# PROJECT REPORT

**Project Title:** Bug Detection and Fixing

**Name:** Harini M (Team Lead)

**College:** Sri Sairam Engineering College

**Internship Organization name:** Intel (Unnati Industrial training)

**Internship Duration:** 36 days (01.03.2025 to 05.04.2025)

**Mentor:** Dr.P.Vijayakumari

**Date of Submission:** 04.04.2025

# ACKNOWLEDGEMENT

I would like to express my sincere gratitude to **Intel Unnati Industrial Training 2025** for providing me with the opportunity to undertake this internship and gain valuable industry experience. I am immensely thankful to my mentor for their continuous guidance, support, and encouragement throughout the project. Their insights and expertise have been invaluable in shaping my understanding and improving my skills.

I would also like to extend my appreciation to my colleagues and team members for their cooperation, constructive feedback, and willingness to share knowledge, making this experience both enriching and enjoyable. Their support has played a crucial role in the successful completion of this project.

Lastly, I am grateful to my family and friends for their unwavering encouragement and motivation during this journey. This internship has been a significant learning experience, and I look forward to applying the knowledge and skills gained in future endeavors.

**TABLEOFCONTENTS**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **TITLE** | **PAGE NO.** |
| 1 | * PROBLEM STATEMENT * AIM * DESCRIPTION * OVERVIEW * GOAL | 1 |
| 2 | * SOFTWARE REQUIREMENT * PROCEDURE | 2-7 |
| 3 | * WORKING VIDEO * FLOWCHART | 8 |
| 4 | * CONTRIBUTION * CONCLUSION | 9 |

# PROBLEM STATEMENT:

A system which aims to enhance software code quality, reduce debugging time and improve developer productivity by identifying bugs in software code and provide recommended fixes.

# AIM:

The aim of this project is to develop an interactive interface that allows users to detect errors in their Python code and receive recommended fixes along with relevant examples

# DESCRIPTION:

The Python Bug Detector is a web-based application designed to help developers identify and fix errors in their Python code efficiently. The tool provides syntax error detection, runtime error analysis, and common logical error identification, making it a valuable resource for both beginners and experienced programmers. With features like line highlighting for errors and a light/dark theme toggle, the interface ensures a seamless debugging experience.

Built using Flask for the backend, JavaScript for interactivity, and CodeMirror for an intuitive code editing experience, the application allows users to upload Python files or paste code directly into the editor. Once analyzed, the system highlights errors and offers contextual recommendations for fixes, aiding in faster debugging and improved code quality.

The project follows the MIT License, enabling open-source contribution and free usage, allowing developers to modify and enhance the tool as needed.

# OVERVIEW:

* A web-based application designed to detect and fix errors in Python code.
* Supports syntax error detection, runtime error analysis, and common logical error identification.
* Allows users to upload Python files or paste code directly into an interactive editor.
* Highlights errors in the code and provides recommended fixes with examples.
* Features a user-friendly interface with a light/dark theme toggle for better accessibility.
* Built using Flask for the backend, JavaScript for interactivity, and CodeMirror for the code editor.
* Helps developers debug their code efficiently, improving code quality and learning.
* Open-source project licensed under MIT, allowing modifications and contributions.

# GOAL:

* Detect and fix Python code errors efficiently.
* Provide error explanations with recommended fixes.
* Enhance debugging with code highlighting.
* Improve user experience with an intuitive interface.
* Support learning and better coding practices.
* Offer an open-source, customizable solution.

# SOFTWARE REQUIREMENTS:

**1. Frontend:**

* HTML, CSS, JavaScript
* CodeMirror (Code Editor)

**2. Backend:**

* Flask (Python Framework)

