AnalytixAI

Automated Business Analytics with LLM and LangChain

Capstone Project Report

END-SEMESTER EVALUATION

Submitted by:

(102003130) Arpit Sagar COE

(102003171) Manpreet Singh COE

(102003184) Madhvan Jindal COE

(102017180) Jitesh Garg CSE

BE Fourth Year

CPG No: 17

Under the Mentorship of

Dr. Sharad Saxena

(Associate Professor)

Computer Science and Engineering Department

Thapar Institute of Engineering and Technology, Patiala

May - 2024

ABSTRACT

In today's fast-paced business environment, the demand for real-time, data-driven insights is ever-increasing. Businesses are constantly seeking ways to automate their analytics processes to gain a competitive edge and make informed decisions swiftly. This growing need for automation in business analytics has paved the way for innovative solutions that integrate advanced technologies like Machine Learning (LLMs) and Data Analytics. These solutions aim to streamline the analytics workflow, reduce manual efforts, and deliver actionable insights at scale.

The project, "AnalytixAI: Automated Business Analytics Dashboard using LLM and LangChain," is a pioneering effort in this domain. It focuses on developing a fully automated business analytics dashboard that leverages LLMs and Data Analytics to generate visual charts and graphs based on customer prompts. This dashboard aims to provide businesses with a comprehensive view of their data, enabling them to identify trends, patterns, and anomalies with ease. Moreover, the inclusion of automated text insights enhances the dashboard's utility by providing detailed explanations and actionable recommendations.

The successful implementation and deployment of this project will be detailed in the report, showcasing how the dashboard has transformed the way businesses analyze their data. It will highlight the efficiency gains, cost savings, and strategic advantages that businesses can achieve by adopting automated analytics. Additionally, the report will discuss the technical challenges faced during the implementation process and the strategies employed to overcome them. Overall, the report will serve as a comprehensive guide for businesses looking to harness the power of automated analytics in driving their success.

DECLARATION

We hereby declare that the design principles and working demonstration of the project entitled "AnalytixAI: Automated Business Analytics with LLM and LangChain" is an authentic record of our own work carried out in the Computer Science and Engineering Department, TIET, Patiala, under the guidance of Dr Sharad Saxena during 8th Semester (2024).

Date: 29/05/2024

Roll No.	Name	Signature
102003130	Arpit Sagar	
102003171	Manpreet Singh	
102003184	Madhvan Jindal	
102017180	Jitesh Garg	

Counter Signed By:

Faculty Mentor:

Dr. Sharad Saxena
Associate Professor
CSED, TIET, Patiala

ACKNOWLEDGEMENT

We extend our sincere gratitude to Dr. Sharad Saxena, our mentor, whose guidance and technical expertise has been invaluable throughout our project. His support has been instrumental in our journey, and we are grateful to have such an exceptional mentor.

We are also thankful to Dr Shalini Batra, Head, Computer Science and Engineering Department, entire faculty and staff of Computer Science and Engineering Department, and also our friends who devoted their valuable time and helped us in all possible ways towards successful completion of this project. We thank all those who have contributed either directly or indirectly towards this project.

Finally, we would like to express our gratitude to our families for their unwavering love and support. Their constant encouragement and sacrifices have been a source of strength for us, and we deeply appreciate their desire for our success.

Date:29/05/2024

Roll No.	Name	Signature
102003130	Arpit Sagar	
102003171	Manpreet Singh	
102003184	Madhvan Jindal	
102017180	Jitesh Garg	

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

LLM	Large Language Models
DFD	Data Flow Diagram
ER	Entity Relation
AI	Artificial Intelligence
ML	Machine Learning
RTA	Real Time Analytics
DF	Data Frame
PD	Pandas Library

TABLE OF CONTENTS

INTRODUCTION

1.1 Project Overview

Business analytics is the process of utilizing data analysis and statistical methods to derive actionable insights for informed decision-making within an organization. As businesses strive for efficiency and effectiveness in their operations, the need for automation in business analytics becomes paramount. Automation involves the use of technologies such as Machine Learning (LLMs) and Data Analytics to streamline and expedite the analytics process, reducing manual efforts and enabling real-time insights.

Machine Learning models, particularly LLMs, are instrumental in automating business analytics. These models can analyze vast amounts of data and extract meaningful patterns and trends without human intervention. LangChain, on the other hand, facilitates the integration of LLMs with Data Analytics, providing a comprehensive solution for advanced analytics.

The project, "AnalytixAI: Automated Business Analytics Dashboard using LLM and LangChain," is an effort to develop a highly automated business analytics dashboard. This dashboard will leverage LLMs, LangChain and Data Analytics to generate visual charts and graphs, such as bar charts and pie charts, based on customer prompts. It aims to provide insights into various metrics, including average spending among departments, to facilitate data-driven decision-making. Additionally, the dashboard will incorporate automated text insights about the graphs, enhancing its informativeness. The project's ultimate goal is to create an interactive dashboard that simplifies data interpretation and analysis, thereby exploring the real-world business potential of automated analytics.

1.2 Need Analysis

Traditional data analytics methods often require significant manual effort, are time-consuming, and may not be able to keep up with the rapid pace at which data is generated.

Problems with Existing Data Analytics Solutions:

1. **Manual Processes and Time-Consuming Tasks:**

- Data Preparation: Current analytics processes often involve extensive manual data preparation, cleaning, and formatting, which can be labor-intensive and prone to human error.

- Report Generation: Generating reports and visualizations manually takes considerable time, delaying critical decision-making processes.

2. **Limited Real Time Insights:**

- Lag in Analysis: Traditional analytics tools may not support real-time data processing, resulting in outdated insights that do not reflect the current state of the business.

- Static Reports: Many existing solutions provide static reports that lack interactivity and fail to offer deep, real-time insights.

3. **Inability to Handle Large Datasets:**

- Scalability Issues: As the volume of data grows, traditional analytics systems struggle with scalability, impacting their performance and accuracy.

