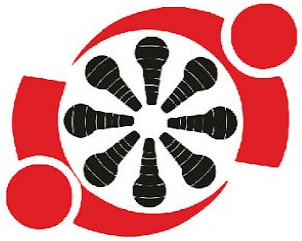


INNOMATICS®
RESEARCH LABS

INNOVATION. AUTOMATION. ANALYTICS

PROJECT ON

Code Refactoring and Bug Fixing For a Note taking application

Prepared By:
Raman Kumar

About me

Background:

Bachelor of Technology in Computer Science & Engineering

Dr. A.P.J. Abdul Kalam Technical University

Linkedin Profile:

linkedin.com/in/ramank97

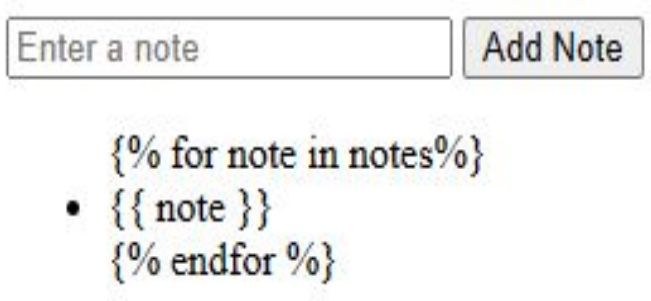
Why Data Science ?

Data science is a transformative field that empowers organizations to extract valuable insights from vast amounts of data. By leveraging statistical analysis, machine learning algorithms,and programming skills, data scientists uncover patterns, trends, and correlations that drive informed decision-making.From optimizing business strategies to enhancing customer experiences, data science plays a crucial role in various industries.

INNOMATICS RESEARCH LABS

Objective of the project

The objective of this report is to provide a thorough documentation of the bug fixing and code refactoring endeavors carried out for the Flask Note Taking App. Through meticulous analysis and prioritization of existing issues, encompassing debugging strategies and code restructuring techniques, the project endeavors to augment the application's stability, performance, and maintainability. By furnishing concrete examples of bug fixes and code enhancements, couple with discussions on encountered challenges and gleaned insights, the report offers valuable elucidation into the process and ramifications of the bug fixing and refactoring initiatives. Initially, the webpage appears entirely static and unresponsive due to malfunctioning code, resembling



Bugs Fixed

Bug 1:Incorrect request method specified in the Flask route

The application Encountered a method error initially due to Routing configuration

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

Solution:- Updated the routing Configuration in app.py to accept both GET and POST requests , resolving the method error and ensuring the Proper Functionality of the app.

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

INNOMATICS RESEARCH LABS

Bugs Fixed:

Bug 2:Incorrect request method in Index Function & missing ‘if note’ condition:

We updated the the route to handle the GET and POST requests, The index Function also needs to be updated to handle both GET & POST request. Also the logic for adding the note to the list.

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

Solution:- The index function has been updated to accommodate both GET and POST requests by checking the request method. Additionally, a condition has been added to verify if the note is not empty before appending it to the list of notes. This addition ensures that only non-empty notes are included in the list, preventing the addition of empty notes. Consequently, after implementing the if note condition, the application now functions properly, ensuring that only meaningful notes are displayed in the list.

Bug Fixed

Bug 3:Incorrect Flask's request object used to retrieve data

The method 'request.args.get("note") is used to retrieve notes here but this method only handles GET requests and not POST requests

```
note = request.args.get("note")
```

Solution:-

The note retrieval code is updated to 'request.form.get("note") to handle the POST requests as well.

```
if request.method=='POST':
    note = request.form.get("note")
```