**3. Dependencies:**

* Python (Version 3.10)
* Flask

**4. Development Tools:**

* Code Editor (VS Code)
* Git for version control

**5. Browser Compatibility:**

* Chrome

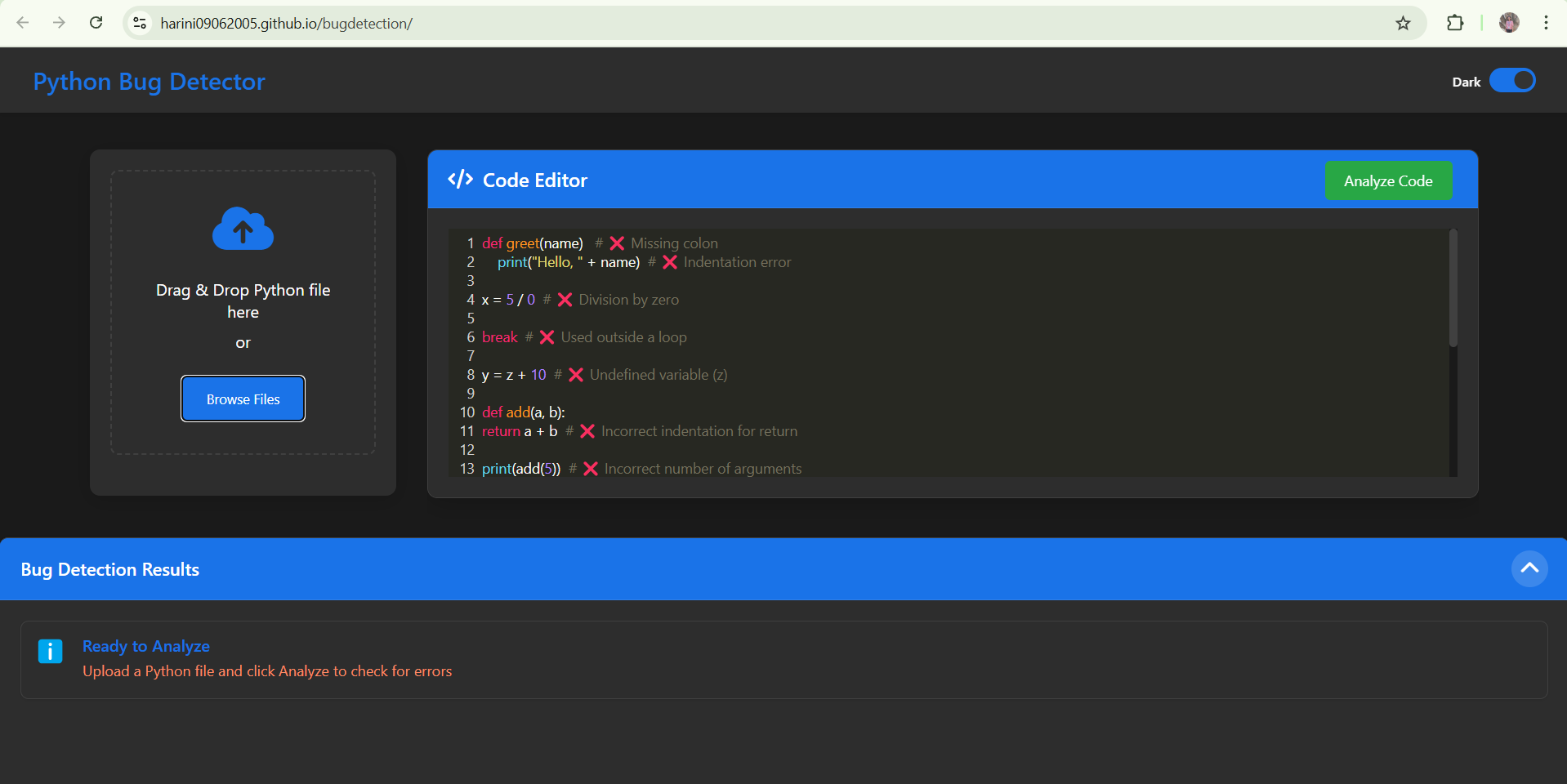
# PROCEDURE:(ON HOW TO USE)