- Data Overload: Organizations often face challenges in managing and analyzing large datasets effectively, leading to missed opportunities and insights.

1.3 Research Gaps

1. Scalability and Performance Optimization

- Investigating advanced optimization techniques and scalable architectures to enhance the system's ability to handle increasing data volumes and concurrent user queries.
- Research should explore distributed computing, parallel processing, and cloud-based solutions to ensure the system remains efficient and responsive under high load conditions.

2. Advanced Textual Insight Generation

- Developing more sophisticated natural language processing (NLP) models to improve the accuracy and contextual relevance of automated textual insights.
- Future research should aim at enhancing the algorithms that interpret data trends and generate descriptive insights, ensuring they are precise, meaningful, and easily understandable by users.

3. Integration of Predictive Analytics

- Incorporating predictive analytics capabilities to extend the system's functionality beyond descriptive analytics.
- Research should focus on integrating machine learning models that can predict future trends and outcomes based on historical data, providing users with actionable foresight.

4. Security and Privacy Measures

- Strengthening data security and privacy protocols to protect sensitive business information.
- Research should investigate advanced encryption techniques, access control mechanisms, and compliance with data protection regulations to ensure robust security in the system.

1.4 Problem Definition and Scope

Problem Definition:

In the rapidly evolving landscape of data-driven decision-making, businesses face significant challenges in managing and analyzing vast amounts of data efficiently. Traditional data analytics processes are often labor-intensive, time-consuming, and prone to human error, hindering the ability to derive timely and actionable insights. The lack of automation in data analytics further exacerbates these issues, resulting in delayed decision-making and suboptimal resource allocation. The existing systems struggle to process large datasets quickly, generate accurate visual representations, and provide meaningful textual insights, thereby limiting their effectiveness and scalability. Addressing these problems is crucial for enhancing operational efficiency, improving decision accuracy, and maintaining competitive advantage in the business world.

Scope:

The scope of addressing the challenges in data analytics and its automation is extensive and multifaceted, encompassing technological, operational, and strategic dimensions. Technologically, it involves developing advanced algorithms and leveraging state-of-the-art machine learning models, such as LLMs, to automate data processing, visualization, and insight generation. This includes creating robust, scalable systems capable of handling large datasets and multiple concurrent user queries without compromising performance. Operationally, the scope extends to integrating these automated solutions within existing business workflows to streamline operations and enhance productivity. Strategically, the challenge includes ensuring that the automated analytics system is adaptable across various industries, providing tailored solutions that meet the specific needs of different business sectors. By addressing these areas, businesses can significantly improve their data-driven decision-making processes, ultimately leading to better resource allocation, increased efficiency, and a stronger competitive position in the market.

1.5 Assumptions and Constraints

1.5.1 Assumptions

S. No.	Assumptions
1.	The datasets provided by users are complete, accurate, and well-structured, allowing the system to generate meaningful insights.
2.	Users are assumed to have a basic understanding of data analytics concepts and how to interact with the dashboard to submit queries and interpret results.
3.	A reliable internet connection is assumed for seamless interaction with the cloud-based components of the system, ensuring smooth data uploads, query processing, and result retrieval.
4.	It is assumed that the system will receive regular updates and maintenance to address any emerging issues, improve performance, and integrate new features, ensuring ongoing reliability and relevance.
5.	The project assumes that the system can process and respond to user queries within a reasonable timeframe, ensuring that users receive timely insights without significant delays.

1.5.2 Constraints

S. No.	Constraints
1.	The system's effectiveness is heavily reliant on the quality of the input data. Inaccurate, incomplete, or poorly structured data can lead to misleading insights and visualizations.
2.	The system must maintain high processing speeds and performance levels to handle large datasets and multiple concurrent user queries efficiently.
3.	While designed to be scalable, there are practical limits to the system's ability to handle extremely large datasets or an excessive number of simultaneous queries without performance optimization and infrastructure scaling.
4.	The system requires significant computational resources and storage capacity.
5.	Ensuring data security and privacy is a critical constraint. The system must adhere to stringent security protocols and comply with data privacy regulations.

1.6 Standards

1. Data Automation Standards:

The project adheres to established data automation standards to ensure consistency, reliability, and efficiency in data handling. This includes the use of standardized protocols for data ingestion, transformation, and storage. By following these standards, the system ensures that data flows seamlessly from input to analysis, reducing the likelihood of errors and improving overall data quality. Automation standards also facilitate interoperability between different systems and tools, enhancing the flexibility and scalability of the project.

2.Data Processing Standards:

The project implements rigorous data processing standards to maintain accuracy and integrity throughout the analytical process. These standards encompass best practices for data cleaning, normalization, and validation. Adhering to data processing standards ensures that the data used for analysis is accurate, complete, and free from inconsistencies, leading to more reliable and meaningful insights. It also involves compliance with performance benchmarks to ensure efficient processing and quick response times, even with large datasets.

3.Technology and Security Standards:

The project follows industry-standard security protocols and best practices to protect data integrity and privacy. This includes the use of encryption for data at rest and in transit, secure authentication and authorization mechanisms, and regular security audits. The project follows established software development standards to ensure robust, maintainable, and scalable code.

1.7 Approved Objectives

To develop a sophisticated dashboard that revolutionizes the way businesses analyze and interpret their data.

By leveraging Machine Learning (LLMs) and Data Analytics, the dashboard will provide automated visual charts and graph generation, data representation and data-driven insights, tailored to customer prompts.

This project aims to create a highly informative and interactive dashboard that empowers businesses to make informed decisions and drive their success through automated analytics.

1.8 Methodology

The methodology can be broken down into several key steps, each leveraging advanced technologies to streamline the process.

1. User Query and Data Access

The process begins when a user inputs a query related to data representation, such as requesting a specific type of chart or graph. The user also uploads the relevant dataset that needs to be analyzed.

