Project description: Developing a Note Taking Application

A team of enthusiastic data scientists started developing a Note Taking application using Python, Flask, and HTML. However, their limited experience in backend development has resulted in challenges, preventing the application from functioning properly. They seek assistance in fixing the broken code and ensuring the application works seamlessly.

Objective:

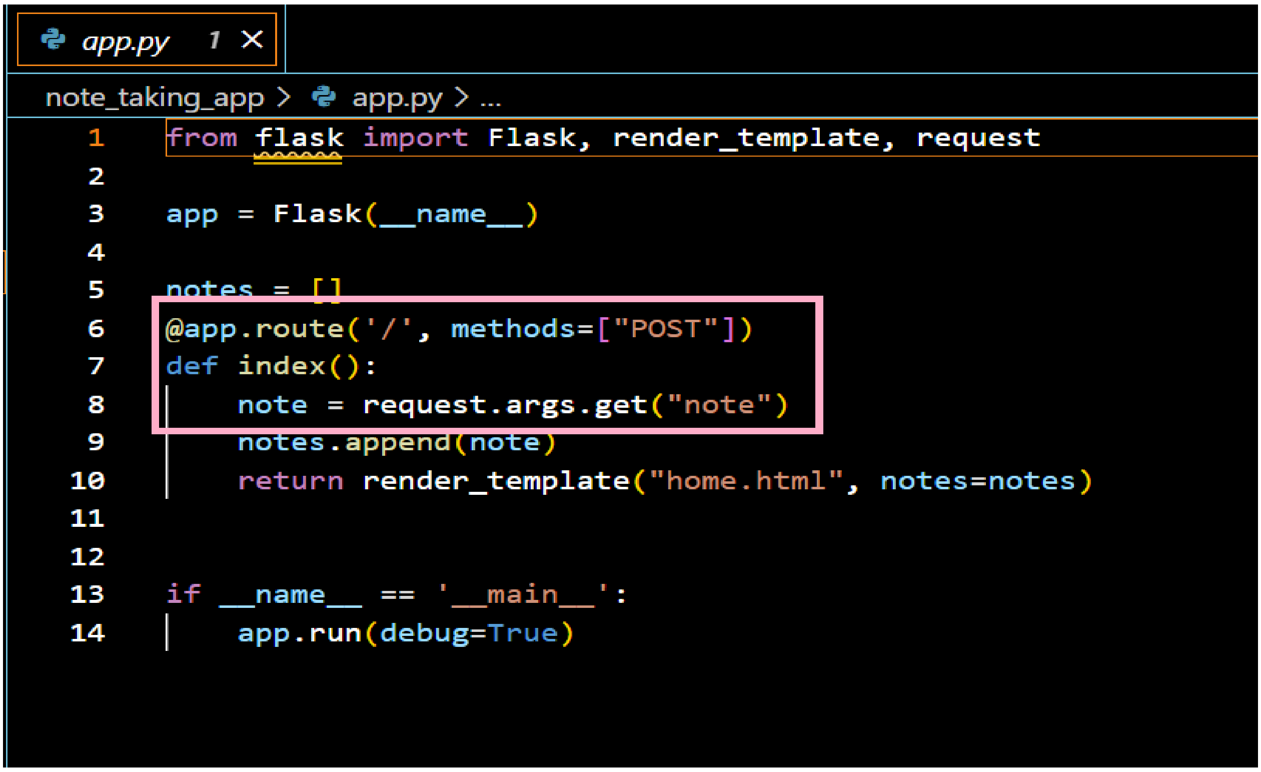
The objective is to refactor the existing code of the Note Taking Application and ensure its proper functioning.

Steps to be followed:

- Identify the existing issues and bugs with in the code.
- Analyse the code to understand the root causes.
- Update the code to address the identified bugs and improve the code quality.
- Test the application to verify the effectiveness of the changes made.
- Ensure that the Note Taking Application works seamlessly, allowed users to add notes without encountering any errors.

