SRM UNIVERSITY AP

Department of Computer Science and Engineering.

Software Engineering Project Report on

"AI-Powered Code Review Assistant"

Arshad Shaik (AP21110010926)
Akash (AP21110010927)
Dhakshayani G (AP21110010930)
Akhil (AP21110010931)

CSE-O

Guided by,
Assistant Prof. Anusha Nalajala

Table of Contents :

1) Abstract :

Our project introduces an AI code review assistant, acting as a knowledgeable companion for programmers. This initiative serves as a supportive tool for individuals engaged in coding tasks. It offers automated suggestions on how to enhance the quality of code.

This project boasts proficiency in comprehending both code and everyday language. It conducts comprehensive checks to ensure code quality and identifies potential security vulnerabilities, thereby fostering collaboration within the team. The development of this assistant heavily relies on Python, ensuring its intelligence and usefulness for both experienced and novice coders. Ultimately, the primary objective is to elevate the coding experience, making it more enjoyable, while also facilitating seamless teamwork. The overarching goal is to infuse coding with enjoyment and ensure smooth collaboration within teams.

2) Introduction :

An AI-powered code review assistant is a software tool created to make the code review process smoother and more effective in software development. In traditional code reviews, developers look through each other's code manually to find errors, style issues, and other problems. But with AI technology, we can make this process even better.

These assistants use machine learning and natural language processing to check code faster and better than humans can alone. They can find potential bugs, spots where the code might be slow, security risks, and whether the code follows the rules automatically.

Not only that, but they also give helpful advice to developers on how to make their code better. By taking care of repetitive tasks and offering smart suggestions, these assistants help teams save time, make fewer mistakes, and deliver software that works well.

In the end, these assistants are really important in modern software development. They help teams work together better, keep the code up to standards, and get reliable and safe software out the door faster.

3) Existing System/ Literature review :

GitHub Code Scanning: GitHub's code scanning feature provides automated security analysis for code repositories, detecting potential vulnerabilities and coding errors using static analysis techniques. It offers actionable insights directly within the GitHub interface, facilitating proactive code review and enhancement.

Open-Source Tools: Various open-source tools contribute to code analysis and review, incorporating AI techniques to improve accuracy and efficiency. Examples include ESLint and Pylint for linting JavaScript and Python code, respectively, and SonarQube for comprehensive static code analysis across multiple programming languages. These tools offer customizable configurations and integrations, empowering developers to maintain high code quality standards.

Literature Review: Academic research in this domain explores diverse methodologies and technologies for AI-powered code review assistants. For instance, studies delve into static analysis, dynamic analysis, and machine learning algorithms applied to code analysis tasks. Noteworthy topics include accuracy enhancement, scalability for large codebases, and seamless integration with existing development workflows. A comprehensive literature review provides insights into the evolving landscape of AI-driven code review and highlights avenues for further exploration and improvement.

4) System Requirements:

1. Hardware Requirements:

   - Multi-core processor and ample RAM for efficient processing.
   - Sufficient disk space for storing code and analysis data.

2. Software Requirements:

   - Compatibility with major OSs and programming languages.
   - Integration with Git, AI frameworks, and web interfaces.

3. Scalability and Performance:

   - Scalable architecture for handling growing codebases and user loads.
   - Algorithm and pipeline optimization for minimal latency.

4. Customization and Configuration:

   - Flexible customization options for project needs and standards.
   - Extensibility via plugins and integrations.

5. Visual Studio Code
6. NodeJS
7. HTML
8. CSS
9. JS
10. Flask A

5) Proposed System

1. Functionality :

   - Automated Code Analysis
   - Bug Detection
   - Security Vulnerability Identification
   - Code Style Adherence Checks

2. Architecture :

   - Backend Infrastructure
   - Code Analysis Algorithms
   - User Interface Components
   - Integration with External Systems

3. User Interface :

   - Actionable Insights
   - Suggestions
   - Reports

4. Integration :

   - IDE Integration
   - Version Control System Integration
   - CI/CD Pipeline Integration
   - Project Management Platform Integration