2. Data Sampling

Once the dataset is uploaded, it undergoes a unique sampling process. This step ensures that the dataset is refined to obtain unique rows, which helps in reducing redundancy and improving the quality of the data that will be analyzed.

3. LangChain Document Retrieval

This tool extracts valid instructions from a vector database pertinent to the user query. By retrieving these specific instructions, LangChain ensures that the subsequent steps are based on relevant and precise information.

4. Input to LLM Model

The retrieved code instructions, along with the processed dataset, are then fed into the (Large Language Model) in the backend. This model is specifically designed to interpret the inputs and generate appropriate responses.

5. Generation of Prompts and Code Functions

The inputs trigger the generation of relevant prompts that consist of code functions. These functions are tailored to execute the required data visualizations and analyses as per the user query.

6. Execution of Code Functions

The generated code functions are executed, resulting in the creation of visual chart representations. This could include bar charts, pie charts, line graphs, or other forms of visual data representation as requested by the user.

7. Extraction of Textual Insights

In addition to visual representations, a secondary prompt (Prompt2) is used to extract textual insights related to the dataset and the depicted graph. This involves analyzing the results shown in the chart and generating a narrative that explains the key findings and insights.

8. Display on Dashboard

Finally, the visual charts and textual insights are displayed on the AnalytixAI dashboard. This interactive dashboard allows users to view the results of their query in a comprehensive and easy-to-understand format. The dashboard not only shows the visual representation of the data but also provides detailed analytics and insights, enabling data-driven decision-making.

1.9 Project Outcomes and Deliverables

- The final outcome of this project is the successful implementation of Automated Business Analytics Dashboard which is user friendly and interactive.
- Real time data processing to handle multiple requests and large datasets efficiently.
- Automated data visualization with the inclusion of various chart types to provide comprehensive options.
- Integration of advanced algorithms to generate accurate and relevant textual insights accompanying the visual data representations.

1.10 Novelty of Work

Unlike traditional analytics systems that rely heavily on manual data processing and static reporting, this project leverages advanced machine learning models and LangChain technology to automate and streamline the entire analytics process. Key innovations include:

1. Automated Visual and Textual Insights:

The system uniquely combines visual chart generation with automated textual insights, providing a comprehensive and user-friendly analysis experience. This dual approach enhances understanding and decision-making by presenting data in both graphical and narrative forms.

2. Real Time, Scalable Data Processing:

The project introduces a robust architecture capable of real-time data processing and handling large datasets and multiple concurrent user queries. This scalability ensures the system remains efficient and responsive, addressing a common limitation in existing solutions.

3.Interactive and Intuitive User Interface:

The dashboard's design prioritizes user interaction and ease of use, allowing users to easily upload datasets, generate queries, and interpret results. This focus on user experience sets it apart from more complex, less accessible analytics tools.

4. Integration with Advanced Technologies:

By incorporating LangChain for document retrieval and LLM for generating relevant code prompts, the system provides accurate and contextually relevant analytics. This integration of cutting-edge technologies enhances the depth and accuracy of the insights generated.

2.1 Literature Survey

2.1.1 Theory Associated with Problem Area

Business analytics is essential for contemporary organizations, enabling them to use data analysis and statistical methods to generate actionable insights for informed decision-making. Traditionally, this process has been time-consuming and labor-intensive, requiring significant manual effort to analyze and interpret large datasets. As businesses increasingly strive for operational efficiency and effectiveness, the demand for automation in business analytics has grown. Automation involves employing advanced technologies like Machine Learning (ML) and Data Analytics to expedite the analytics process, thereby reducing manual effort and enabling real-time insights.

By using algorithms that learn from data, LLMs continuously improve their performance over time. LangChain enhances this capability by integrating LLMs with data analytics, creating a cohesive solution for advanced analytics. This integration allows for the seamless extraction of meaningful insights from both structured and unstructured data, providing a comprehensive approach to business analytics.

This project aims to harness these technological advancements to develop a highly automated business analytics dashboard. This dashboard leverages the capabilities of LLMs and LangChain to generate visual charts and graphs, such as bar charts and pie charts, based on customer prompts. It aims to provide insights into various metrics, including average spending among departments, to facilitate data-driven decision-making. Additionally, the dashboard incorporates automated text insights that offer detailed explanations of the visualized data, making it more accessible and understandable for users.

2.1.2 Existing System and Solutions

1. Microsoft Excel and SQL Databases

Excel and SQL databases are foundational tools for data storage, manipulation, and analysis, but they require significant manual effort for data preparation and are not well-suited for real-time insights or large-scale data handling.

Drawbacks*:*

- **Manual Effort:** Significant manual effort is required for data extraction, transformation, and loading (ETL).
- **Time-Consuming:** Analysts spend considerable time writing SQL queries, cleaning data, and manually creating reports.
- **Scalability Issues:** Managing and analysing large datasets in Excel can be cumbersome and prone to errors.
- **Limited Real-Time Insights:** Real-time data integration and analysis are challenging without advanced configurations.

Proposed System:

- **Automated Text Insights**: Incorporating automated text insights that provide detailed explanations of visualized data, enhancing the informativeness and usability of the dashboard.
- **Simplified Analytics**: Leveraging LLMs to automatically analyze data and generate insights, reducing the need for specialized knowledge and manual interpretation.

2. Business Intelligence Platforms

BI platforms like Tableau, Power BI, and QlikView provide powerful data visualization and reporting tools, yet they demand technical expertise for setup and maintenance, and often result in static, rather than real-time, insights.

Drawbacks:

- **Complex Setup:** Setting up and maintaining BI platforms requires deep knowledge of data sources, data modeling, and platform features.
- **Technical Expertise:** Users need a certain level of technical expertise to create and interpret reports.
- **Fragmented Solutions:** Often, separate systems are needed for data processing, analysis, and visualization, leading to integration challenges.
- **Static Insights:** Generated insights are often static, requiring continuous updates as new data becomes available.