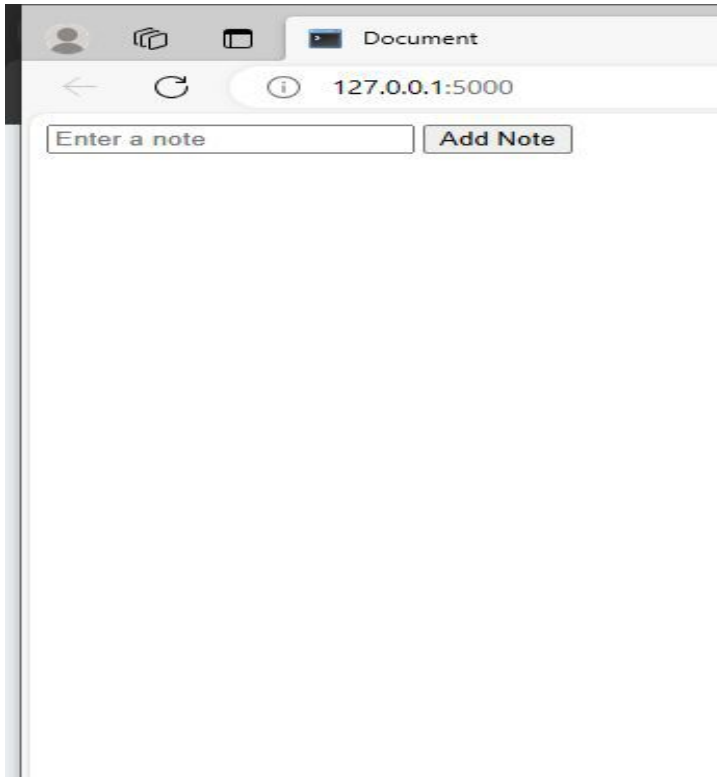
INNOMATICS RESEARCH LABS

Flask Code after the refactoring:-

app.py 1 ×

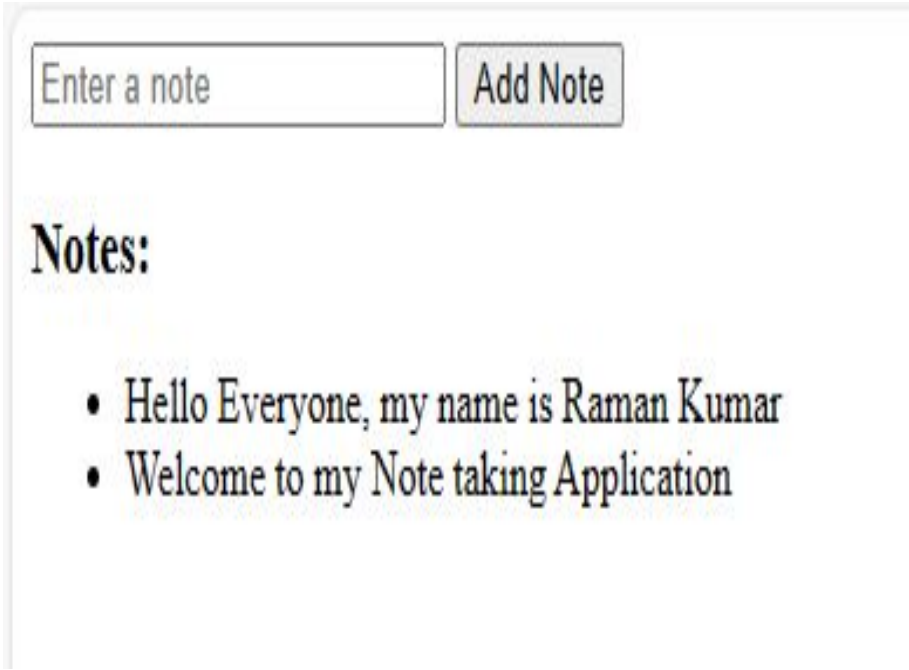
D: > Users > raman > Downloads > note_taking_app > note_taking_app > app.py > index

```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)
```



