



INNOVATION. AUTOMATION. ANALYTICS

PROJECT ON CODE REFACTORING AND BUG FIXING

About me

- Vijay sada
- B-tech in Electronics and Communications Engineering.
- Fresher
- LinkedIn - <https://www.linkedin.com/in/vijay-sada/>
- GitHub - <https://github.com/vijaysada>

Aim :

- **To Develop A Note Taking Application Using Python, Flask, And HTML.**

Objective :

- **To Fix The Existing Codebase And Make The Application Work As Intended.**

Agenda

- **Identifying The Bugs**
- **Debug And Document Bugs**
- **Refactor The Code**
- **Summarize**

Initial Code Snippets

Flask Code :

```
from flask import Flask, render_template, request

app = Flask(__name__)

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

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

HTML Code :

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

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

Identifying The Bugs

Observed Error 1 :

We got the 'Method Not Allowed' message because of the methods declared is only "POST"

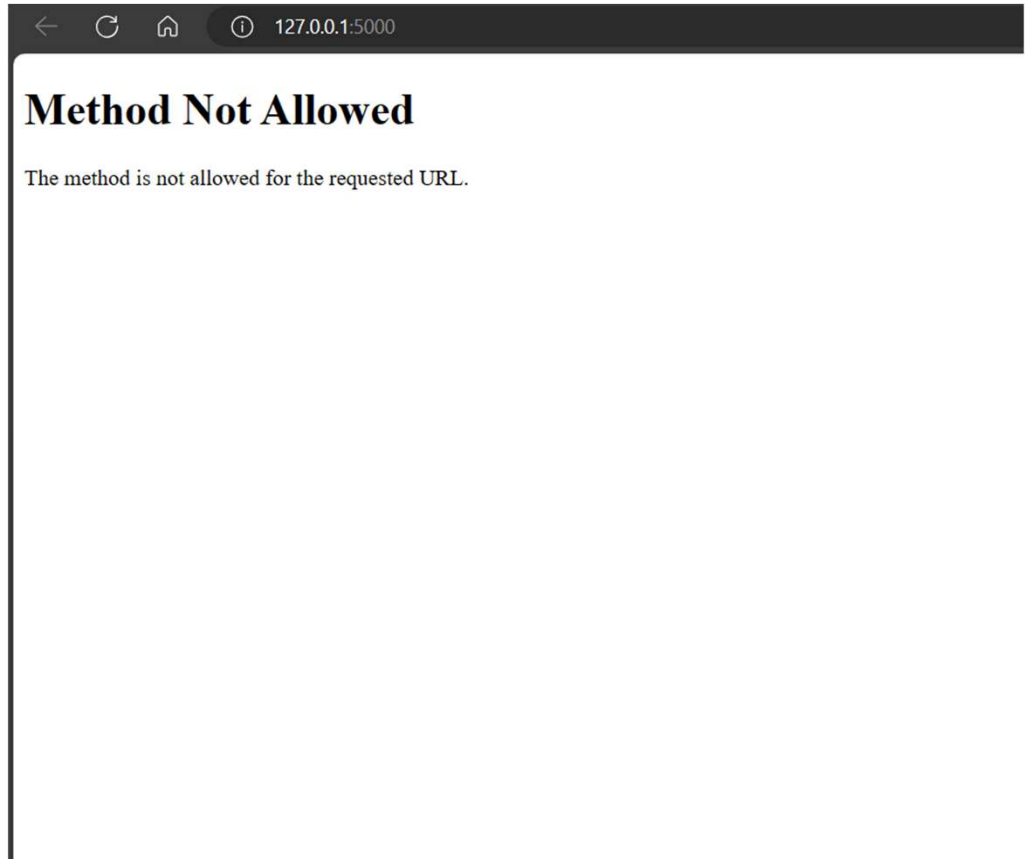
```
@app.route('/', methods=["POST"])
```

Resolution :

We can fix this by adding the "GET" statement inside the methods attribute.

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

Error Snippet :



Identifying The Bugs

Observed Error 2 :

There is an issue with the code which make different users access the same content and make changes to it. But that's not an valid operation because of which the users data isn't protected. This is something that needs to be fixed.

Resolution : `app.secret_key = secrets.token_bytes(32)`

- Utilizing sessions in Flask empowers you to maintain state information for individual users as they navigate through your web application.
- Sessions provide a seamless way to differentiate between users, ensuring that each user's interactions with the application remain independent and secure.

Identifying The Bugs

Observed Error 3 :

As there are multiple methods declared in the same route. When the “GET” method is requested the input gets appended to notes which is an unintended behaviour as we don’t want that to happen.

