**Code Bottleneck Analyzer**

An AI-powered tool to detect performance bottlenecks in code snippets and suggest optimizations. It includes a simple web interface for users to upload code and view results.

**Features**

- Detects common performance issues (e.g., nested loops, function calls in loops).

- Provides optimization suggestions based on static analysis.

- User-friendly interface for uploading code and viewing results.

**a). How to Run the Project**

Prerequisites

- Python 3.8+

- Pip (Python package manager)

Installation

1. Install Dependencies

pip install flask pygments

2. Run the application

python app.py

3. Open the browser and navigate to

http://127.0.0.1:5000

**b). Design Choices:**

1. Backend:

Built using Flask for simplicity and rapid prototyping.

Static analysis is performed using regex and logic rules for known bottlenecks.

2. Frontend:

A minimal HTML interface for uploading code and displaying results.

JavaScript is used to fetch and render analysis results dynamically.

3. Expandable Architecture:

The analyze code function can be extended to support advanced checks or integrate AI models.

**c). Assumptions and Limitations:**

**Assumptions**:

i).The tool is designed for Python code but can be extended for other languages.

ii). Input code is provided as .py files or plain text.

**Limitations**:

i). Currently detects only predefined patterns of inefficiencies (e.g., nested loops).

ii). Advanced optimizations require AI integration, which is not included in this version.

**d). Screen Shots:**

