1. INTRODUCTION

1.1 Project Overview

Gemini Pro Financial Decoder is an AI-powered analytical system designed to interpret complex financial data and convert it into clear, actionable insights. The application uses Google's Gemini Pro Financial Decoder model to analyze structured data, reports, spreadsheets, and financial documents. It generates detailed insights such as trend interpretation, ratio analysis, risk evaluation, revenue patterns, and investment recommendations.

- **Scenario 1: Buying a New Motorcycle**
  Arun uploads company financial statements into the system. The Gemini Pro Financial Decoder analyzes profitability, liquidity, and market trends, helping him decide whether the company is a good investment.
- **Scenario 2: Vehicle Maintenance Tips**
  A business owner uses the system to decode quarterly revenue data. The model identifies growth patterns, cash flow issues, and risk areas, enabling more informed strategic planning..
- **Scenario 3: Finding Eco-Friendly Vehicles**
  Vamsi uses the tool to analyze her monthly spending, categorize expenses, and receive suggestions on budgeting and saving strategies, helping her improve financial stability.

1.2 Objectives

The primary purpose of the Gemini Pro Financial Decoder project is to develop an intelligent AI-driven system capable of analyzing complex financial data and producing clear, actionable insights. It aims to assist users such as investors, analysts, business owners, students, and financial planners by offering quick, accurate interpretations of financial statements, reports, and datasets—reducing manual effort and significantly improving decision-making efficiency.

Key Objectives:

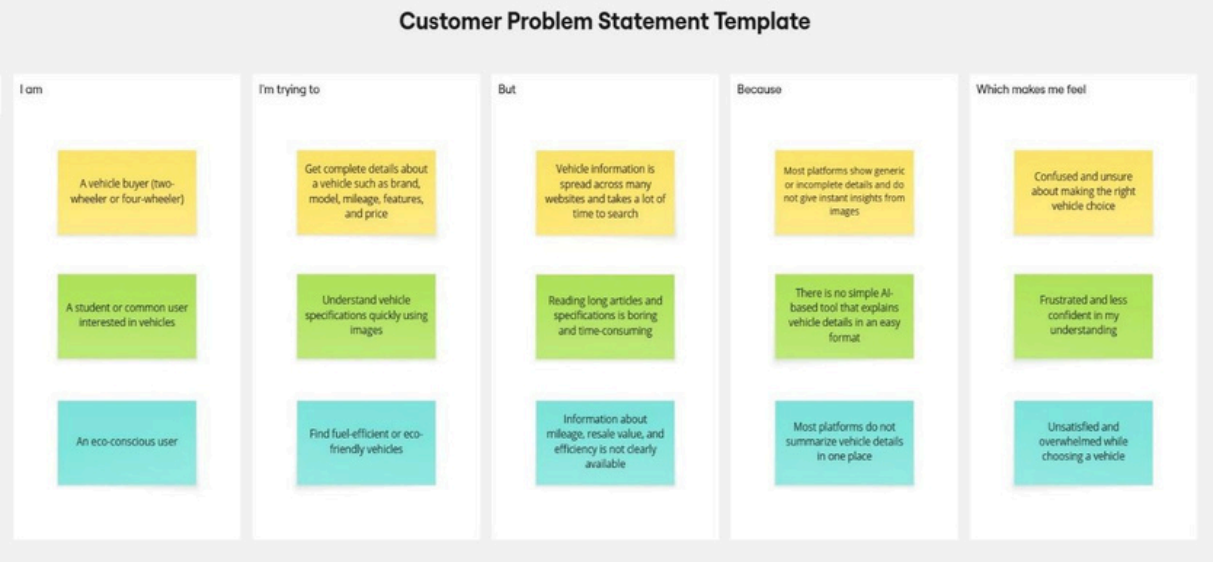
- To analyze financial data using an advanced AI-powered financial decoding model.
- To generate structured insights such as trend analysis, ratio interpretation, revenue patterns, risk evaluation, and investment recommendations.
- To develop a clean and user-friendly financial analysis web application using Streamlit.