Proposed System:

- **Automated Model Integration**: Seamlessly integrating machine learning models into the dashboard, allowing for automated data analysis without requiring extensive technical expertise.

2.1.3 Research Findings for Existing Literature

S.No.	Roll Number	Name	Paper Title	Tools/ Technology	Findings	Citation
1	102003130	Arpit Sagar	Learning Analytics in the Era of Large Language Models	Learning Analytics (LA) Large Language Models (LLMs) Natural Language Processing (NLP)	Learning analytics face challenges with design, and interpretability. Future research should enhance model generalizability, address ethical concerns, and involve stakeholders in development and evaluation.	Mazzullo, E., Bulut [1]
2	102003130	Arpit Sagar	AgentBoard: An Analytical Evaluation Board of Multi-turn LLM Agents	Large Language Models (LLMs), AgentBoard benchmark, open-source evaluation framework, fine-grained progress rate metric, and interactive visualization toolkit.	AgentBoard enhances the evaluation of LLM agents by offering a comprehensive benchmark and toolkit that captures incremental progress and facilitates detailed, interactive analysis.	Ma, C., Zhang, J., Zhu, Z., Yang [2]
3	102003171	Manpreet Singh	Automating Research Synthesis with Domain-Specific Large Language Model Fine-Tuning	Fine-tuned Large Language Models (LLMs) Open-sourced LLMs	Fine-tuned Large Language Models (LLMs) automate Systematic Literature Reviews	Susnjak, T., Hwang [3]

				PRISMA-conforming SLR replication	(SLRs) effectively, maintaining high factual accuracy and proposing solutions for mitigating LLM hallucination.	
4	102003171	Manpreet Singh	Generative AI Enabled Conversational Chatbot for Drilling and Production Analytics	Large Language Models (LLMs) Machine Learning Models (ML) Natural Language Processing (NLP) Cloud Hosting	The paper proposes workflow integrating large language models and machine learning for an AI-enabled chatbot to analyze rig and production data, offering diagnostic insights with transformative potential in the energy industry.	Ajay Singh [4]
5	102003184	Madhvan Jindal	Conversational AI Threads for Visualizing Multidimensional Datasets	Large Language Models (LLMs) AI Threads (Analytic Chatbot) Conversational Interfaces Visualization Tools	Generative LLMs hold promise for data analysis, inspiring the creation of AI Threads, an analytic chatbot that improves efficiency with proactive conversational context management. Usability was assessed	Hong, M., & Crisan [5]

					through crowdsourced studies and expert interviews, highlighting future research challenges.	
6	102003184	Madhvan Jindal	Real-Time Analytics: Concepts, Architectures, and ML/AI Considerations	Real-time Analytics (RTA) Complex Event Processing (CEP) Machine Learning (ML)	The paper reviews real-time analytics solutions, covering infrastructure, processing, and analytics platforms, and explores the integration of machine learning and AI. Case studies in finance and health research.	W. Chen, Z. Milosevic [6]
7	102017180	Jitesh Garg	A Preliminary Roadmap for LLMs as Assistants in Exploring, Analyzing, and Visualizing Knowledge Graphs	Large Language Models (LLMs) Knowledge Graphs (KGs) Chat-based Widgets Data Visualization Tools	A mixed-methods study explores how Large Language Models assist users in visual exploration and analysis of Knowledge Graphs (KGs). Participants desire LLMs to facilitate data retrieval, identify relationships, and create on-demand visualizations from KGs,	Li, H., Appleby [7]

					preferably through chat-based widgets integrated into regular analysis workflows. Concerns include maintaining semantic intent, hallucinating false data, and engineering perfect prompts.	
8	102017180	Jitesh Garg	A Prescriptive Learning Analytics Framework: Beyond Predictive Modelling and onto Explainable AI with Prescriptive Analytics and ChatGPT	Machine Learning Models Large Language Models (LLMs) Predictive Analytics Prescriptive Analytics	Recent Learning Analytics research focuses on predictive analytics but lacks interpretability and prescriptive analytics. AI enables transparent predictions and tailored advice. A novel framework integrates transparent machine learning, prescriptive analytics, and large language models, showcasing ChatGPT's potential for human-readable feedback.	Susnjak, T. [8]

2.1.4 Problem Identified

Problems in various types of existing tools and systems of data analytics:

1.Manual Effort in Traditional Tools

Microsoft Excel and SQL Databases

- Extensive manual data extraction and transformation.
- Time-consuming data preparation and reporting processes.
- Inadequate handling of large datasets and real-time insights.

2. Complexity of Business Intelligence Platforms

Setup and Maintenance

- Requires deep technical knowledge for proper configuration.
- Ongoing maintenance and updates are needed for accurate insights.

User Accessibility

- Limited usability for non-technical users.
- Insights often remain static and require continuous updates.

4. Integration Challenges with Machine Learning Platforms

Complex Integration

- Significant technical expertise needed to integrate ML models into workflows.
- High learning curve for non-technical users to utilize these platforms.

Ongoing Oversight

- Continuous monitoring and refinement of models required for accuracy.
- Barrier to widespread adoption due to complexity and maintenance needs.

2.1.5 Survey of Tools and Technologies Used

1. **Large Language Models (LLM):**

   Evaluation of various LLMs, such as GPT-4 and BERT, to determine their efficacy in generating accurate code prompts and textual insights. Assessment criteria include model accuracy, response time, and ease of integration with other tools.

2. **LangChain:**

   Examination of LangChain's document retrieval capabilities, particularly its ability to fetch relevant code instructions quickly and accurately from a vector database. The survey includes a comparison with other document retrieval systems to validate performance and reliability.