**STEP 1: DRAG AND DROP OR BROWSE A PYTHON FILE:**

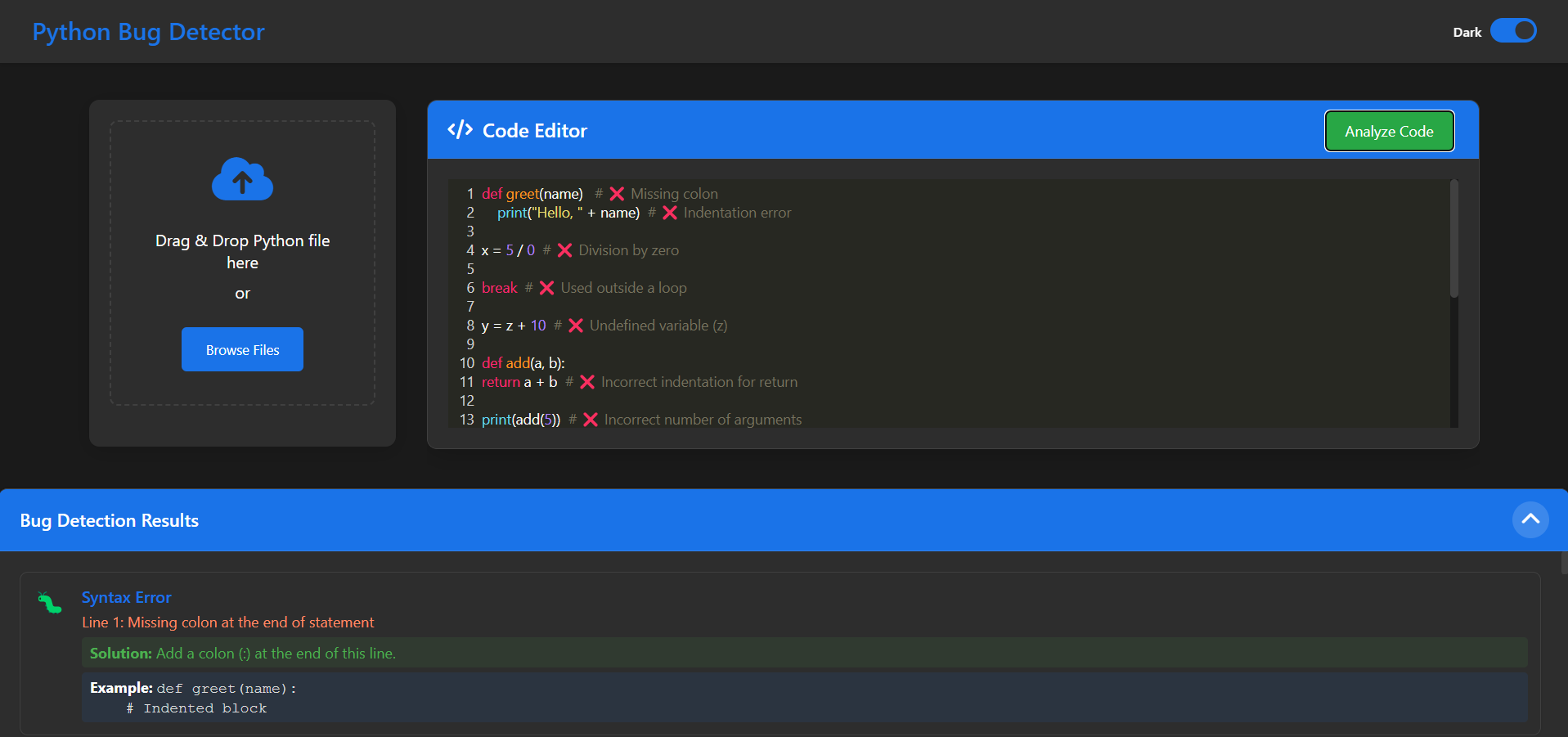
# 

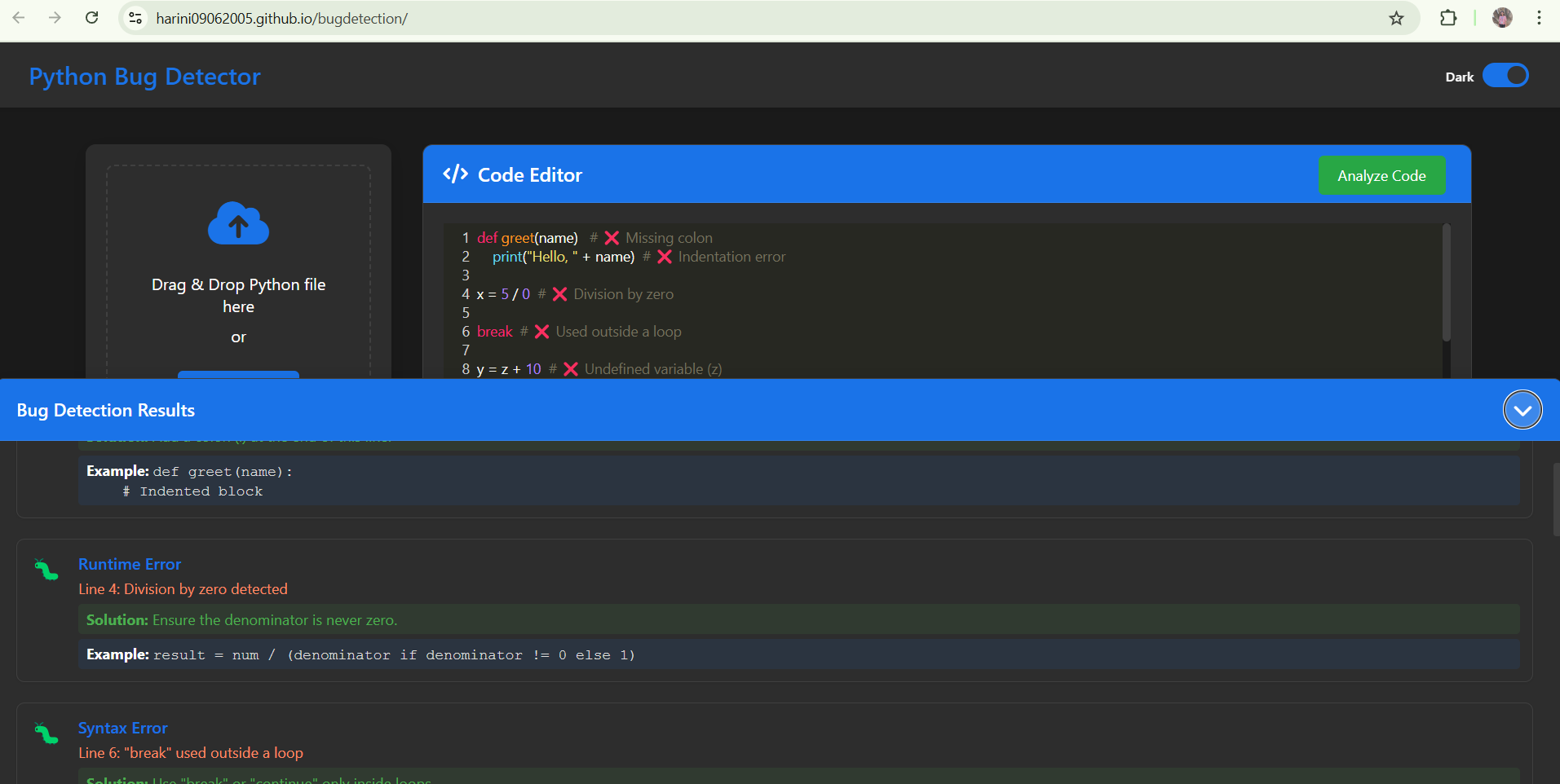
# 

# STEP 2: THE CODE WILL DISPLAY IN THE CODE EDITOR SPACE ,NOW PRESS ANALYZE CODE BUTTON:

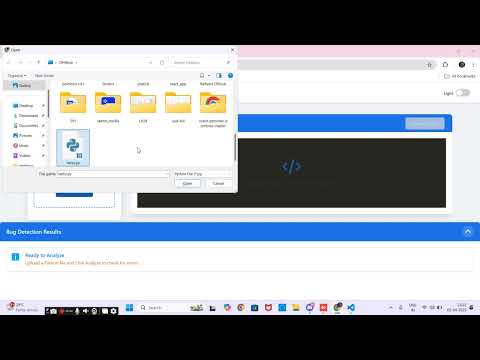


**STEP 3: THE ERRORS PRESENT IN THE CODE WILL DISPLAY IN THE BUG DETECTION RESULT ALONG WITH A SUGGESTION AND EXAMPLE FIX:**

