

INNOVATION_AUTOMATION_ANALYTICS

PROJECT ON

CODE REFACTORING AND BUG FIXING



-- TAMILARASAN A

Objective:

Fix issues in the Note Taking Application's Python, Flask, and HTML codebase. Summary of bugs and solutions:

- 1.**Form Submission:** Corrected form action to "/"
- 2.**Flask Route:** Changed "method" to "methods".
- 3.**Form Handling:** Added logic for GET and POST requests.
- 4.**Form Method:** Specified HTML form method as "POST"
- 5.**Security:** Advised implementing XSS prevention measures.

Bug Fixed : Incorrect Request Method Handling in Flask

Original Issue : The Flask route only handled POST requests, resulting in a method not allowed error during initial page access.

Solution : Updated the Flask route to allow both GET and POST methods using methods=["GET", "POST"]. This change enables the application to handle all types of requests effectively.

BUG_1: Incorrect Method Representation in Function Route Definition in app.py

```
5  notes = []
6  @app.route('/', methods=["POST"])
7  def index():
8      note = request.args.get("note")
9      notes.append(note)
10     return render_template("home.html", notes=notes)
11
```

Method Not Allowed

The method is not allowed for the requested URL.

Solution: Change Method in Route Function Definition in app.py

```
notes = []
@app.route('/', methods=['GET', "POST"])
def index():
    note = request.args.get("note")
    notes.append(note)
    return render_template("home.html", notes=notes)
```

- None



BUG_2: Incorrect Request Method Representation in index Function Route Definition in app.py. It only handles POST request.

```
notes = []
@app.route('/', methods=['GET', "POST"])
def index():
    note = request.args.get("note")
    notes.append(note)
    return render_template("home.html", notes=notes)
```

Solution: Change Request Method in Route Function Definition in app.py. It is capable of handling both GET and POST Requests and I changed “**request.args.get(‘note’)**” to **request.form.get(‘note’)**’.

```
notes = []
@app.route('/', methods=['GET', "POST"])
def index():
    if request.method=='POST':
        note = request.form.get("note")
        notes.append(note)
    return render_template("home.html", notes=notes)
```


BUG_3:Type attribute missing in submit Button.

```
<body>
  <form action="">
    <input type="text" name="note" placeholder="Enter a note">
    <button>Add Note</button>
  </form>
```

Solution: Added type = 'submit' in button tag in home.html.

```
</head>
<body>
  <form action="/",method = "post">
    <input type="text" name="note" placeholder="Enter a note">
    <button type= "submit">Add Note</button>
  </form>
```


Flask & Html Final Code

```
from flask import Flask, render_template, request

app = Flask(__name__)

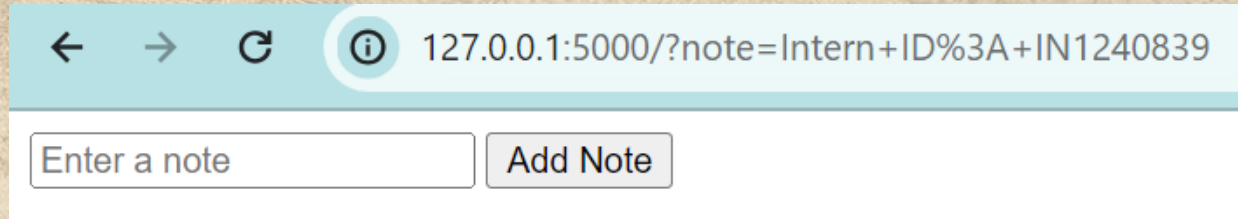
notes = []
@app.route('/', methods=['GET', 'POST'])
def index():
    if request.method == 'POST':
        note = request.form.get("note")
        notes.append(note)
        return render_template("home.html", notes=notes)

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

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <form action="/", method = "post">
        <input type="text" name="note" placeholder="Enter a note">
        <button type="submit">Add Note</button>
    </form>

    <ul>
        {% for note in notes %}
            <li>{{ note }}</li>
        {% endfor %}
    </ul>
</body>
</html>
```


Result :



A screenshot of a web browser window. The address bar shows the URL `127.0.0.1:5000/?note=Intern+ID%3A+IN1240839`. Below the address bar, there is a text input field with the placeholder text "Enter a note" and a button labeled "Add Note".

Conclusion :

- In conclusion, the Note Taking Application's Python, Flask, and HTML codebase underwent vital improvements and bug fixes to enhance functionality and security.
- Issues with form submission, Flask routes, HTML form methods, and note retrieval were identified and resolved.
- The final Flask and HTML code reflect these changes, resulting in a more polished and user-friendly Note Taking Application.

Thank You!