

Hackathon Project Phases Template

Project Title:

Project Insight: AI Feedback Generator for Development Teams using Palm's models/text-bison-001

Team Name:

Datastorm

Team Members:

- B.Kaushal
 - V.Dheeraj Kumar
 - A.Sai Ratna
 - Y.Shruthi
 - P.Suvitha Reddy
-

Phase-1: Brainstorming & Ideation

Objective:

Develop an AI-powered Project Insight website using Gemini Flash to help users compare and analyze and modify their project code based on feedback provided.

Key Points:

1. Problem Statement:

- Software development teams struggle with manual code reviews, which are time-consuming, inconsistent, and prone to errors.
- Ensuring code quality, security, and best practices across large codebases is challenging.

2. Proposed Solution:

Project Insight uses **Google Gemini Flash API** (advanced version of PaLM's models/text-bison-001) to analyze source code, detect issues, and provide AI-driven feedback with precise change tracking. The chatbot-style interface and vertical code comparison help developers quickly understand and apply improvements.

3. Target Users:

- **Working Professionals & Development Teams** looking for specifications and comparisons **and** for debugging and testing.
- **Code reviewers & QA engineers** for efficient validation.
- **Students** for optimizing their project and code feedback.

4. Expected Outcome:

- A functional **AI-powered Project Insight Website** that provides insights based on their query to project and **README**.
-

Phase-2: Requirement Analysis

Objective:

Project Insight follows a **modular and API-driven architecture**, leveraging **Google Gemini Flash API** for AI processing, a **Node.js backend**, and a **Next.js frontend with shadcn** for a smooth user experience.

Key Points:

1. Technical Requirements:

- Programming Language: **Python**
- Backend: **Google Gemini Flash API, Node JS,**
- Frontend: **Next JS, shadcn**
- Database: **Not required initially (API-based queries)**

2. Functional Requirements:

- Chat-Based Folder Input
- AI-Powered Code Review & Modification using **Gemini Flash API**.
- Code Difference Tracking.
- **README** File Updates
- Code Modification.

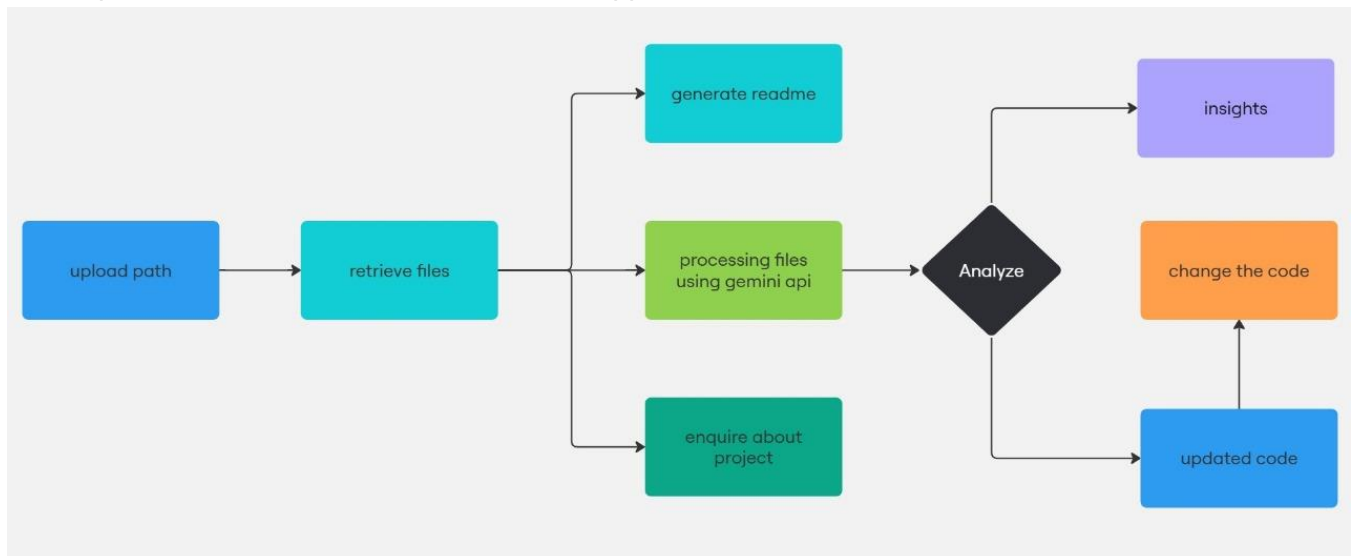
3. Constraints & Challenges:

- Ensuring real-time updates from **Gemini API**.
- Providing a **smooth UI experience** with NextJS.

Phase-3: Project Design

Objective:

Develop the architecture and user flow of the application.



Key Points:

1. System Architecture:

- User gives the folder path of project as query via UI.
- Folder is processed using **Google Gemini API**.
- AI model fetches code feedback.
- The frontend displays **Code Diff Viewer, Chat Based Interaction**.