5. Customization :

- Language Support
- Coding Standards
- Project-Specific Requirements

6. Scalability and Performance :

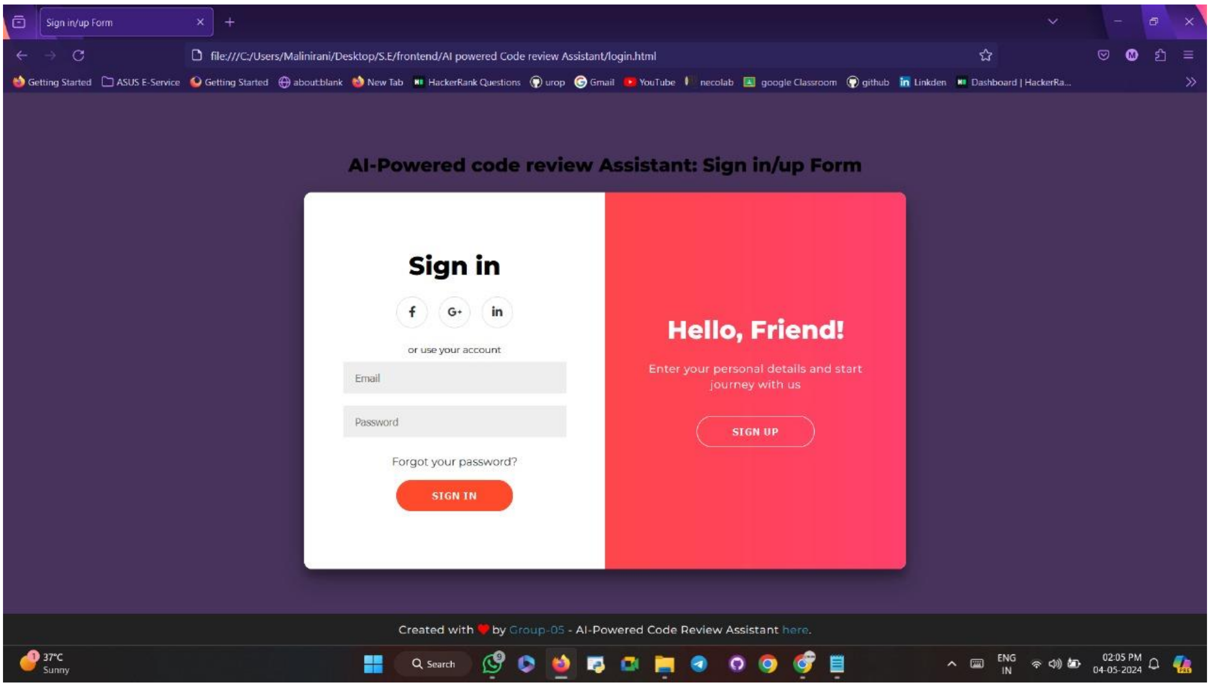
- Handling Large Codebases
- Optimization Strategies

7. Security :

- Access Controls
- Encryption
- Secure Communication Protocols

6) Results/ Screenshots:

a) LOGIN –



AI-Powered code review Assistant: Sign in/up Form

Sign in

or use your account

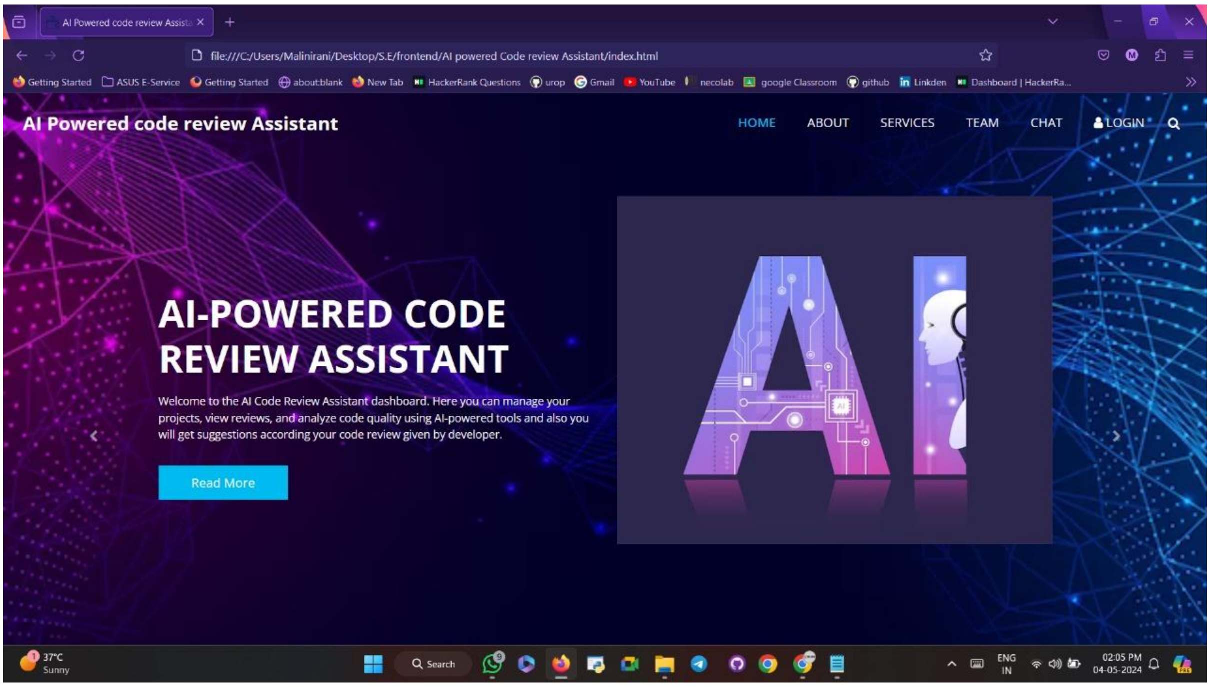
Forgot your password?