3. **Python:**

   Analysis of Python as the primary programming language for the project. The survey covers the suitability of Python for backend development, its library ecosystem (e.g., Scikit-Learn, Pandas, NumPy), and its capability to support machine learning and data processing tasks.

4. **Code Interpreter:**

   Investigation of different code interpreters to evaluate their effectiveness in executing generated code prompts. The survey includes performance benchmarks, error handling capabilities, and integration ease with the project's backend.

5. **Machine Learning Libraries:**

   Assessment of machine learning libraries such as Scikit-Learn, TensorFlow, and PyTorch. The survey examines their compatibility with the project requirements, ease of use, and performance in model training and deployment.

2.2 Software Requirement Specification

2.2.1 Introduction

2.2.1.1 Purpose

The purpose of the AnalytixAI project is to develop an automated business analytics

dashboard that leverages advanced machine learning models, specifically Large Language Models (LLMs), in conjunction with data analytics technologies like LangChain. This dashboard aims to provide real-time, data-driven insights and generate visual charts based on user prompts, facilitating informed decision-making within an organization. The system will cater to various business scenarios, such as analyzing average spending among departments, and will offer automated textual insights to enhance the interpretability of the visual data. This project seeks to explore and demonstrate the real-world potential of automated analytics in improving business efficiency and effectiveness by reducing manual efforts and enabling rapid, accurate analysis.

2.2.1.2 Intended Audience and Reading Suggestions

Intended Audience:

The intended audience for a report on AnalytixAI Automated Dashboard System includes:

- **Business Analysts**:
    - Primary users who leverage the automated analytics dashboard to gain insights into business performance, identify trends, and make data-driven decisions to optimize operations and strategies.
- **Data Monitoring Administrators**:
    - Professionals responsible for overseeing data integrity, ensuring the smooth operation of the data pipeline, and monitoring the performance and reliability of the analytics system.
- **Data Scientists and Analysts**:
    - Individuals who can use the system to automate repetitive analysis tasks, enabling them to focus on more complex data modeling and interpretation work, thereby enhancing productivity and analytical depth.

- **IT and System Administrators**:
    - Technical staff who manage the deployment, maintenance, and security of the analytics platform, ensuring its scalability, reliability, and compliance with organizational IT policies.
- **Decision Makers and Executives**:
    - Senior leaders who utilize the high-level insights and visualizations provided by the dashboard to inform strategic planning, resource allocation, and performance evaluation.
- **Market Researchers**:
    - Researchers who benefit from the automated generation of market trends and consumer behaviour insights, aiding in the development of marketing strategies and product development plans.
- **Financial Analysts**:
    - Professionals who use the tool to analyse financial data, identify cost-saving opportunities, and support budgeting and forecasting activities with real-time data insights.
- **Educators and Researchers in Data Science**:
    - Academics and students who can use the system as a teaching tool to demonstrate the practical applications of machine learning, data visualization, and business analytics in real-world scenarios.

Reading Suggestions:

To gain a deeper understanding of Business Analytics and Automation using LLM, the intended audience is encouraged to explore the following resources:

1. **"Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow" by Aurélien Géron**:

- This book provides a practical introduction to machine learning, offering insights into various algorithms and their implementation using popular Python libraries. It's invaluable for understanding the machine learning components of the project.

2. **"Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking" by Foster Provost and Tom Fawcett**:

    - A comprehensive guide to data science concepts, this book helps bridge the gap between theoretical knowledge and practical application, making it an essential read for business analysts and data scientists involved in the project.

3. **"Python Data Science Handbook" by Jake VanderPlas**:

    - This book offers an extensive overview of data science tools in Python, including NumPy, Pandas, Matplotlib, and Scikit-Learn. It's particularly useful for data preprocessing, visualization, and analysis tasks within the project.

4. **"Natural Language Processing with PyTorch: Build Intelligent Language Applications Using Deep Learning" by Delip Rao and Brian McMahan**:

    - This resource provides insights into implementing NLP models using PyTorch, which can be beneficial for generating textual insights and understanding user queries in the project.

5. **"The Elements of Data Analytic Style" by Jeffrey Leek**:

    - This book offers practical advice on how to conduct data analysis projects effectively. It’s an excellent read for anyone looking to improve their data analysis skills and produce clear, reliable results.

6. **"Building Machine Learning Powered Applications: Going from Idea to Product" by Emmanuel Ameisen**:

    - A guide to developing machine learning applications, this book is useful

for understanding the end-to-end process of creating ML-powered systems, from concept to deployment.

2.2.1.3 Project Scope

The scope of the AnalytixAI System project encompasses the design, development, implementation, and evaluation of an innovative Dashboard providing data analytics features and comprehensive components.

The project is designed to cater to the following specific areas:

1. **Automated Data Processing**
    - Integration of LLMs to automatically analyze vast datasets and extract meaningful insights without human intervention.
    - Use of LangChain to facilitate the seamless integration of machine learning models with data analytics workflows.
2. **Visual Chart Generation**
    - Development of features that allow users to generate various types of visualizations such as bar charts, pie charts, line charts, and more, based on their specific queries and data inputs.
    - Implementation of dynamic and interactive charts that update in real-time as new data is received or user inputs change.
3. **Automated Text Insights**
    - Creation of a system that provides automated text summaries and insights about the generated visualizations, highlighting key trends, patterns, and anomalies.
    - Ensuring that the text insights are clear, concise, and provide actionable information for decision-makers.

4. **User Interface and Experience**

    - Design of an intuitive and user-friendly interface that allows users of varying technical expertise to easily interact with the dashboard.
    - Focus on user experience to ensure that the dashboard is accessible, responsive, and provides a seamless user experience.

2.2.2 Overall Description

2.2.2.1 Product Perspective

The AnalytixAI dashboard is designed to be a standalone product that enhances and integrates with existing data analytics and business intelligence systems within an organization.

1. **Enhancement of Capabilities**: By offering automated data processing, visualization, and insights generation, AnalytixAI enhances the capabilities of existing systems. It adds value by simplifying complex data analytics tasks and providing real-time insights.