2. User Flow:

- Step 1: User enters a query (e.g., "Path of your project").
- Step 2: The backend **calls the Gemini Flash API** to retrieve feedback and code changes.
- Step 3: The website processes the code and **displays results** in an easy-to-understand format.
- Step 4: Users can chat about their project.
- Step 5: Users can automatically get the created README file. (Also can download the file)

3. UI/UX Considerations:

- **Minimalist, user-friendly interface** for seamless navigation.
 - **Chat-Based Interaction** related to given project
 - **Code Diff Viewer**.
 - **Performance & Speed**
 - **Changes can be done to original file by clicking change button**
-

Phase-4: Project Planning (Agile Methodologies)

Objective:

Break down development tasks for efficient completion.

| Sprint | Task | Priority | Duration | Deadline | Assigned To | Dependencies | Expected Outcome |
|----------|-------------------------------------|---------------------------------|-------------------|--------------|---------------------|---|---------------------------------------|
| Sprint 1 | Environment Setup & API Integration | <input type="checkbox"/> High | 6 hours (Day 1) | End of Day 1 | Kaushal & Sai Ratna | Google API Key, javaScript, Next JS setup | API connection established & working |
| Sprint 1 | Frontend UI Development | <input type="checkbox"/> Medium | 2 hours (Day 1) | End of Day 1 | Dheeraj | API response format finalized | Basic UI with input fields |
| Sprint 2 | Code Difference and changes | <input type="checkbox"/> High | 3 hours (Day 2) | Mid-Day 2 | Kaushal & Suvitha | API response, UI elements ready | Vertical code comparison |
| Sprint 2 | Error Handling & Debugging | <input type="checkbox"/> High | 1.5 hours (Day 2) | Mid-Day 2 | Shruthi & Sai Ratna | API logs, UI inputs | Improved API stability |
| Sprint 3 | Testing & UI Enhancements | <input type="checkbox"/> Medium | 1.5 hours (Day 2) | Mid-Day 2 | Entire team | API response, UI layout completed | Responsive UI, better user experience |
| Sprint 3 | Final Presentation | <input type="checkbox"/> Low | 1 hour (Day 2) | End of Day 2 | Entire Team | Working prototype | Demo-ready project |

Sprint Planning with Priorities

Sprint 1 – Setup & Integration (Day 1)

- ☐ **High Priority**) Set up the **environment** & install dependencies.
- ☐ **High Priority**) Integrate **Google Gemini API**.
- ☐ **Medium Priority**) Build a **basic UI** with input fields.

Sprint 2 – Core Features & Debugging (Day 2)

- ☐ **High Priority**) Implement **Chat Based Interaction** related to project, **README**, **Code Diff viewer**.
- ☐ **High Priority**) Debug API issues & handle **errors in queries**.

Sprint 3 – Testing, Enhancements & Submission (Day 2)

- ☐ **Medium Priority**) Test API responses, refine UI, & fix UI bugs.
- ☐ **Low Priority**) Final **demo preparation**.

Phase-5: Project Development

Objective:

Implement core features of the **Project Insight** website.

Key Points:

- 1. **Technology Stack Used:**
 - **Frontend:** Next JS,shadcn
 - **Backend:** Google Gemini Flash API,Express,React
 - **Programming Language:** Javascript
- 2. **Development Process:**
 - Implement **API key authentication** and **Gemini API integration**.
 - Develop **Code comparision & chat Interaction**.
 - Optimize **code of given query & update changes in original files,README**.
- 3. **Challenges & Fixes:**
 - **Challenge:** Collaboration
 - **Fix:** Enabling developers to iterate faster with AI-assisted recommendations..
 - **Challenge:** Traceability
 - **Fix:** Highlighting exact lines of change..

Phase-6: Functional & Performance Testing

Objective:

Ensure that the **ProjectInsight** website works as expected.

| Test Case ID | Category | Test Scenario | Expected Outcome | Status | Tester |
|--------------|--------------------|--|--|----------|----------|
| TC-001 | Functional Testing | Query "Path to project folder: C:\Users\suman\Downloads\sign_lan\Sign-Language- | Code Difference viewer and Optimization. | ✔ Passed | Kausha I |

| | | | | | |
|--------|--------------------------|--|--|----------|-----------|
| | | Translator" | | | |
| TC-002 | Functional Testing | Query "Chat interaction regarding project" | Change in code and extra modules used. | ✓ Passed | Sai Ratna |
| TC-003 | Performance Testing | API response time under 500ms | API should return results quickly. | ✓ Passed | Suvitha |
| TC-004 | Bug Fixes & Improvements | Fixed incorrect API responses. | Data accuracy should be improved. | ✓ Fixed | Shruthi |
| TC-005 | Prototype | Ensure UI is responsive for given query. | UI should work on desktop. | ✓ Passed | Dheeraj |
| | | | | | |

Final Submission

1. **Project Report Based on the templates**
2. **Demo Video (3-5 Minutes)**
3. **GitHub/Code Repository Link**