Work Sample: Reducing Developer Surprises with Intelligent PR Summaries

Overview

Date: 2024
Project Type: Web Development / Generative AI
Role: Software Developer

During a yearly team summit, developers were encouraged to identify opportunities to integrate AI into our workflows. I collaborated with two teammates to address a common developer frustration: the lack of insight into potentially problematic files. The tool we envisioned highlights recurring issues within a file's history, providing proactive insights before development begins.

Pain Point

In development, surprises mid-build happen all the time. You plan thoroughly and dive into the code to uncover hidden complexities or recurring issues within a file. These unexpected challenges often require a change in strategy, typically occurring late in the process. If developers could identify trends and problem areas before planning or coding, they could avoid unnecessary rework and reduce the number of review cycles.

Solution

We designed an automated GitHub workflow powered by ChatGPT to analyze recent comment history over time and directly surface relevant context in new pull requests. This design enables developers to anticipate common pain points and approach their work with more informed strategies.

Our team divided responsibilities as follows:

- Developer A: Created the vector database and synced 30 days of Git comment history.
- Me: Authored the AI prompt, designed the prompt logic, and implemented the top-level GitHub Action that triggers the workflow.
- Developer B: Built the script that sends the AI request and parses the response for inclusion in the pull request comment.

The result was an AI-enhanced development tool that catches hidden file issues and adds historical context when developers need it most.

Execution

The workflow is triggered via GitHub Actions when a pull request is opened. Once a pull request is opened, the action automatically retrieves relevant comments from the past 30 days, embeds them into a vector store, sends a tailored prompt to ChatGPT, and posts a concise, contextual comment on the pull request. This comment informs the developer of recurring concerns or unusual file activity, giving them a head start before coding or reviews begin.

Conclusion

The tool was well-received by the team. Developers saw the potential for fewer surprise issues during reviews and a clearer understanding of file history during planning. This experiment demonstrated how targeted AI integration can meaningfully streamline processes and improve developer experience.