2. PROJECT INITIALIZATION AND PLANNING PHASE

2.1 Defining Problem Statement

Individuals, investors, and businesses often struggle to interpret complex financial data spread across multiple reports, spreadsheets, and online sources. Understanding key financial metrics—such as profitability, liquidity, cash flow, market trends, and risk factors—requires technical expertise and considerable time.

Most existing platforms provide raw numerical data but do not offer AI-driven, simplified, and actionable insights. Users must manually read lengthy financial documents, decode ratios, and interpret trends, which is often confusing, time-consuming, and prone to human error.



Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS-1	A vehicle buyer	Understand vehicle details before buying a bike or car	Searching for vehicle information takes too much time	Vehicle details are spread across many websites and need manual comparison	Confused and unsure about choosing right vehicle

PS-2	A student or general user	Identify and understand a vehicle using its image	Readin g ling specific ations and reviews is difficult	Most platforms do not support image-based vehicle analysis	Frustrated and less confident
PS-3	An eco-conscious user	Find fuel-efficiency or eco-friendly vehicles	It is hard to get clear informa tion about mileage and efficien cy	Vehicle informatio n is not summarize d in one place	Dissatisfied and overwhelmed

2.2 Project Proposal (Proposed Solution)

Project Overview	
Objective	The main objective of the Gemini Pro Financial Decoder project is to build an intelligent application that can analyze complex financial data and automatically generate clear, structured insights.
Scope	The project allows users to upload financial documents, reports, or datasets and receive decoded financial insights through an advanced AI model
Problem Statement	

Description	Users often struggle to gather complete and meaningful financial information from a single place. Financial data is scattered across multiple reports, spreadsheets, and online sources, making it difficult to interpret and compare.
Impact	This problem affects users by increasing the time and effort needed to understand financial statements. It makes investment decisions, business evaluations, and financial planning more difficult and less accurate.
Proposed Solution	
Approach	This problem affects users by increasing the time and effort needed to understand financial statements. It makes investment decisions, business evaluations, and financial planning more difficult and less accurate
Key Features	• Upload financial documents or datasets • AI-based financial decoding and insight generation • Clear, structured analytical output • User-friendly Streamlit interface • Secure API key and data handling

Resource Requirements

Resource Type	**Description**	**Specification/Allocation**
Hardware		
Computing Resources	Laptop or Desktop	Standard System
Memory	RAM	Minimum 8 GB RAM
Storage	Disk space	10 – 20 GB
Software		
Frameworks	Python frameworks	Streamlit
Libraries	Additional Python libraries	Google-generativeai, python-dotenv, Pillow

Development Environment	IDE, version control	VS Code, Github
Data		
Data	User-uploaded vehicle images	Small to medium-sized images (PDF, CSV, XLSX)

2.3 Initial Project Planning

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

Thefollowingtable represents the product backlog and sprint-wise planning for the **Gemini Pro Financial Decoder** project.

Sprint	User Story / Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members	Sprint Start Date	Sprint End Date (Planned)
Sprint-1	User Interface Setup	USN-1	As a user, I want a Streamlit-based interface to upload financial documents.	2	High	All Team members	28 January 2026	31 January 2026
Sprint-1	Image Upload Validation	USN-2	As a user, I want the system to validate uploaded file formats (PDF, CSV, XLSX)	1	High	All Team members	28 January 2026	31 January 2026
Sprint-2	AI Model Integration	USN-3	As a user, I want the system to analyze financial data using the Gemini Pro Financial Decoder model.	3	High	All Team members	02 February 2026	09 February 2026
Sprint-2	Prompt Engineering	USN-4	As a user, I want the system to generate structured financial insights such as ratios, trends, and risks.				February 2026	09 February 2026

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members	Sprint Start Date	Sprint End Date (Planned)
Sprint-3	Output Display	USN-5	As a user, I want to clearly view the generated financial insights on the screen.	2	High	All Team members	12 February 2026	18 February 2026
Sprint-3	Deployment	USN-6	As a user, I want the application to be deployed and accessible via a browser.	2	Medium	All Team members	12 February 2026	18 February 2026

3. DATA COLLECTION AND PREPROCESSING PHASE

3.1 Data Collection Plan and Raw Data Sources Identified

Data Collection Plan Template

Section	Description
Project Overview	Gemini Pro Financial Decoder is an AI-powered system that analyzes financial documents and datasets to generate structured insights such as trend analysis, ratio evaluation, risk assessment, and investment recommendations.
Data Collection Plan	Data is collected directly from users in the form of uploaded financial documents (PDFs, CSV files, and Excel sheets) through the Streamlit-based web interface.
Raw Data Sources Identified	The raw data consists of financial reports, statements, and datasets uploaded by users for analysis (e.g., balance sheets, income statements, monthly expenses, sales records).