Hello, Friend!

Enter your personal details and start journey with us

Created with ❤ by Group-05 - AI-Powered Code Review Assistant here.

b) INDEX

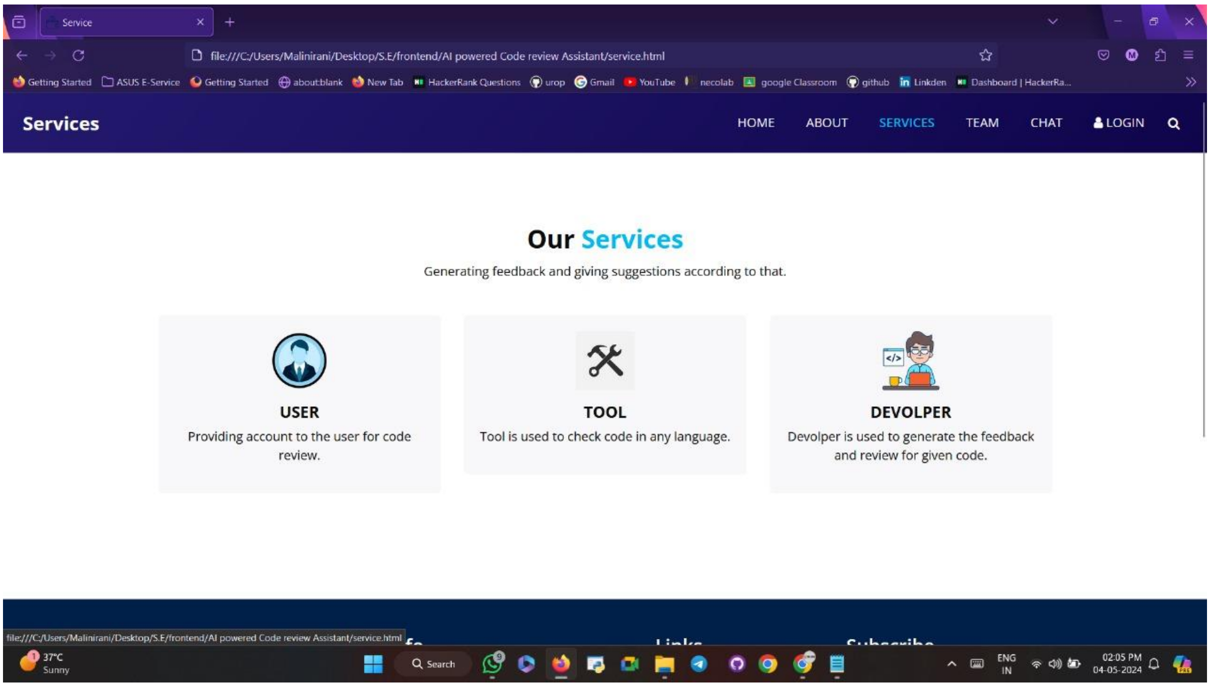


AI Powered code review Assistant

AI-POWERED CODE REVIEW ASSISTANT

Welcome to the AI Code Review Assistant dashboard. Here you can manage your projects, view reviews, and analyze code quality using AI-powered tools and also you will get suggestions according your code review given by developer.

c) SERVICES


Service
file:///C:/Users/Malinirani/Desktop/S.E/frontend/AI powered Code review Assistant/service.html
Services
HOME
ABOUT
SERVICES
TEAM
CHAT
LOGIN

Our Services

Generating feedback and giving suggestions according to that.

