INNOMATICS®
RESEARCH LABS

INNOVATION. AUTOMATION. ANALYTICS

PROJECT ON

Flask-Powered Note-Taking Application: Debugging and Optimization

Presented by: Kandukuri Jaswanth

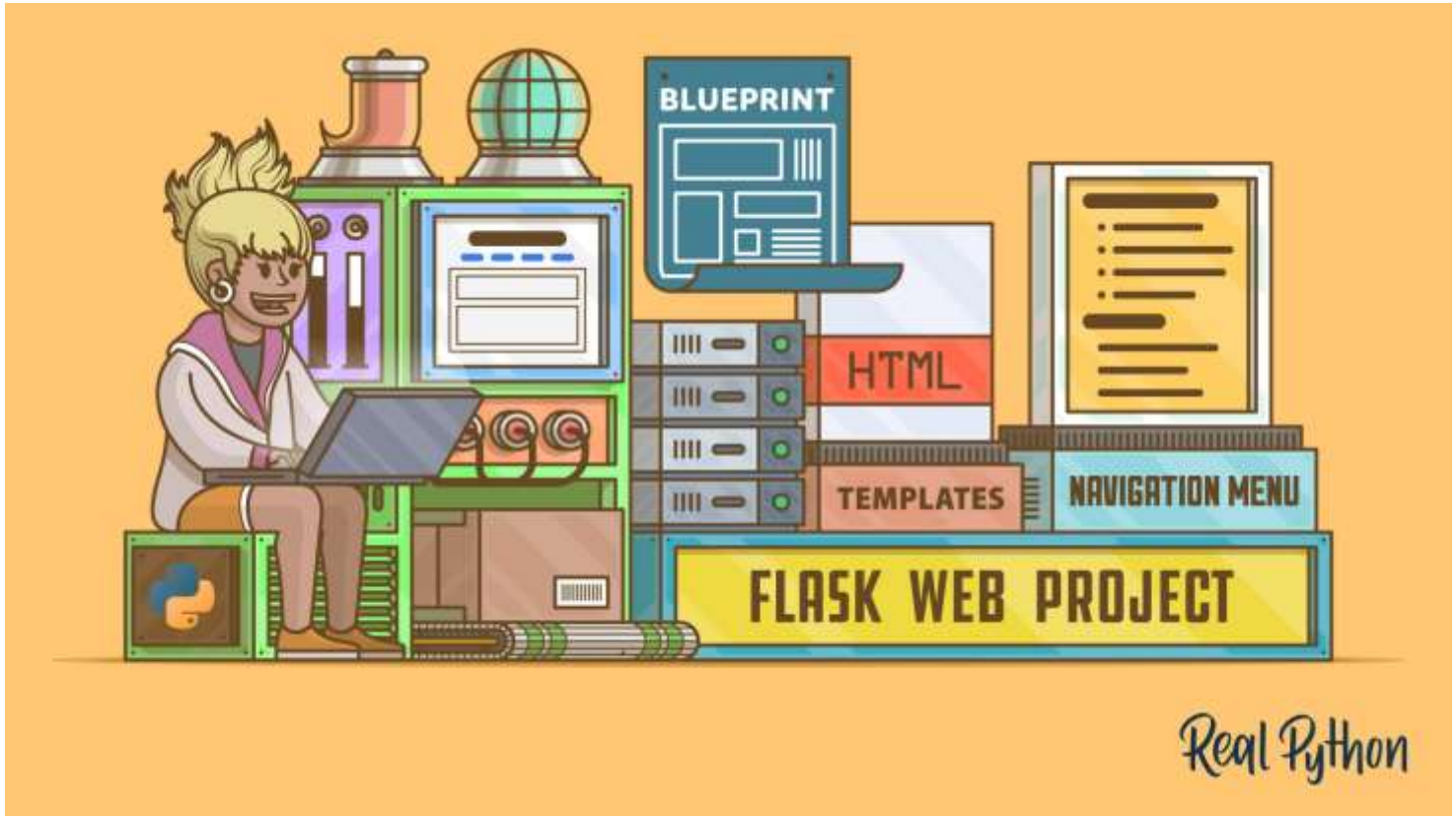
Introduction

The Note-Taking Application is a lightweight, web-based tool designed for users to seamlessly add and view notes. Developed with Flask for the backend and HTML for the frontend, the project aimed to debug and optimize an existing implementation, ensuring efficient functionality and an improved user experience. Through comprehensive debugging and enhancement, key issues such as data handling, form validation, and resubmission errors were addressed, resulting in a more reliable and user-friendly application. This refined version lays the foundation for future enhancements, including database integration, user authentication, and advanced note formatting capabilities.

Project Objectives

1. Develop a fully functional and user-friendly note-taking application using Flask.
2. Identify and resolve existing bugs to enhance application stability and performance.
3. Implement proper data handling techniques to ensure efficient note storage and retrieval.
4. Prevent form resubmission issues to avoid duplicate entries and improve user experience.
5. Enhance input validation mechanisms to prevent empty or invalid note submissions.
6. Optimize the overall application workflow for seamless interaction and usability.