Raw Data Sources Template

Source Name	Description	Location/URL	Format	Size	Access Permissions
User-uploaded financial documents	Financial statements, reports, or datasets submitted by users (e.g., CSV, PDF, XLSX)	Local system via Streamlit web application	PDF, CSV, XLSX	Small to Medium	Private(User provided)

3.2 Data Quality Report

Data Quality Report Template

Data Source	Data Quality Issue	Severity	Resolution Plan
User-uploaded financial document	Files may contain missing values, incomplete tables, or inconsistent formatting	Moderate	Apply validation checks and notify users to upload clean documents; handle missing values programmatically
User-uploaded financial document	Different file formats (PDF, CSV, XLSX) may require different preprocessing pipelines	Low	Use separate handlers for PDFs, spreadsheets, and CSV files to standardize input
User-uploaded vehicle images	Numerical values may contain currency symbols, commas, or non-numeric characters	High	Clean and normalize numerical values during preprocessing

3.3 Data Preprocessing

Preprocessing Template

In the Gemini Pro Financial Decoder project, financial documents uploaded by users are preprocessed before being analyzed. The files are validated, extracted, cleaned, and structured to ensure high-quality input for the AI model. PDF text extraction, numerical cleaning, and formatting normalization are applied to maintain consistency. These preprocessing steps help the model generate accurate, actionable financial insights.

Section	Description
Data Overview	The data used in this project consists of financial documents (PDF, CSV, XLSX).

	These include balance sheets, income statements, expense records, and financial summaries
Document Validation	Uploaded files are checked to ensure they are in supported formats (PDF, CSV, XLSX) before further processing. Non-financial or unsupported files are rejected.
Document Handling	Text or tabular data is extracted and prepared for analysis.
Prompt Formatting	A structured financial-analysis prompt is generated by combining extracted data with predefined instructions to produce organized insights
Model Input Processing	he cleaned data and formatted prompt are sent to the Gemini Pro Financial Decoder, which internally performs data interpretation
Data Preprocessing Code:	
Loading Data	Financial documents (PDF, CSV, XLSX) are collected directly from users through the Streamlit file uploader.
Input Validation	TThe application ensures a valid financial document is uploaded before processing. File type, size, and basic content structure are checked.
Prompt Creation	The extracted financial data is combined with a predefined financial-analysis prompt to guide the model .
Model Invocation	The cleaned data and structured prompt are sent to the Gemini pro financiaal
Output Handling	The generated financial insights (summaries, ratios, risks, trends) are received from the model and displayed clearly on the Streamlit interface.

4. MODEL DEVELOPMENT PHASE

4.1 Model Selection Report

Model Selection Report

In the Gemini Pro Financial Decoder project, the focus is on selecting a powerful pre-trained financial reasoning and document-analysis model capable of interpreting complex financial data and converting it into clear, actionable insights.
Unlike traditional machine learning workflows that require training models on large financial datasets, this system uses an advanced pre-trained generative AI model. The priority is accurate interpretation of financial documents, robust numerical reasoning, and high-quality summarization.

Model Selection Report:

Model	Description
Gemini Pro Financial Decoder	A powerful pre-trained generative AI model optimized for financial data understanding. It can analyze PDFs, CSVs, and Excel files, extract key metrics, perform numerical reasoning, summarize financial performance, detect trends, and generate actionable insights. Its high accuracy and low latency make it ideal for real-time financial analysis and reporting in the application.

