CHITKARA UNIVERSITY

PROJECT SYNOPSIS REPORT

ON

DevInsights

SUBMITTD TO

Dr. Lekha Rani

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

FOR

INTEGRATED PROJECT (22CS038)

Submitted by

Name : Ishaan Singla (2210992582) ,

Sanya (2210992255),

Sanskriti(2210992252),

Nutan(2210992005)

Semester : 6

Session: 2022-2026

CHITKARA UNIVERSITY

Index

Problem Statement:

Developers often face challenges in understanding and managing code changes in large repositories. Manually reviewing commit messages and searching through code can be time-consuming and inefficient. There is a need for an intelligent tool that can summarize commits, provide meaningful insights, and enable code search using natural language, making the development process faster and more efficient.

Title of project:

DevInsights

Objective & Key Learnings:

To develop an AI-powered web application that helps developers gain insights from their code repositories by providing AI-generated commit summaries and enabling natural language-based code search.

Key learnings :

- Deep understanding of Next.js for building server-side rendered web applications.
- Integration of AI APIs (Google Gemini, Assembly AI) for commit summarization and code search functionality.
- Working with Clerk for user authentication and Neon Console for database management.
- Deployment using Vercel and managing GitHub API for repository linking.
- Understanding **Bun** and its use in optimizing JavaScript bundling and server-side rendering.
- Leveraging **Bun** to improve the performance and speed of building the project.

Options available to execute the project:

1) Programming Languages:

a) JavaScript & TypeScript :

- JavaScript is used for both frontend and backend, ensuring seamless development across the stack.
- TypeScript adds type safety, making code more robust and maintainable.

2) Frontend Frameworks :

a) **Next.js** :

- A React-based framework that supports **server-side rendering (SSR)** and **static site generation (SSG)** for better performance.
- Provides built-in routing, API routes, and SEO optimization.

b) **React.js** :

- A powerful JavaScript library for building interactive UIs with a component-based architecture.

3) Backend Frameworks & services :

a) **Node.js** :

- A JavaScript runtime built on Chrome's V8 engine, ideal for handling asynchronous operations efficiently.

b) **Api integration** :

- **Google Gemini AI**: Used for AI-powered commit summaries and intelligent code search.
- **GitHub API**: Fetches commit history, repository details, and enables interaction with linked repositories.

c) **Bun**:

- fast JavaScript runtime that enhances the speed of the development process, bundling, and server-side rendering, improving both backend performance and efficiency..

d) **Clerk (Authentication Service)**:

- A user authentication and access control service that simplifies secure login management.

4) Version Control:

a) **Git & GitHub**:

- Git is a widely used **version control system**, allowing tracking of code changes.
- GitHub provides cloud-based repository hosting with collaboration tools

5) Database:

a) **Neon Console (PostgreSQL-based DB)** :

- A managed cloud database offering serverless capabilities and high-performance scalability.

6) Deployment & Hosting:

a) **Vercel**:

- A platform optimized for **Next.js** applications, providing seamless deployment and serverless computing.

Advantages

1) AI-Powered Code Understanding

- Automatically generates commit summaries, reducing manual effort in tracking changes.
- Helps developers quickly understand project updates without reviewing entire commits.

2) Smart Code Search with Natural Language

- Enables developers to search for code snippets or logic using plain English.
- Improves efficiency by eliminating the need for manual repository navigation.

3) Improved Collaboration & Productivity

- Teams can quickly grasp recent changes and contributions.
- Helps onboard new developers faster by summarizing past work.

4) Seamless Integration with GitHub

- Works directly with GitHub repositories, making it easy to link and manage projects.
- Uses the GitHub API to fetch real-time data, ensuring up-to-date insights.

5) Web-Based & User-Friendly

- No need to install additional software—accessible from any browser.
- Provides a clean and intuitive UI using **Next.js, ShadCN, and Clerk for authentication**.

6) Scalable & Cost-Effective Deployment

- Hosted on **Vercel**, ensuring high availability and fast performance.
- Uses **Neon Console (PostgreSQL)** for efficient database management.

Disadvantages

1) AI Accuracy & Limitations

- AI-generated summaries may not always be **100% accurate** or capture the full context of changes.
- Natural language code search may struggle with **complex queries or ambiguous phrasing**.

2) Dependence on External APIs:

- Relies on **Google Gemini AI and GitHub API**, meaning service downtime or API rate limits could affect functionality.
- Potential **API cost implications** if usage scales up significantly.

3) Limited Offline Functionality

- Since it's a **web-based platform**, it requires an active internet connection.
- No local repository analysis without cloud-based access.

4) Learning Curve for Users

- Developers unfamiliar with **AI-powered tools** might need time to adapt.
- Some may **prefer traditional methods** over AI-generated insights.

References :

- Next.js Documentation – https://nextjs.org/docs
- Google Gemini AI - https://ai.google.dev/gemini-api/docs
- GitHub API Documentation - https://docs.github.com/en/rest
- Vercel documentation - https://vercel.com/docs
- Neon PostgreSQL - https://neon.tech/docs/reference/api-reference
- Clerk (user authentication) - https://clerk.com/docs/quickstarts/nextjs
    - https://clerk.com/docs/references/nextjs/custom-sign-in-or-up-page
- ShadCn UI library - https://ui.shadcn.com/docs