```
def index():  
    note = request.args.get("note")  
    notes.append(note)  
    return render_template("home.html", notes=notes)
```

Resolution :

So we can improve the code by adding the if clause with the condition of checking if method = “POST” to perform the required set of action. And as we want to get the data from the “**form**” tag in HTML we need to change the request.

```
if request.method == "POST":  
    note = request.form.get("note")  
    if note:  
        notes.append(note)  
        session['notes'] = notes  
    return redirect('/')
```

Identifying The Bugs

Observed Error 4 :

There are some missing parts in the HTML file which defines the functionality of the form tag in the file.

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

Resolution :

We can fix this by adding the "/" statement inside the **action** attribute of the form and "POST" in the **method** attribute and "submit" in **button type** tag.

```
<form action="/" method="POST">
  <input type="text" name="note" placeholder="Enter a note">
  <button type="submit">Add Your Note</button>
</form>
```


Refactored Code Snippet

Flask Code :

```
note_taking_app > app_final.py > index
1  from flask import Flask, render_template, request, session, redirect
2  import secrets
3
4  app = Flask(__name__)
5
6  app.secret_key = secrets.token_bytes(32)
7
8  @app.route('/', methods=["GET", "POST"])
9  def index():
10     global notes
11     notes = session.get('notes', [])
12
13     if request.method == "POST":
14         note = request.form.get("note")
15         if note:
16             notes.append(note)
17             session['notes'] = notes
18             return redirect('/')
19
20     return render_template("sample.html", notes=notes)
21
22 if __name__ == '__main__':
23     app.run(debug=True)
24
```

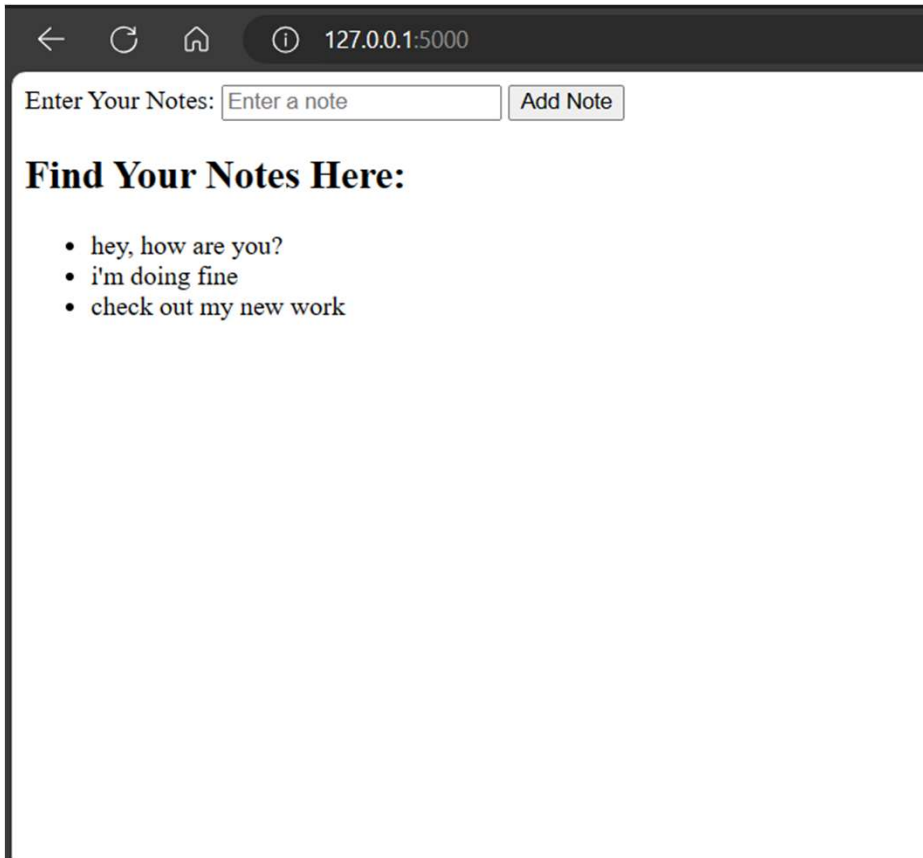
HTML Code :

```
<!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>Note Taking App</title>
</head>
<body>
    <form action="/" method="POST">
        Enter Your Notes: <input type="text" name="note" placeholder="Enter a note">
        <button type="submit">Add Note</button>
    </form>

    <h2>Find Your Notes Here:</h2>

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

Final Output :