4.2 Initial Model Training Code, Model Validation and Evaluation Report

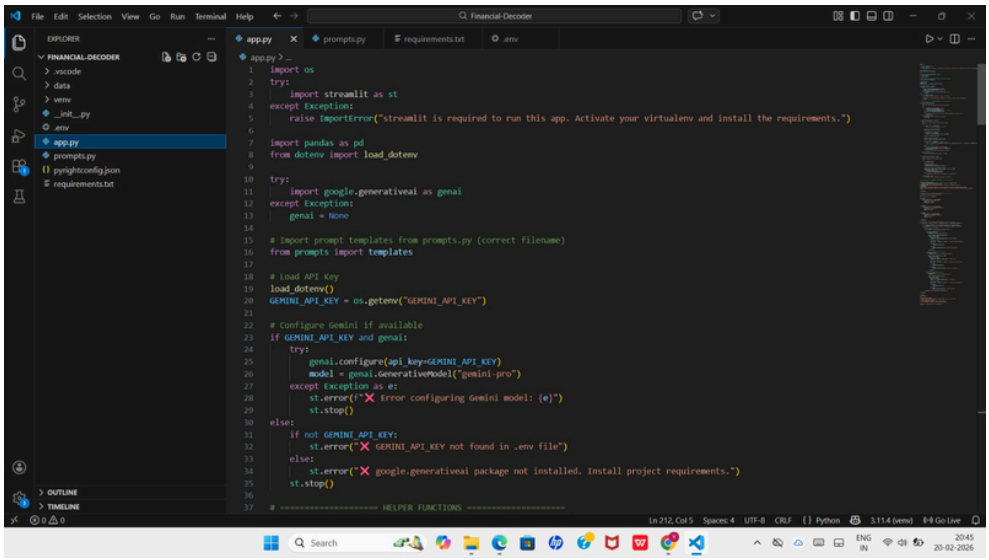
In this project, no custom model training is required. Instead, the application integrates the Gemini Pro Financial Decoder, a pre-trained model capable of financial interpretation.

Initial Model Training Code (5 marks):

Model Selection and Initialization

The Gemini Flash (models/gemini-2.5-flash) model is chosen for the AutoSage application because of its strong multimodal capabilities, allowing it to process both vehicle images and text prompts efficiently. Its fast response time makes it suitable for real-time vehicle detail extraction.

The model is accessed through the Google GenAI SDK, and it is securely initialized using an API key stored in environment variables. Since the model is pre-trained, there is no need for training loops, loss functions, or optimizers. The application directly sends the uploaded vehicle image along with a structured automobile-specific prompt to the model, which then generates organized vehicle insights.



Model Validation and Evaluation Report (5 marks):

Model	Summary	Training and Validation Performance Metrics
Gemini 2.5-flash	Pre-trained multimodal generative AI model optimized for fast and accurate image–text understanding.	The model shows strong performance with fast, low-latency responses and accurate interpretation of vehicle images. It consistently generates clear.

5. MODEL OPTIMIZATION AND TUNING PHASE

5.1 Tuning Documentation

Model Optimization and Tuning Phase

The Model Optimization and Tuning Phase in the AutoSage project focuses on improving the clarity, accuracy, and consistency of AI-generated vehicle information. Since the system uses a pre-trained Gemini Flash model, optimization does not involve retraining. Instead, the tuning process is carried out through well-designed prompts, structured output formatting, and fine-tuning inference parameters to ensure reliable and real-time vehicle analysis.

Hyperparameter Tuning Documentation (8 Marks):

Model	Tuned Hyperparameters
Gemini 2.5 Flash	**Temperature: Adjusted to a balanced value to ensure factual, clear, and stable vehicle information without unnecessary creativity.** **- Top-p: Used to control nucleus sampling so responses stay relevant and focused on vehicle details.** **- Top-k: Restricts token selection to avoid incorrect or unrelated information.** **- Max Output Tokens: Set high enough to generate complete vehicle details including brand, model, mileage, price, and resale value.** **- Response Format: Configured as structured plain text to ensure clean and readable output in the Streamlit interface.**

5.2 Final Model Selection Justification

Final Model	Reasoning
Gemini 2.5 Flash *(models/gemini-2.5-flash)*	The Gemini 2.5 Flash model is chosen because of its fast response time, strong multimodal capabilities, and ability to efficiently process both images and text. It generates accurate and structured vehicle information in real time and integrates seamlessly with the Streamlit interface, making it the ideal model for the AutoSage application

6. RESULTS

6.1 Output Screenshots

Thecomplete execution of the AutoSage application is shown in the images step by step as shown below.