2. **Complementary Tool**: Rather than competing with or replacing existing systems, AnalytixAI complements them. It fills gaps in automation and provides a user-friendly interface for data analysis.

3. **User Accessibility**: The dashboard is designed to be user-friendly, catering to a wide range of users from business analysts to executives. Its intuitive interface makes it accessible to users with varying levels of technical expertise.

4. **Scalability and Adaptability**: The dashboard is scalable and adaptable, capable of handling large volumes of data and evolving with changing business needs. It can accommodate new data sources and analytics requirements over time.

2.2.2.2 Product Features

1. **Automated Data Visualization:**

   The dashboard seamlessly generates a variety of visual charts (bar charts, line graphs, pie charts, etc.) based on user queries, providing instant, high-quality visual representations of data without the need for manual coding.

2. **Advanced Natural Language Processing (NLP):**

   Utilizing cutting-edge NLP techniques, the system accurately interprets user queries in natural language, transforming them into actionable insights and precise data visualizations, making it highly user-friendly and intuitive.

3. **Real-Time Insights and Analysis:**

   The platform not only displays visual charts but also provides automated textual insights that highlight key patterns and trends, offering a comprehensive analytical view that aids in faster and better-informed decision-making.

4. **Scalability and Performance:**

   Designed to handle large datasets and multiple simultaneous user requests, the system ensures high performance and reliability, making it suitable for both small businesses and large enterprises seeking efficient data analytics solutions.

2.2.3 External Interface Requirements

2.2.3.1 User Interfaces

- The AnalytixAI dashboard will have a web-based user interface accessible through standard web browsers such as Chrome, Firefox, and Safari. The interface will be designed to be intuitive and user-friendly, with interactive elements for data visualization and exploration.

- The dashboard will support multiple user roles, including administrators,

analysts, and executives, each with specific permissions and access levels. User authentication will be implemented to ensure secure access to the dashboard.

- The user interface will include features such as input functionality for creating visualizations, customizable dashboards, and real-time data updates. It will also provide options for exporting charts in various formats, and uploading data in forms such as CSV.

2.2.3.2 Hardware Interfaces

- The AnalytixAI dashboard will be designed to run on standard hardware configurations, including desktop computers, laptops, and mobile devices. It will be compatible with Windows, macOS, and Linux operating systems.
- For optimal performance, the dashboard may require hardware components such as a modern CPU, sufficient RAM, and a graphics card capable of rendering complex visualizations. However, the dashboard will be designed to be lightweight and responsive, ensuring compatibility with a wide range of hardware configurations.

2.2.3.3 Software Interfaces

1. LLM Requests:

The system utilizes Large Language Models (LLMs) to process and interpret user queries. User inputs are sent to the LLM, which generates relevant code prompts and analytical instructions based on the query. This interface ensures the accurate conversion of user queries into actionable data processing and visualization tasks.

2. Langchain Integration:

LangChain serves as the bridge between user queries and backend processing. It retrieves relevant documentation and code instructions from a vector database, facilitating smooth interaction with the LLM. This interface is essential for dynamically generating the code required for data analysis and

visualization.

3. Data Management and Processing:

The system interfaces with various data management tools and libraries for dataset uploads, unique sampling, and preprocessing. This ensures the data is clean, well-structured, and ready for analysis, enhancing the overall efficiency and accuracy of the analytics process.

4. Visualization Libraries:

The interface includes integration with visualization libraries such as Matplotlib, Seaborn, and Plotly. These libraries are used to generate a wide range of charts and graphs, ensuring high-quality visual representations of the data as per user requests.

5. Frontend Interface (Streamlit):

Streamlit serves as the frontend interface, providing a user-friendly platform for query input and displaying results. It interacts seamlessly with the backend, ensuring that user queries are processed efficiently and results are rendered in real-time on the dashboard.

2.2.4 Other Non-Functional Requirements

2.2.4.1 Performance Requirements

Performance requirements are a set of specific and measurable criteria that outline how a system or software application should perform in various aspects. These requirements are essential for ensuring that the system meets its intended objectives and delivers a satisfactory user experience.

1. **Speed**:

    - The system must process user queries and generate visualizations within

a few seconds to ensure a smooth and efficient user experience. This involves optimizing data retrieval, processing, and rendering times.

2. **Accuracy**:

    - The analytics and visualizations generated by the system must be accurate and reliable. The LLMs and LangChain integration should ensure precise interpretation of user queries and correct execution of data processing tasks.

3. **Response Time**:

    - The dashboard should maintain a minimal response time for user interactions, including query submissions and dataset uploads.

4. **Scalability**:

    - The system should be able to handle an increasing number of user requests and larger datasets without degradation in performance. This includes supporting multiple concurrent users and ensuring consistent performance as data volume grows.

2.2.4.2 Safety Requirements

Safety requirements aim to protect user data and ensure the secure operation of the AnalytixAI dashboard, mitigating risks related to data security and privacy.

1. **Data Security**: The dashboard shall ensure the security and confidentiality of user data through encryption and secure data storage practices.

2. **User Authentication**: The system shall require users to authenticate themselves before accessing sensitive data or performing certain actions, ensuring that only authorized personnel can access the dashboard.

3. **Data Integrity**: The dashboard shall maintain the integrity of data by ensuring that it is not tampered with or altered during processing or storage.

4. **Backup and Recovery**: The system shall regularly backup data and provide mechanisms for data recovery in case of data loss or corruption.

5. **Access Control**: The dashboard shall implement access control measures to restrict access to sensitive features or data based on user roles and permissions.

2.2.4.3 Security Requirements

1. **Data Protection:**

   All data transmitted between the user and the server, including queries and datasets, must be secured and protected using industry-standard encryption protocols to protect sensitive information from unauthorized access.