← ↻ 🏠 ⓘ 127.0.0.1:5000

Enter Your Notes: Add Note

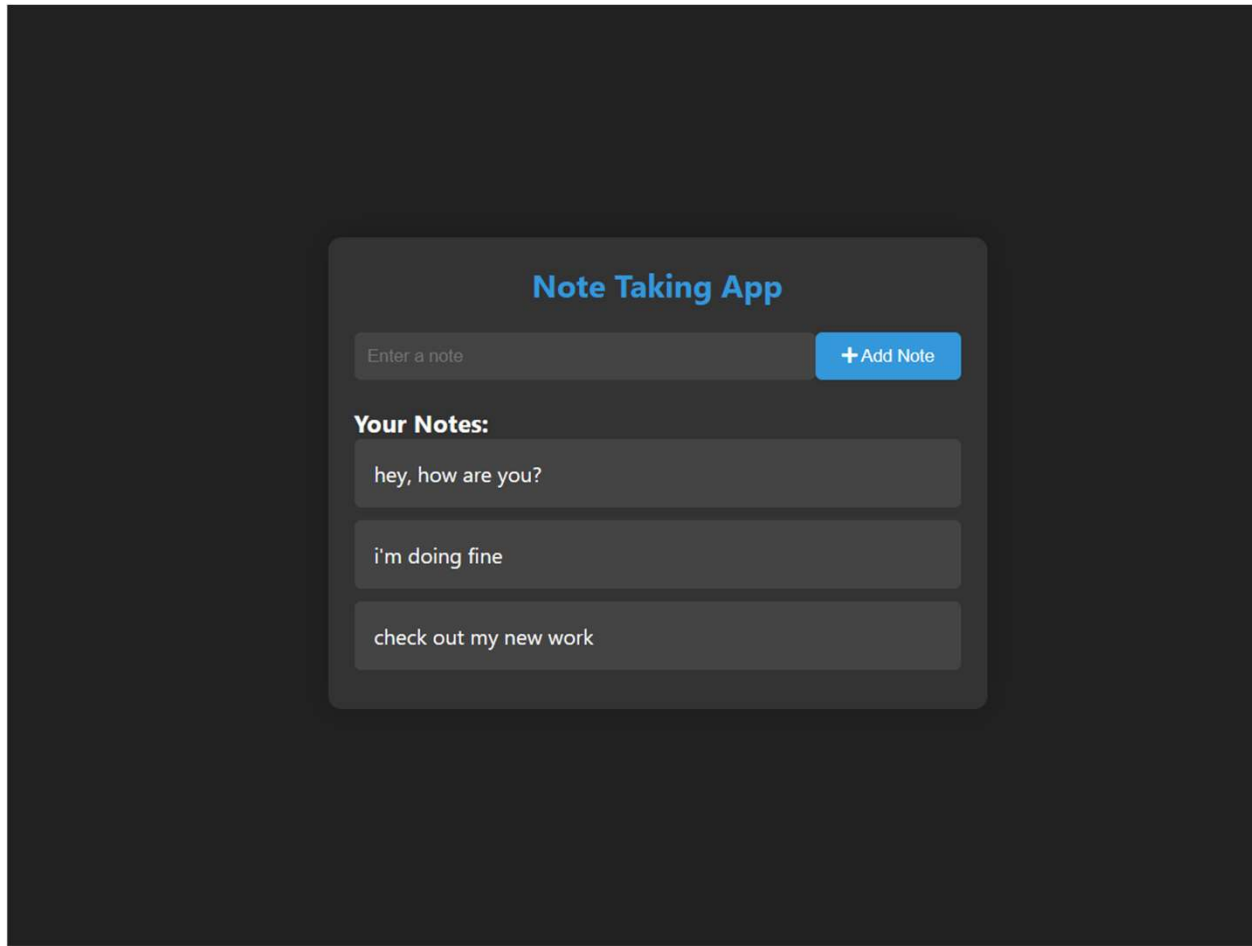
Find Your Notes Here:

- hey, how are you?
- i'm doing fine
- check out my new work

We have fixed the application.

Now, users can add a note, and all the notes will be displayed as an unordered list below the text field on the same page.

Extra Sample Output:



- Beautification after using CSS for the file.

Conclusion :

The note-taking app had some problems(bugs) that stopped it from working the way we expected it to. So we have fixed those issues by refactoring and fixing the code and making it better.

Here are the key things to conclude:

- We found and fixed three main problems: getting the data from wrong source, missing details in the attributes like action, methods, etc. and adding notes when we didn't need to.
- We have fixed the issue of generic content and behavior by leveraging sessions to maintain individual user states.
- We have fixed the issue of duplicates upon refresh by using if statement and redirect.
- We have made changes to both the Flask code and the HTML code to resolve these issues.
- Now, the app works well. It can take notes from users and show them on the page correctly.

THANK
YOU