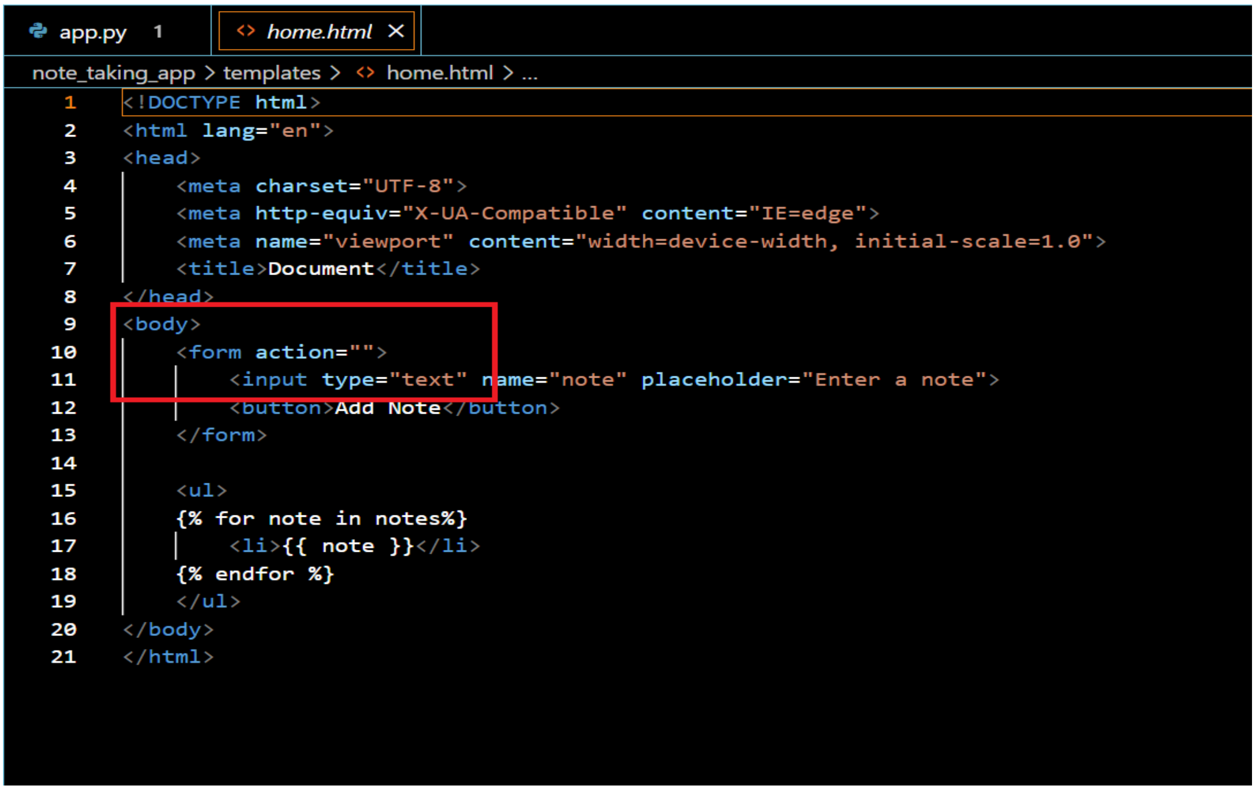
Initial code before fixing the bug:

The following images shows the initial code before identifying the issues and bugs.



```python
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)
```

Figure 1: Python Flask code before bug fixing



```html
<!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>
```

Figure 2: HTML code before bug fixing

Here the highlighted boxes are showing the bug need to be fixed.

Identifying the problem with code:

When submitting notes via “add a note” form, the flask application is encountering an issue with handling POST request. Specially, while the GET request work as expected, POST requests are not being handled correctly, resulting in a method error.