Step 1: Run the app.py code using **streamlit run app.py** and you will get a link in terminal http://localhost:8501 which opens automatically in the browser.

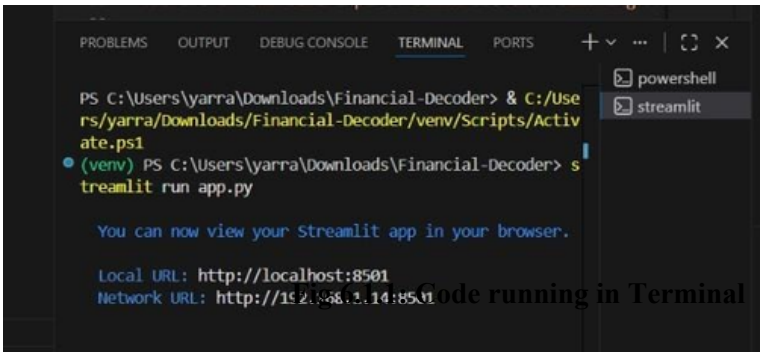
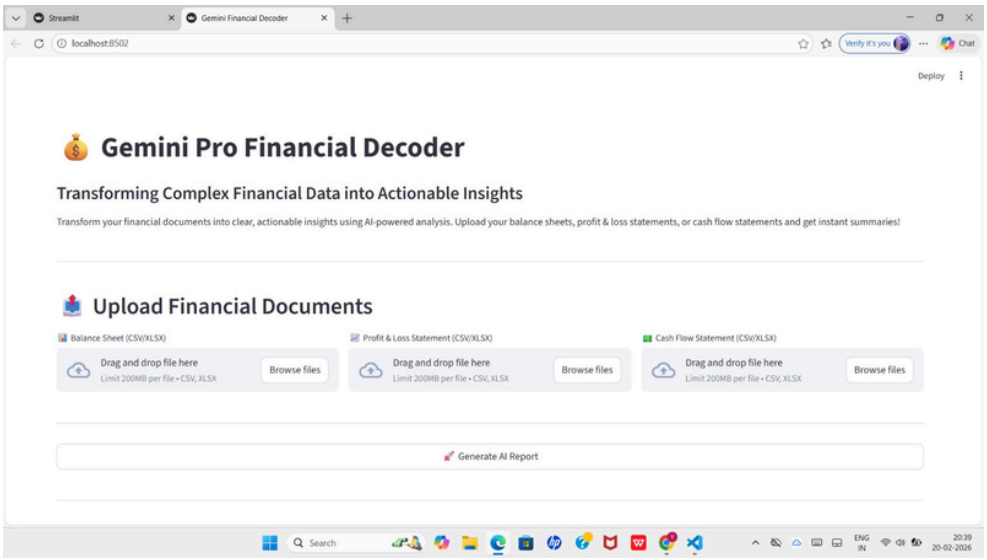
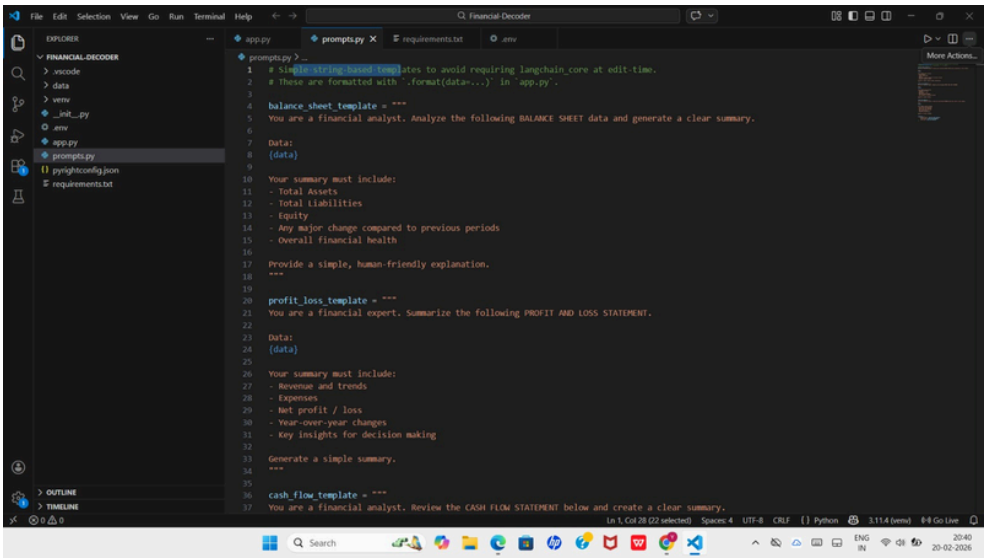