Technologies Used



- **Backend:** Flask (Python)
- **Frontend:** HTML
- **Web Server:** Flask development server

Identified Bugs and Issues

- Form method mismatch: Used GET instead of POST for data submission.
- Incorrect data retrieval: request.args.get() used instead of request.form.get().
- Syntax error: Incorrect HTML syntax within the Python file.
- Form resubmission issue: Notes were re-added on page refresh.
- Empty notes allowed: No validation for empty submissions.
- Missing templates folder: home.html was not placed in the correct directory

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

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

INNOMATICS RESEARCH LABS

Fixes and Improvements

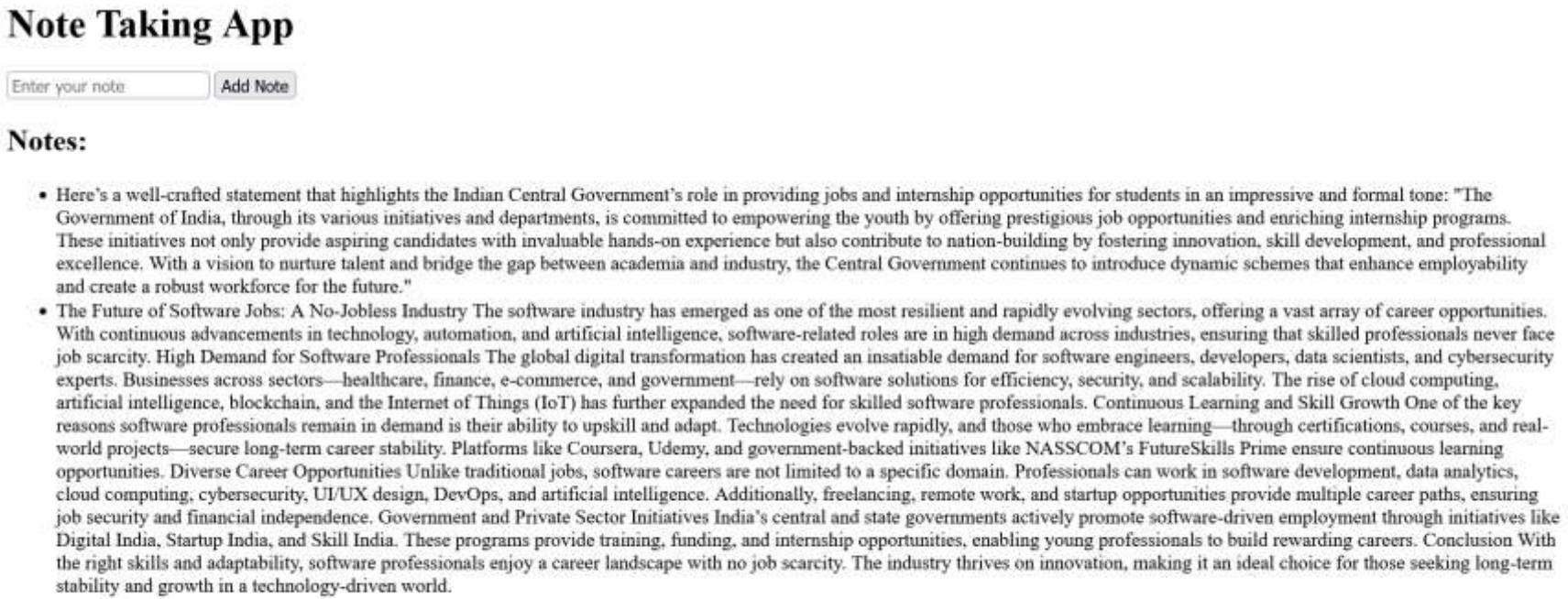
- hanged form method to POST.
- Used request.form.get() to correctly retrieve input data.
- Removed incorrect syntax from app.py.
- Implemented redirect(url_for("index")) to prevent duplicate submissions.
- Added validation to prevent empty notes.
- Ensured home.html is correctly placed inside the templates/ folder

Application Workflow

1. User accesses the home page.
2. User enters a note and clicks "Add Note."
3. The note is saved and displayed below the form.
4. Page reloads automatically to prevent duplicate form submissions.

Testing & Results

- Successfully tested adding notes.
- Verified that notes persist correctly without duplication.
- Checked that empty notes are not accepted.
- Ensured form submission does not cause unintended refresh issues.



INNOMATICS RESEARCH LABS

Conclusion

- The Note-Taking Application is now fully optimized and operational following comprehensive debugging. Key issues, including form validation errors, resubmission glitches, and data handling inconsistencies, have been successfully resolved, significantly enhancing reliability and user experience. The application now ensures a seamless and efficient note-taking process. Future enhancements may include database integration for persistent storage, user authentication for secure access, and advanced formatting options to improve usability. These upgrades will further refine functionality, making the application more robust and user-centric.

Attachments:

Bug Report:
https://docs.google.com/document/d/15TkQhZIEq7Cgt4KB76mDOlqdC2kZgZEClUitqGELs-k/edit?tab=t.0

INNOMATICS RESEARCH LABS