# Project Title:

**GEMINI PRO FINANCIAL DECODER**

# Team Name: CIPHERMINDS

# Team Members:

* P MANOJ (Team Lead)
* G MITHIL
* N SAIKUMAR
* G RAJESH

# Phase-1: Brainstorming & Ideation

## Objective:

* The Gemini Pro Financial Decoder is designed to provide real-time financial data analysis, predictive insights, and risk management tools, helping users make data-driven decisions in financial markets.
* Gemini Pro Financial Decoder is to enhance financial decision-making by leveraging AI to process complex data, provide real-time insights, predict outcomes, and automate tasks. It aims to improve accuracy, efficiency, and forecasting in financial analysis.

## Key Points:

1. **Problem Statement:** The financial industry faces challenges in processing and analysing vast amounts of complex data, leading to slow decision-making, missed opportunities, and inaccurate predictions. Traditional financial tools often struggle with real-time analysis, integrating diverse data sources, and providing reliable forecasts. There is a need for an advanced solution that can efficiently decode financial data, automate tasks, and deliver accurate, data-driven insights to optimize decision-making and risk management.
2. **Proposed Solution:** The **Gemini Pro Financial Decoder** uses AI to efficiently analyse complex financial data in real-time, integrate multiple data sources, and enhance predictive accuracy. It automates routine tasks, provides actionable insights for faster decision-making, and helps forecast market trends and risks. This solution improves efficiency, accuracy, and overall financial strategy, reducing costs and optimizing decision-making.
3. **Target Users:**

 Financial Analysts – For trend analysis and data insights.

 Investment Firms – To optimize portfolios and predict market movements.

 Banks & Financial Institutions – For real-time analysis and risk management.

1. **Expected Outcome:**

* Better Decision-Making: Faster, data-driven insights.
* Increased Efficiency: Automation of routine task

# Phase-2: Requirement Analysis

## Objective:

* + Define technical and functional requirements for Gemini Pro Financial Decoder

## Key Points:

1. **Technical Requirement:**

* Programming Language: Python
* Backend: Gemini Pro
* Frontend: Streamlit

1. **Functional Requirements:**

 Advanced Reporting: Generate customizable financial reports and dashboards.

 Scenario Analysis: Model and simulate financial outcomes.

 User-Friendly Interface: Simple, intuitive design for easy use.

1. **Constraints & Challenges:**

 Data Quality: Inaccurate or incomplete data can lead to errors.

 Integration: Challenges in connecting with legacy systems.

 Scalability: Performance may drop with large data volumes.

 Security: Risk of data breaches and privacy violations.

# Phase-3: Project Design

## Key Points:

1. **System Architecture Diagram:**

* **User Uploads PDF**: The user uploads a financial statement PDF
* **Extract Data**: Python libraries (PyMuPDF, tabula-py, or camelot-py) extract text and table data from the PDF
* **Process with GEMINI API**: The extracted data is sent to the GEMINI AI model for analysis, which summarizes financial insights, identifies risks, and suggests improvements.
* **Display Results & Chatbot**: The front-end displays the financial summary, risk areas, and improvements, along with a chatbot for user queries.

1. **User Flow:**

* Step 1: User Uploads the data
* Step 2: The backend calls the GEMINI PRO to retrieve the uploaded data
* Step 3: The web app process the data and **displays results**

1. **UI/UX Considerations**

* Minimalist, User-friendly interface for seamless navigation
* Navigation-bar with four tools(home, analysis, compare, chatbot, dynamic dashboard)

# 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  & Summarisation | 🔴 High | 6 hours  (Day 1) | End of Day  1 | Member 1 | Google API Key,  Python(lib), Streamlit setup | Data extraction& Summarisation |
| Sprint 1 | Frontend UI Development | 🟡 Medium | 2 hours  (Day 1) | End of Day  1 | Member 2 | API response format finalized | Basic UI with input fields |
| Sprint 2 | Summarisation & Chat bot | 🔴 High | 3 hours  (Day 2) | Mid-Day 2 | Member 1& 2 | API response, UI elements ready | Search Queries with Chatbot |
| Sprint 2 | Error Handling &  Debugging | 🔴 High | 1.5 hours  (Day 2) | Mid-Day 2 | Member 1&4 | API logs, UI inputs | Improved API stability |
| Sprint 3 | Testing & UI  Enhancements | 🟡 Medium | 1.5 hours  (Day 2) | Mid-Day 2 | Member 2& 3 | 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 |

## Key Points:

1. **Sprint Planning:**

* Sprint 1 – Setup & Integration (Day 1)

(🔴 High Priority) Set up the **environment** & install dependencies.

(🔴 High Priority) Integrate **Google Gemini PRO**

(🟡 Medium Priority) Build a **basic UI with input fields**

* Sprint 2 – Core Features & Debugging (Day 2)

(🔴 High Priority) Implement search & comparison functionalities.

(🔴 High Priority) Debug API issues & handle errors in queries.

* Sprint 3 – Testing, Enhancement & Submission (Day 2)

(🟡 Medium Priority) Test API responses, refine UI, & fix UI bugs.

(🟢 Low Priority) Final **demo preparation & deployment**.

# Phase-5: Project Development

## Objective:

Implement core features of the GEMINI PRO & FINANCIAL DECODER

## Key Points:

## Technology Stack Used:

* Programming Language: Python
* Backend: Gemini Pro
* Frontend: Streamlit

**Development Process:**

* Extracting **API key authentication** and **Gemini API Integration**.
* Develop **Financial Summarization, Chatbot,**

**Comparison.**

* Optimize **Prompts for performance and relevance**.

**Challenges & Fixes:**

* **Challenge:** Extracting table data.

**Fix:** Implementing multilayer extraction

* **Challenge:** Navigating through page

**Fix:** creating **st.sidebar.selectbox()** and separate page functions

# Phase-6: Functional & Performance Testing

## Objective:

* + Ensure that the FINANCIAL DECODER works as expected.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Category** | **Test Scenario** | **Expected Outcome** | **Status** | **Tester** |
| TC-001 | Functional  Testing | Uploaded Financial Infosys statements | The given data is summarized. | ✅ Passed | Tester 1 |
| TC-002 | Functional  Testing | Uploaded two financial statements | The given Financial data is compared | ✅ Passed | Tester 2 |
| TC-003 | Performance  Testing | API response time under  1sec | API should return results quickly. | ⚠ Needs Optimization | Tester 3 |
| TC-004 | Bug Fixes & Improvements | Fixed Multilayer  function | The Data loss is being  reduced | ✅ Fixed | Developer |
| TC-005 | Final Validation | Ensure UI is responsive across devices. | UI should work on mobile & desktop. | ❌ Failed - UI broken on mobile | Tester 2 |

# 

# Final Submission

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