
Hackathon Project Phases

Project Title:

Gemini Pro Financial Decoder

Team Name:

NeoNova

Team Members:

- Kamutam Vaishnavi
 - K Shruthi
 - Aditi A
 - Greeshma Pranathi
 - P Sirinmai
-

Phase - 1: Brainstorming & Ideation

Objective:

- Develop an AI-powered financial analysis tool using **Gemini Pro** to help users analyze and interpret financial statements easily.

Key Points:

1. Problem Statement:

- Many users struggle to interpret financial statements due to complex terminology.
- Manual financial analysis is time-consuming and prone to errors.
- Businesses and investors need AI-powered insights for decision-making.

2. Proposed Solution:

- A Streamlit-based web app that takes financial statements (CSV/PDF) as input.
- Uses Gemini Pro to generate insights on profitability, liquidity, risks, and trends.
- Provides comparative analysis between different financial reports.

3. Target Users:

- Investors and analysts looking for quick financial insights.
- Business owners who want AI-driven financial summaries.
- Students & researchers learning financial statement analysis.

4. Expected Outcome:

- A working AI-powered financial statement decoder that provides real-time analysis and user-friendly insights.
-

Phase - 2: Requirement Analysis

Objective:

- Define the technical and functional requirements for the **Gemini Pro Financial Decoder**.

Key Points:

1. Technical Requirements:

- Programming Language: Python
- Backend: Gemini Pro API
- Frontend: Streamlit Web Framework
- Database: Not required initially (file-based input)

2. Functional Requirements:

- Upload and extract financial data from CSV/PDF.
- Send extracted text to Gemini Pro for analysis.
- Display key financial insights in a structured format.
- Provide comparisons between multiple financial statements.

3. Constraints & Challenges:

- Ensuring accurate data extraction from PDF financial reports.
 - Optimizing Gemini API prompts for meaningful responses.
 - Handling large datasets efficiently in Streamlit.
-

Phase - 3: Project Design

Objective:

- Develop the **architecture and user flow** of the application.

Key Points:

1. System Architecture:

- User uploads financial statement (CSV/PDF).

- Extract text & numerical data from uploaded files.
- Send data to Gemini Pro API for analysis.
- Display AI-generated insights in a structured format.

2. User Flow:

- Step 1: User uploads a financial statement.
- Step 2: Extract text and convert data to structured format.
- Step 3: AI processes data & provides key insights.
- Step 4: Display results in an easy-to-read format.

3. UI/UX Considerations:

- Minimalist, finance-friendly dashboard.
- Filters for selecting key financial ratios.
- Downloadable reports for further use.

Phase - 4: Project Planning

Objective

Break down development tasks for efficient completion.

Sprint	Task	Priority	Duration	Deadline	Assigned To	Dependencies	Expected Outcome
Sprint 1	Environment Setup & API Integration	High	6 hours (Day 1)	End of Day 1	vaishnavi	Gemini API Key, Python, Streamlit setup	API connection working
Sprint 1	Frontend UI Development	Medium	2 hours (Day 1)	End of Day 1	greeshma	API response format finalized	Basic UI with file upload
Sprint 2	Financial Data Extraction & Processing	High	3 hours (Day 2)	Mid-Day 2	Shruthi&sirinmai	API response, UI elements ready	Text extraction from CSV/PDF
Sprint 2	Error Handling & Debugging	High	1.5 hours (Day 2)	Mid-Day 2	aditi	API logs, UI inputs	Improved API stability
Sprint 3	Testing & UI Enhancements	Medium	1.5 hours (Day 2)	Mid-Day 2	Aditi &vaishnavi	API response, UI layout completed	Responsive UI, better user experience
Sprint 3	Final Presentation & Deployment	Low	1 hour (Day 2)	End of Day 2	Entire Team	Working prototype	Demo-ready project

Phase -5: Project Development

Objective:

- Implement core features of the **Gemini Pro Financial Decoder**.

KeyPoints:

- Technology Stack Used:**
 - Frontend:** Streamlit
 - Backend:** Gemini Pro API
 - Programming Language:** Python
- Development Process:**
 - Implement **API authentication** and Gemini API integration.
 - Develop **financial data extraction logic**.
 - Optimize **prompts** for AI-generated insights.
- Challenges & Fixes:**
 - Challenge:** Extracting clean text from PDF files.
 - Fix:** Use pdfplumber for structured extraction.
 - Challenge:** API rate limits.
 - Fix:** Optimize API calls & batch processing.

Phase -6:Functional & Performance Testing

Objective:

- Ensure that the **Gemini Pro Financial Decoder** works as expected.

Test Case ID	Category	Test Scenario	Expected Outcome	Status	Tester
TC-001	Functional Testing	Upload a CSV financial statement	Extracted text should match source data	Passed	Tester 1
TC-002	Functional Testing	Query “Analyze financial health”	AI provides meaningful financial insights	Passed	Tester 2
TC-003	Performance Testing	API response time under 500ms	AI should return results quickly	Needs Optimization	Tester 3
TC-004	Bug Fixes& Improvements	Fixed incorrect data extraction from PDFs	Clean and accurate text extraction	Fixed	Developer
TC-005	Final	Ensure UI is	UI should work on	Failed - UI	Tester 2

	Validation	responsive across devices	mobile & desktop	broken on mobile	
TC-006	Deployment Testing	Host the app using Streamlit Sharing	App should be accessible online	Deployed	DevOps

Final Submission

1. ProjectReportBasedonthetemplates
 2. DemoVideo(3-5Minutes)
 3. GitHub/CodeRepositoryLink
 4. Presentation
-