



ALLIANCE
UNIVERSITY

*Private University established in Karnataka State by Act No.34 of year 2010
Recognized by the University Grants Commission (UGC), New Delhi*

Generative AI Project Idea Proposal Documentation

Bachelor of Computer Applications

SEMESTER – IV

Generative AI & Agentic AI Winter Program

Submitted By:

Anuska Ghosh - 2411021240012

Likitha S - 2411021240067

Faculty Signature

HOD Signature

Department of Computer Applications

Alliance University

Chandapura – Anekal Main Road, Anekal

Bengaluru – 562 106

December 2025

Project Idea Proposal Documentation

1. Project Title

Generative AI–Based Data Analysis and Report Generation Assistant

2. Introduction

Data analysts and data science students often receive raw datasets that require cleaning, analysis, and report creation. These tasks are repetitive, time-consuming, and require writing similar code multiple times. Beginners especially struggle with deciding the correct steps and presenting insights clearly.

This project proposes a cost-effective, Generative AI-based assistant that automates basic data cleaning, analysis, and report generation using agentic design patterns. The system helps users save time while following a structured and understandable workflow.

3. Problem Statement

- Manual data cleaning and report generation take significant time.
- Repetitive tasks reduce productivity and learning efficiency.
- Junior analysts face difficulty in structuring analysis and writing reports.
- Existing tools often require advanced expertise or paid platforms.

4. Objectives of the Project

- To automate routine data analysis tasks using Generative AI.
- To assist beginners in performing structured data analysis.
- To demonstrate the use of **Planner, Tool-Based, and Reflection patterns**.
- To build a cost-effective solution using open-source technologies.
- To create a simple and practical product with real-world relevance.

5. Proposed Solution

The proposed system is a **web-based application** where users can upload a dataset and request basic analysis and report generation.

The system uses **Generative AI as an orchestrator** to:

- Plan tasks
- Execute data tools
- Generate human-readable reports
- Improve outputs based on user feedback

6. Design Patterns Used

A. Planner Pattern

- Breaks the user request into logical steps such as:
 - Data cleaning
 - Statistical summary
 - Visualization
 - Report writing
- Ensures tasks are executed in a structured order.

B. Tool-Based Pattern

- Uses data processing tools for actual execution:
 - Pandas for data cleaning
 - NumPy for calculations
 - Matplotlib/Seaborn for visualization
- Generative AI selects and orchestrates tool usage.

C. Reflection Pattern

- Collects user feedback after report generation.
- Refines explanations and presentation based on feedback.
- Improves output quality without retraining models.

7. Step-by-Step Workflow

Step 1: User uploads a dataset (CSV/Excel).

Step 2: User provides a simple request (e.g., “Clean data and generate report”).

Step 3: Planner pattern breaks the task into steps.

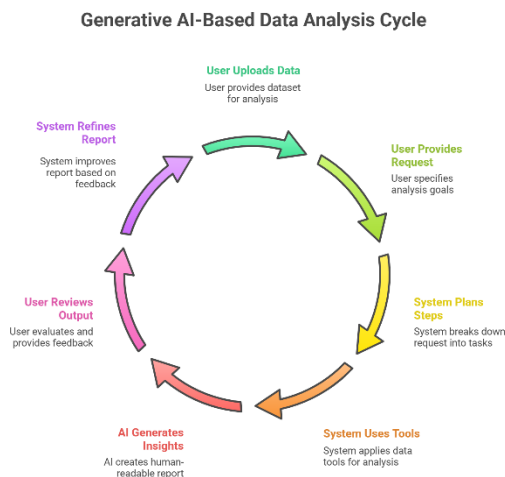
Step 4: Tool-based pattern executes each step using data libraries.

Step 5: Generative AI generates summaries and explanations.

Step 6: User reviews the report and provides feedback.

Step 7: Reflection pattern refines the output.

The following diagram visually represents the complete working cycle of the proposed Generative AI-based data analysis system, from user input to feedback-based refinement.



The workflow starts when the user uploads a dataset and provides an analysis request. The system then plans the required steps, uses data processing tools to perform analysis, generates insights using Generative AI, and presents the results to the user. Based on user feedback, the system refines the report, completing one full analysis cycle.

8. Technology Stack (Cost-Effective & Open Source)

Backend

- Python

Generative AI

- Gemini API

Data Processing

- Pandas
- NumPy

Data Visualization

- Matplotlib
- Seaborn

Frontend

- Streamlit (open-source, lightweight)

Version Control

- GitHub

9. Cost Effectiveness

- Uses open-source libraries only.
- No paid software or proprietary platforms.
- Gemini API provides affordable and scalable Generative AI access.
- Streamlit enables rapid development without infrastructure cost.

10. Scope of the Project (MVP)

- Supports one dataset at a time.
- Performs:
 - Data cleaning
 - Basic statistical analysis
 - Standard visualizations
 - Textual report generation
- Does not include advanced machine learning or prediction models.

11. Expected Deliverables

- A working web-based prototype.
- Upload and processing of datasets.
- Cleaned data output.
- Visual charts and insights.
- Generated analytical report.
- Feedback-based refined output.

12. Real-World Applications

- Academic data analysis projects.
- Beginner data analyst workflows.
- Business reporting for small datasets.
- Learning tool for data science students.

13. Conclusion

This project demonstrates a **practical and cost-effective application of Generative AI** using agentic design patterns. By combining planning, tool execution, and reflection, the system simplifies data analysis workflows and improves productivity. The use of open-source tools and the Gemini API ensures affordability, scalability, and ease of implementation.