2. **Authentication and Authorization:**

   The system must implement robust user authentication and authorization mechanisms to ensure that only authorized users can access specific features and data.

3. **Regular Security Updates:**

   The system must undergo regular security updates to identify and address vulnerabilities. Additionally, it should be updated periodically to incorporate the latest security patches and enhancements, ensuring ongoing protection against emerging threats.

2.3 Cost Analysis

The primary cost involved in this project is the access and integration of Large Language Models (LLMs) for processing heavy datasets. These costs can be substantial, particularly as the system scales to accommodate more users and larger datasets. The following is a general overview on the potential costs:

1. **LLM Access and Usage**:

- **Subscription Fees**: Accessing high-quality LLMs typically involves subscription fees, which can vary based on the model's capabilities and usage volume. These fees can become significant, especially as the number of queries and the complexity of data processing increase.

- **Processing Costs**: Heavy datasets require substantial computational power for processing. This can lead to increased costs associated with cloud computing resources or on-premise hardware capable of handling intensive processing tasks.

2. **Infrastructure and Hosting**:

    - **Cloud Services**: Hosting the application on cloud platforms (e.g., AWS, Google Cloud, Azure) incurs costs related to server usage, data storage, and bandwidth. As the system scales, these costs can grow proportionally, especially with high data throughput and storage requirements.

    - **Scalability Measures**: Implementing scalable infrastructure to ensure consistent performance under high load involves additional costs. This includes load balancers, auto-scaling groups, and advanced database solutions.

3. **Data Management and Storage**:

    - **Database Solutions:** Efficiently managing and storing large volumes of data requires robust database solutions, which can be costly. This includes costs for database software licenses, maintenance, and scaling storage solutions.

    - **Backup and Recovery:** Ensuring data integrity and availability involves costs related to regular backups, disaster recovery solutions, and data replication across multiple locations.

2.4 Risk Analysis

1. **Data Breach**: There is a risk of a data breach leading to unauthorized access to sensitive information. Mitigation: Implement strong security standards for data at rest and in transit, enforce strict access controls, and conduct regular security audits

2. **Performance Degradation**: Under heavy load, the system may experience performance degradation, impacting user experience. Mitigation: Conduct load testing to identify and address performance bottlenecks, and scale resources as needed.

3. **Data Loss**: There is a risk of data loss due to hardware failure or software bugs. Mitigation: Implement regular data backups and ensure that data recovery mechanisms are in place.

4. **Data Accuracy**: Inaccurate data analysis or visualization may lead to incorrect decision-making. Mitigation: Implement data validation checks and ensure that algorithms used for analysis are accurate and reliable.

METHODOLOGY ADOPTED

3.1 Investigative Techniques

Experimental:

The investigative approach employed in this project is predominantly experimental, aligning with the need to rigorously test and validate the effectiveness of the automated business analytics system. This method involves a systematic investigation that includes control groups to benchmark performance, along with independent and dependent variables to test various hypotheses related to data processing, visualization, and insights generation. By employing machine learning techniques, the project aims to refine its algorithms and enhance the overall system performance, ensuring that the analytics produced are both accurate and actionable.

Methodology:

The methodology for the experimental investigation involves the following key steps:

1. Data Collection and Preprocessing:

The project begins with the collection of diverse datasets representing different business scenarios. These datasets are preprocessed to ensure they are clean, structured, and suitable for analysis. Preprocessing steps include data sampling(removing duplicates, handling missing values) and normalization (scaling data to a standard range).

2. Experimental Design and Control Groups:

To validate the system's effectiveness, the experimental design incorporates

control groups that use traditional manual data analysis methods. The experimental groups use the automated system powered by LLMs and LangChain technology. Independent variables include different types of data inputs and user queries, while dependent variables are the accuracy, speed, and relevance of the generated insights and visualizations.

3. Implementation of LLM and LangChain Models:

The core of the project involves implementing LLM and LangChain to automate data analysis and visualization. The models are trained using the preprocessed datasets, with a focus on optimizing performance metrics such as accuracy, processing speed, and resource utilization. LangChain is used to retrieve relevant code instructions from a vector database, which are then fed into the AnalytixAI LLM model. This model generates code prompts that are executed to produce visual charts and textual insights.

3.2 Proposed Solution

The proposed solution for the "AnalytixAI: Automated Business Analytics Dashboard using LLM and LangChain" project involves a comprehensive approach to automate data analytics and visualization. The methodology and techniques used in implementing the project can be described in the following key points:

1. User Query Processing:

Users input queries regarding data representation and analytics needs through an intuitive dashboard interface. The system accepts various query formats, allowing users to request specific types of visualizations (e.g., bar charts, pie charts) and insights based on their uploaded datasets.

2. Data Preprocessing and Sampling:

Uploaded datasets undergo preprocessing steps, including data cleaning and

sampling, to ensure data quality and consistency. Unique sampling techniques are applied to extract representative rows, ensuring that the analytics are based on relevant and clean data subsets.

3. LangChain Document Retrieval:

LangChain's document retriever is utilized to fetch relevant code instructions from a pre-built vector database. This component identifies the necessary analytical procedures and code functions required to process the user queries and datasets effectively.

4. LLM-Driven Analytics Generation:

The core analytics engine, powered by the AnalytixAI LLM model, processes the inputs from LangChain. The model generates relevant code prompts and analytical functions to execute data visualizations and generate textual insights. This includes creating the requested charts and extracting meaningful patterns and trends from the data.

5. Visualization and Insight Display:

The generated visualizations and textual insights are displayed on the user-friendly dashboard. Users can interact with these results, gaining immediate and actionable insights into their data. The system ensures that the visual representations are accurate and the accompanying textual explanations are precise and contextually relevant.