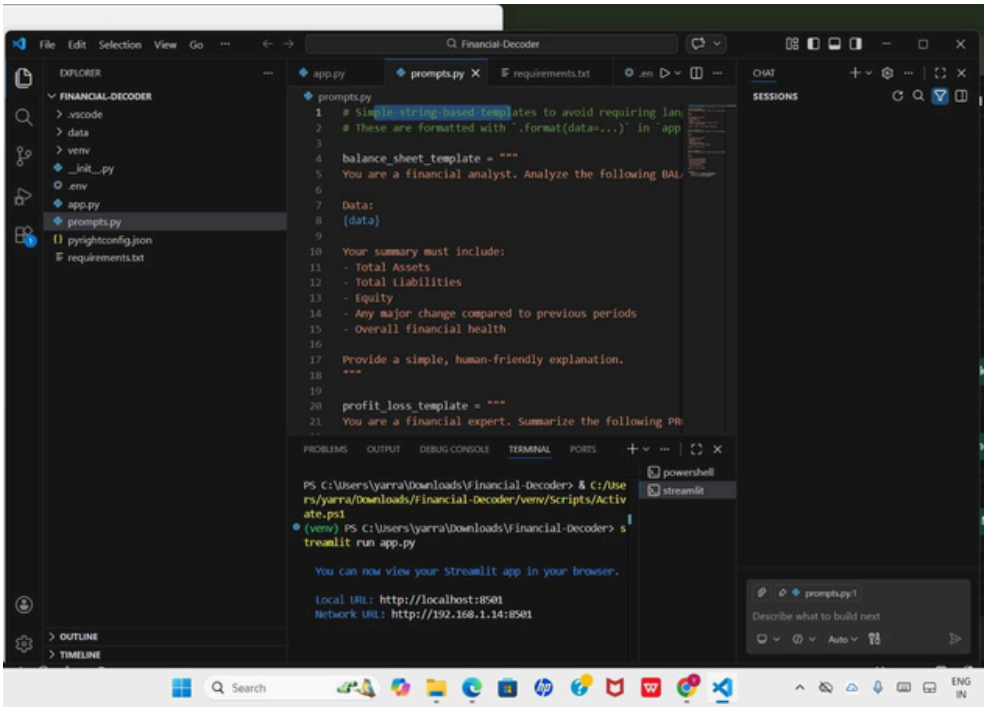
Fig 6.1 Code running in Terminal

Step 2: The link automatically opens to display the application.



Step 3: Click on **Browse Files** optionto predict the vehicle details.


File Edit Selection View Go Run Terminal Help
Financial-Decoder
EXPLORER
FINANCIAL-DECODER
.vscode
data
venv
__init__.py
.env
app.py
prompts.py
pyrightconfig.json
requirements.txt
app.py
prompts.py
requirements.txt
.env
prompts.py > ...
# Simple string-based templates to avoid requiring langchain_core at edit-time.
# These are formatted with `.format(data=...)` in `app.py`.
balance_sheet_template = """
You are a financial analyst. Analyze the following BALANCE SHEET data and generate a clear summary.
Data:
{data}
Your summary must include:
- Total Assets
- Total Liabilities
- Equity
- Any major change compared to previous periods
- Overall financial health
Provide a simple, human-friendly explanation.
"""
profit_loss_template = """
You are a financial expert. Summarize the following PROFIT AND LOSS STATEMENT.
Data:
{data}
Your summary must include:
- Revenue and trends
- Expenses
- Net profit / loss
- Year-over-year changes
- Key insights for decision making
Generate a simple summary.
"""
cash_flow_template = """
You are a financial analyst. Review the CASH FLOW STATEMENT below and create a clear summary.
OUTLINE
TIMELINE
Search
ENG
IN
20:40
20-02-2026