HTML Code After the Refactoring :-

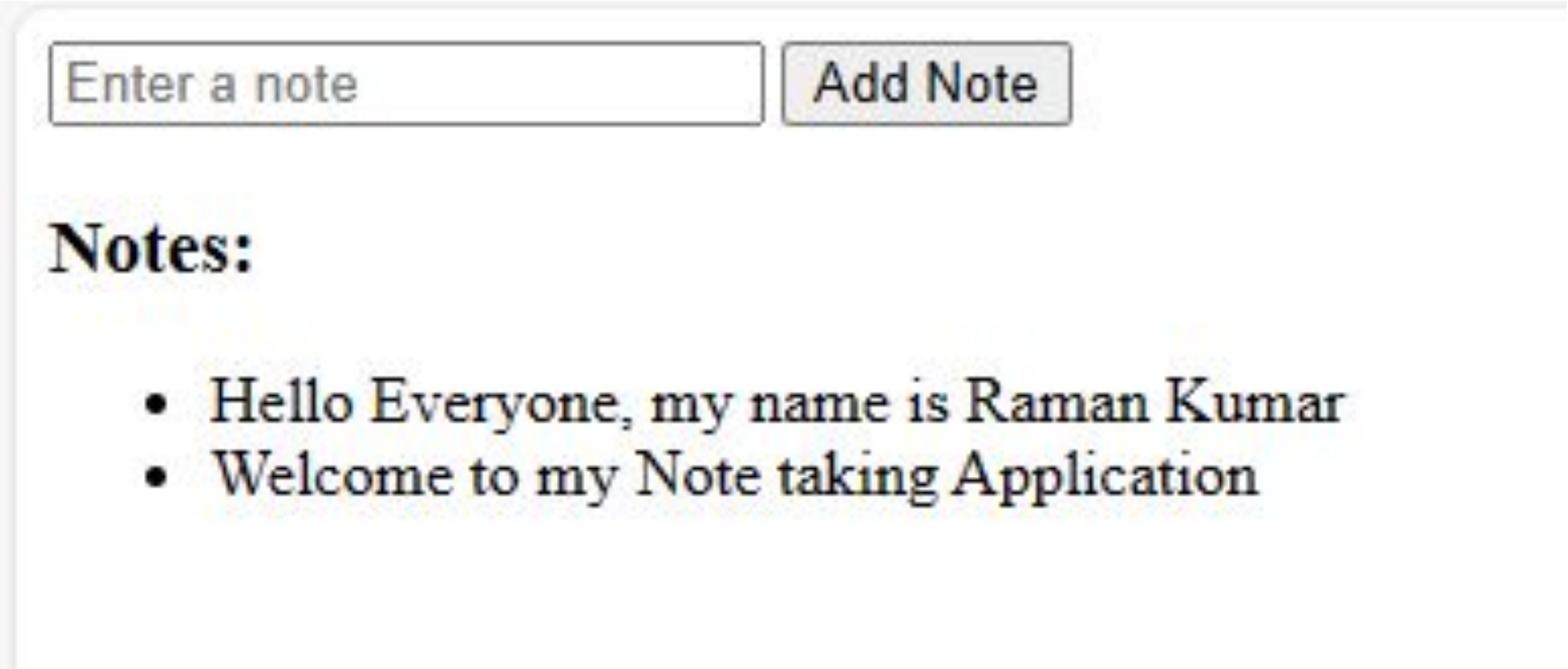
```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <form method="POST">
        <input type="text" name="note" placeholder="Enter a note">
        <button>Add Note</button>
    </form>
    <h3>Notes:</h3>
    <ul>
    {% for note in notes%}
        <li>{{ note }}</li>
    {% endfor %}
    </ul>
</body>
</html>
```



Conclusion:-

- The Note Taking Application project underwent a significant transformation through the process of code refactoring and bug fixing. The primary objective was to enhance the application's performance, and maintainability by addressing existing issues and improving code quality
- The 'if note' condition effectively prevented the addition of empty notes, enhancing data integrity and user experience. Furthermore, updating the 'index' function to handle both GET and POST requests improved the application's responsiveness and usability.

The Snapshot of My Final Note Taking Application's interface



THANK YOU

INNOMATICS RESEARCH LABS