USER

Providing account to the user for code review.

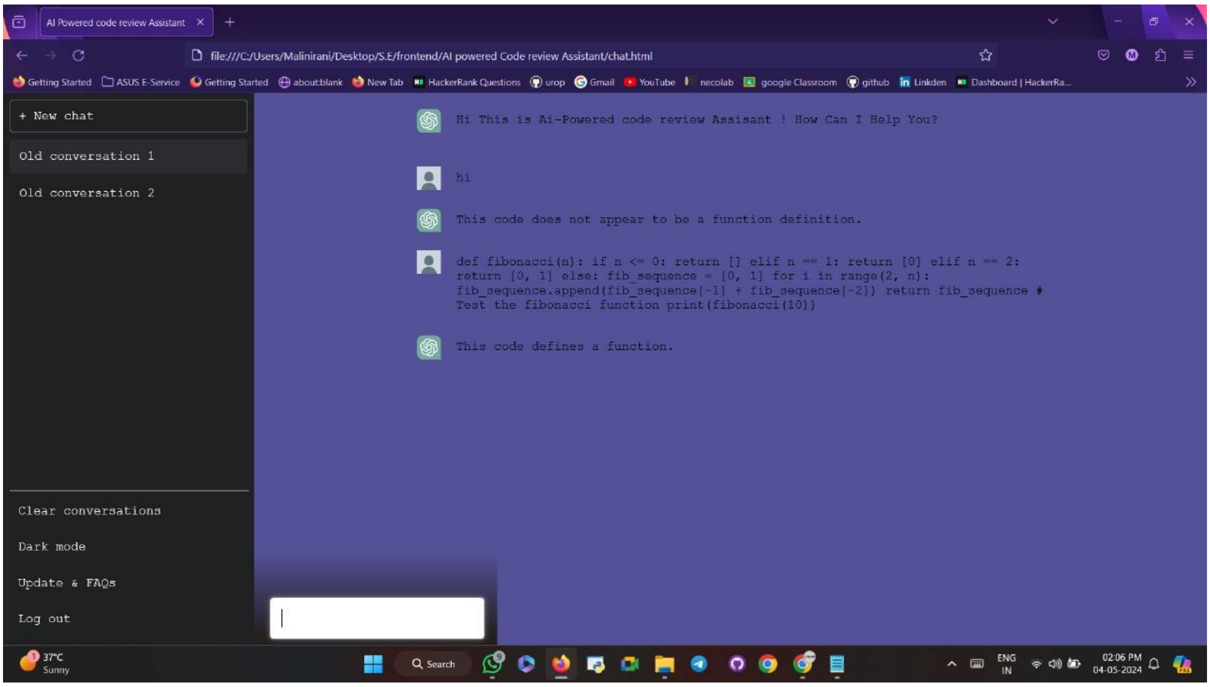
TOOL

Tool is used to check code in any language.

DEVOLPER

Devolper is used to generate the feedback and review for given code.

d) CHAT


AI Powered code review Assistant
file:///C:/Users/Malinirani/Desktop/S.E/frontend/AI powered Code review Assistant/chat.html
+ New chat
Old conversation 1
Old conversation 2
Clear conversations
Dark mode
Update & FAQs
Log out

```
Hi This is Ai-Powered code review Assisant ! How Can I Help You?

hi

This code does not appear to be a function definition.

def fibonacci(n): if n <= 0: return [] elif n == 1: return [0] elif n == 2:
return [0, 1] else: fib_sequence = [0, 1] for i in range(2, n):
fib_sequence.append(fib_sequence[-1] + fib_sequence[-2]) return fib_sequence #
Test the fibonacci function print(fibonacci(10))

This code defines a function.
```

7) Conclusions :

In conclusion, the proposed AI-powered code review assistant presents a comprehensive solution for enhancing the code review process in software development. By leveraging artificial intelligence techniques, the assistant automates code analysis, detects bugs, identifies security vulnerabilities, and ensures adherence to coding standards. Its architecture enables seamless integration with existing development tools and workflows, while customization options allow users to tailor its behavior to specific project requirements. Additionally, scalability and performance optimization strategies ensure efficient handling of large codebases and timely feedback to developers. With robust security measures in place, the assistant maintains the confidentiality and integrity of the code review process. Overall, the proposed system aims to improve code quality, productivity, and collaboration among software development teams, ultimately contributing to the delivery of reliable and secure software products.

8) References :

Google & GitHub

Thankyou