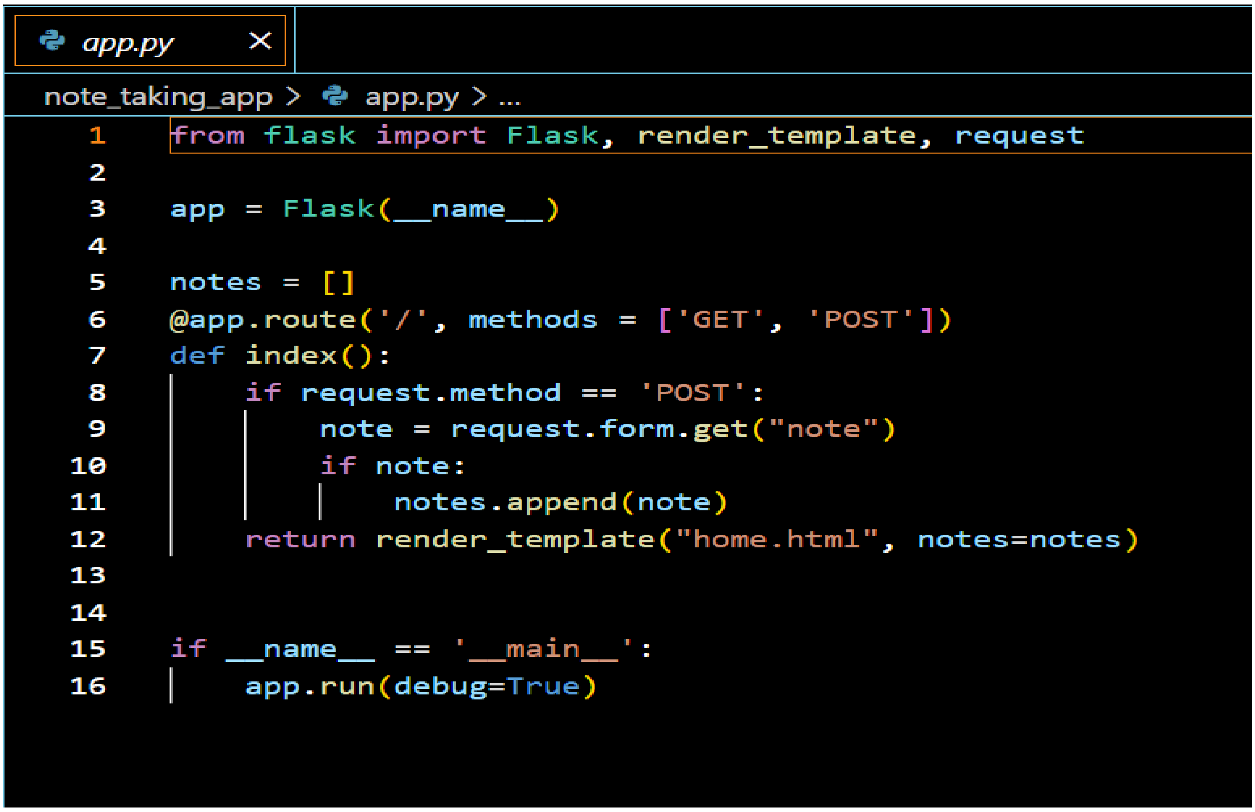
In Figure 1, the highlighted section indicates that the POST request method is being used, but there seems to be an issue with retrieving the note correctly. The syntax for getting the note appears to be incorrect.

Similarly, in Figure 2, the body section of the HTML code shows that the form action is not linked to a proper URL for the intended action, and the request method is also not specified. Both of these aspects need to be corrected to ensure the application functions properly.