File Edit Selection View Go
Financial-Decoder
EXPLORER
FINANCIAL-DECODER
.vscode
data
venv
__init__.py
.env
app.py
prompts.py
pyrightconfig.json
requirements.txt
app.py
prompts.py
requirements.txt
CHAT
SESSIONS
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
powershell
streamlit
PS C:\Users\yarra\Downloads\Financial-Decoder> & C:/Users/yarra/Downloads/Financial-Decoder/venv/Scripts/Activate.ps1
(venv) PS C:\Users\yarra\Downloads\Financial-Decoder> streamlit run app.py
You can now view your Streamlit app in your browser.
Local URL: http://localhost:8501
Network URL: http://192.168.1.14:8501
prompts.py:1
Describe what to build next
Auto
OUTLINE
TIMELINE
Search
ENG
IN

7. ADVANTAGES & DISADVANTAGES

7.1 Advantages of AutoSage Project

Advantage	Description
Advanced Financial Data Interpretation	Efficiently decodes complex financial datasets, enabling clear and meaningful insights without manual analysis.
Real-Time Decision Support	Provides instant analysis and insights, helping users make quick and informed financial decisions.
high accuracy	Uses powerful AI models to extract patterns, trends, and anomalies with high precision
User-Friendly Output	Reduces the need for manual financial calculations and reporting, saving time and effort.
ImprovesProduct	Transforms raw financial data into structured, easy-to-understand summaries, charts, and recommendations.
Data-Driven Decisions	Helps users make informed vehicle purchase and maintenance decisions based on structured AI-generated insights.
Scalable	Can be applied to budgeting, forecasting, stock analysis, expense tracking, and corporate financial reporting.

Table 7.1: Advantages

7.2 Disadvantages / Limitations

Disadvantage	Description
Dependence on Image Quality	Low-quality or unclear vehicle images may reduce the accuracy of generated information.
Internet Dependency	The application requires an active internet connection to access the Gemini API.
API Usage Limits	Free-tier API usage is limited and may cause temporary restrictions under heavy usage.
No Physical Inspection	The system cannot detect hidden mechanical issues or internal vehicle defects.
Requires API Key Security	Improper handling of API keys may lead to security or quota issues.

Table 7.2: Disadvantages

8. CONCLUSION

The Gemini Pro Financial Decoder project successfully demonstrates how advanced Generative AI can transform complex financial data into clear, actionable insights. By integrating intelligent data interpretation with a user-friendly interface, the system enables users to upload financial documents or datasets and instantly receive structured summaries, trend analysis, and decision-ready insights. This reduces manual effort, minimizes errors, and significantly enhances the speed of financial understanding. Overall, the project showcases the powerful role of AI in simplifying financial analysis, boosting productivity, and supporting smarter, data-driven decision-making across personal, business, and enterprise finance domains.

9. FUTURE SCOPE

The **AutoSage** project can be further enhanced by integrating additional features and technologies. In the future, the application can support video-based vehicle analysis to provide more accurate insights. Integration with official vehicle databases can improve data accuracy and reliability. The system can also be extended to include a mobile application for wider accessibility. Additionally, multi-language support and cloud deployment can be implemented to improve scalability and reach a larger user base.

10. APPENDIX

10.1 Source Code

The complete source code for the Gemini pro financial decoder: transforming complex data into acceptable insights project is implemented using Python, Streamlit, and the Google Gemini API. The code includes modules for data upload, AI model integration, prompt handling, and output display.

10.2 GitHub & Project Demo Link

The source code and project demonstration video are hosted on GitHub. The repository contains all required files, including the application code, documentation, and setup instructions.

- **GitHub Repository:** https://github.com/gowrabathinivamsi/gemini_financial_decoder.git