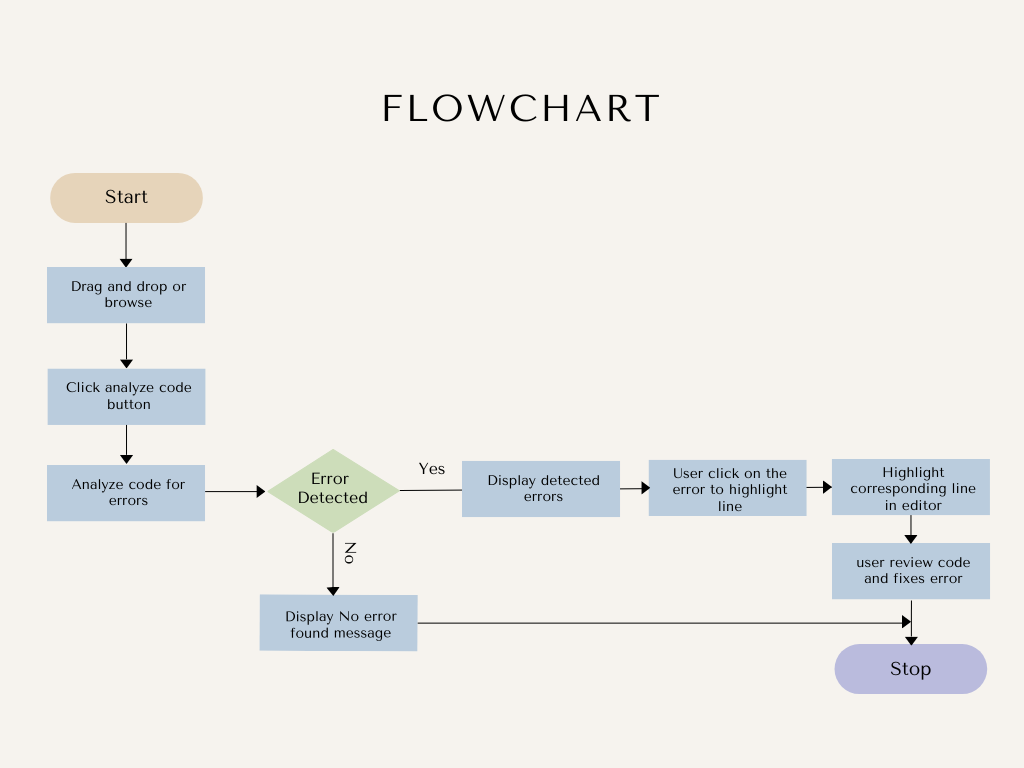


****

**WORKING VIDEO:**

**[](https://www.youtube.com/embed/amdeCUfoJsU?feature=oembed)**

**FLOWCHART:**

****

**CONTRIBUTION:**

* **Developed the Complete UI:** Designed and implemented the front-end using HTML, CSS, JavaScript, and CodeMirror for an interactive code editor.
* **Integrated Error Detection with Backend:** Connected the frontend to the Flask-based backend, enabling real-time error detection and analysis.
* **Debugged and Optimized Code:** Identified and resolved critical issues to ensure smooth functionality and accurate bug detection.
* **Implemented Error Suggestions and Fixes:** Provided users with recommended fixes and example corrections for detected errors.
* **Deployed the Application:** Successfully hosted the project for public use, ensuring accessibility and scalability.

**CONCLUSION:**

The Python Bug Detector provides an efficient way to detect and fix Python code errors with syntax, runtime, and logical analysis. With an interactive UI, error highlighting, and fix suggestions, it enhances debugging for developers. Built with Flask, JavaScript, and CodeMirror, it ensures smooth performance and accessibility. Future improvements could include AI-powered fixes and multi-language support for greater functionality.