Resolving the issue:

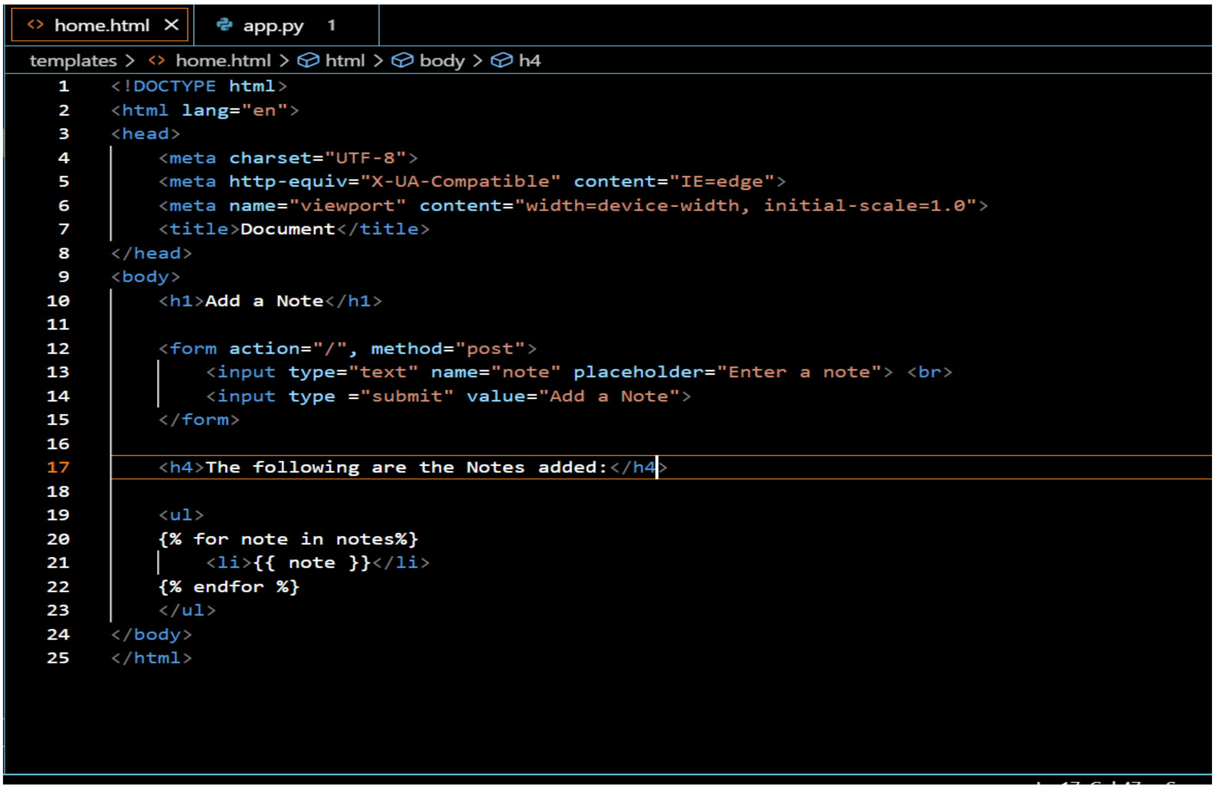
I resolved the POST request issue by updating the code to handle both GET and POST requests by using if statement. Additionally, I fixed the HTML code by linking it to the proper URL and specifying the request method.

