @app.route('/', methods=["POST"])
* def index():
* note = request.args.get("note")
* if note:
* notes.append(note)
* else:
* flash("Note cannot be empty","error")
* return render\_template("home.html", notes=notes)
* if \_\_name\_\_ == '\_\_main\_\_':
* app.run(debug=True)

Template explanation (home.html)

* It’s a simple webpage with a box for typing and button to save notes. Shows all the saved notes.

Fixing problems:

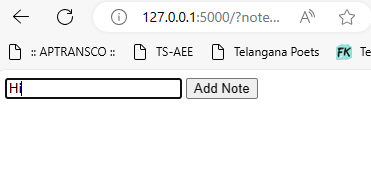


List of Bugs Identified and fixed:

**First Bug Identified:** The code used **‘request.args.get’** to retrieve form data, which is incorrect for POST requests. **Fix:** Changed to **‘request.form.get’** to properly retrieve data submitted through POST method from the form.

**Second Bug Identified:** There was no check for empty notes before appending them to the list. **Fix:** Added a check to ensure that the note is not empty before appending it to notes list.

**Third Bug Identified:** There was no feedback provided to users when they submitted an empty note. **Fix:** Added a flash message to notify users when they try to submit an empty note with a secret key.

Conclusion: 

After identifying and fixing the issues in the application we have successfully improved its functionality and usability. By addressing the methods of data retrieval, ensuring that empty note and implementing flash message for empty message. With these fixes, the app runs smoothly, allowing for note taking and viewing.