The following images shows the fixed code.



```python
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")
        if note:
            notes.append(note)
    return render_template("home.html", notes=notes)


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

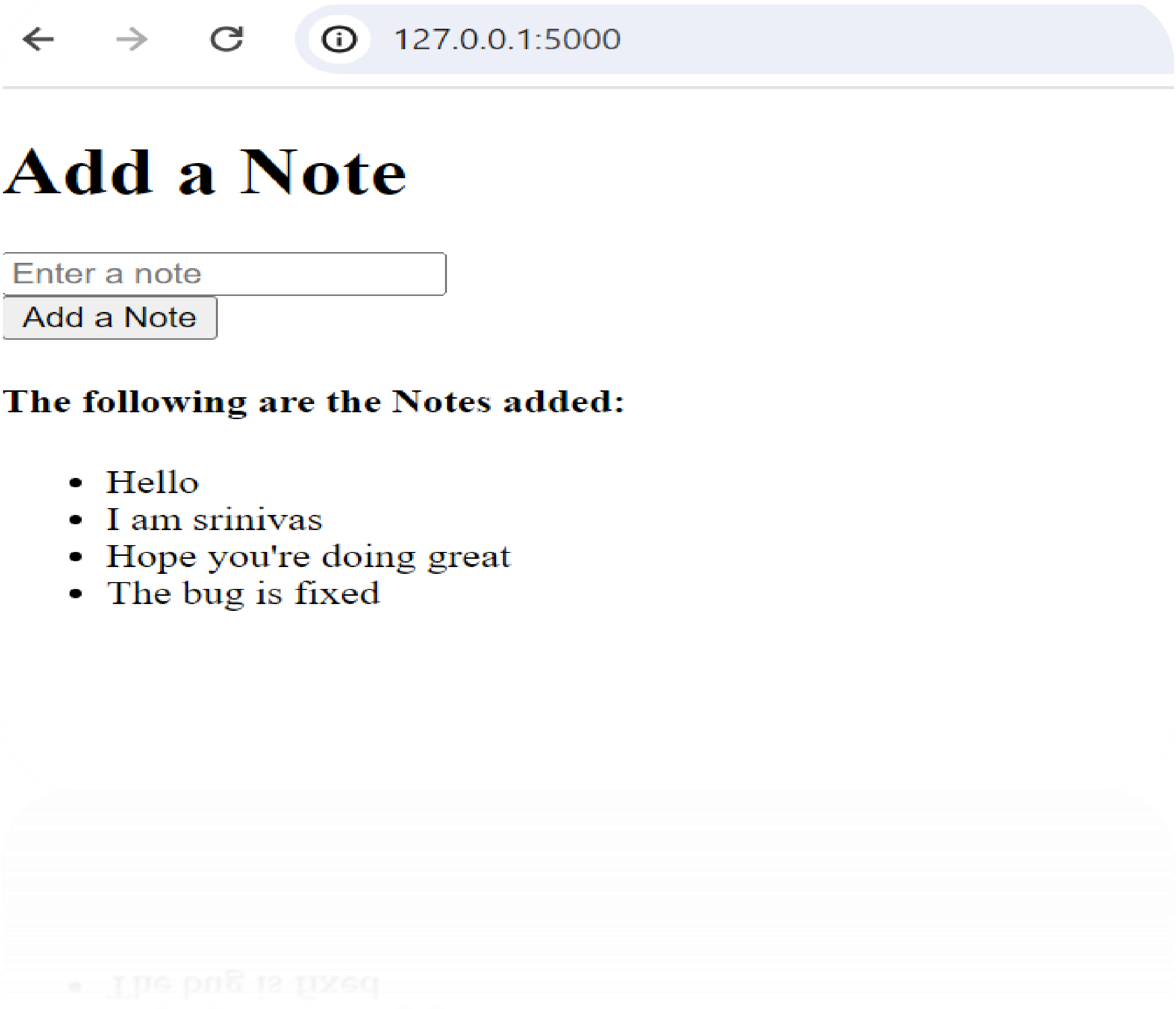
Figure 3: Python flask code after bug fixing



```html
<!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>
    <h1>Add a Note</h1>

    <form action="/", method="post">
        <input type="text" name="note" placeholder="Enter a note"> <br>
        <input type ="submit" value="Add a Note">
    </form>

    <h4>The following are the Notes added:</h4>

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

Figure 4: HTML code after bug fixing

The final outcome of the Note Taking Application looks like this:



Add a Note

The following are the Notes added:

- Hello
- I am srinivas
- Hope you're doing great
- The bug is fixed

Conclusion:

By addressing the issues with POST request handling in the code and fixing the HTML route problem, now the Note Takeing Application is fully functional. Users can seamlessly add notes without encountering any errors.