3.3 Work Breakdown Structure

Work Breakdown Structure showing the various components of the project:

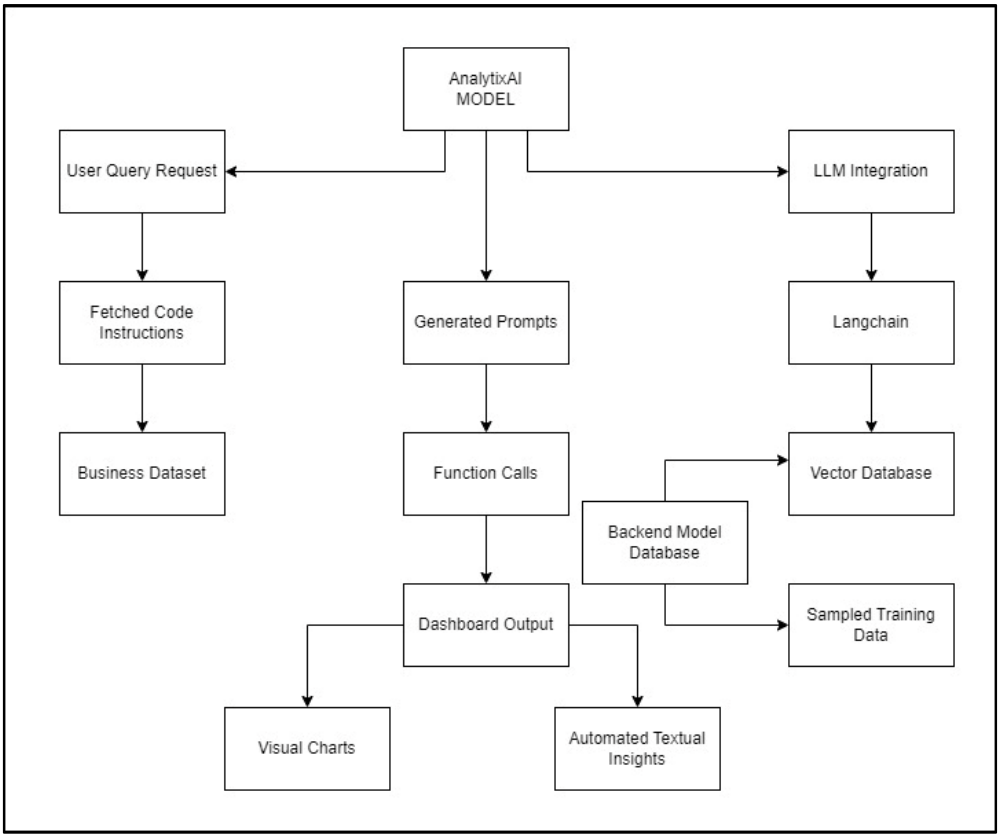


Figure 3.3.1: Work Breakdown Structure for the entire system clearly describing the segments and blocks for the LLM and LangChain structure of the AnalytixAI Model.

Gantt Chart:

	Task	Duration	Start	Finish
1	**Identification, Formulation, Plannning**	**15 Days**	**15-02-2024**	**29-02-2024**
1.1	Defining Strategy	8 Days	15-02-2024	22-02-2024
1.2	Literature Review	Project Duration	15-02-2024	22-05-2024
1.3	Project Scope	7 Days	23-02-2024	29-02-2024
2	**Requirement Analysis**	**10 Days**	**01-03-2024**	**10-03-2024**
2.1	Feasibility Study	10 Days	01-03-2024	10-03-2024
2.1.1	Operational Feasibility	4 Days	01-03-2024	04-03-2024
2.1.2	Resource Availability	6 Days	05-03-2024	10-03-2024
3	**Utility Analysis**	**15 Days**	**11-03-2024**	**25-03-2024**
3.1	Value Proposition	15 Days	11-03-2024	25-03-2024
4	**Designing and Development**	**15 Days**	**26-03-2024**	**09-04-2024**
4.1	Design and Interfacing	15 Days	26-03-2024	09-04-2024
5	**Implementation**	**10 Days**	**10-04-2024**	**19-04-2024**
6	**Optimization**	**10 Days**	**20-04-2024**	**29-04-2024**
7	**Testing and Updation**	**12 Days**	**30-04-2024**	**11-05-2024**
7.1	Result Evaluations	12 Days	30-04-2024	11-05-2024
8	**Final Report**	**10 Days**	**12-05-2024**	**22-05-2024**

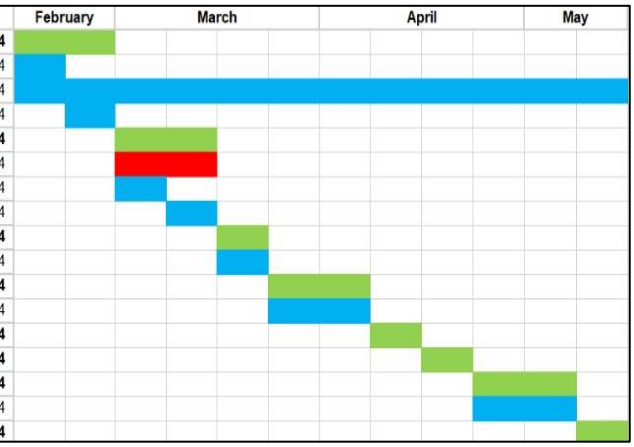


Figure 3.3.2: Gantt Chart explaining the workflow division with corresponding timelines.

3.4 Tools and Technology

- **Large Language Models (LLM)**:
  Utilized to generate code prompts and perform advanced data analysis, enabling automated and accurate insights based on user queries and datasets.
- **LangChain**:
  Employed for document retrieval, fetching relevant code instructions from a vector database to facilitate seamless and contextually accurate data processing.
- **Python**:
  The primary programming language used for backend development, enabling the implementation of machine learning models, data processing algorithms, and integration with other tools.
- **Code Interpreter**:
  Integrated to execute generated code prompts, transforming user queries into actionable data visualizations and insights with high precision.
- **Streamlit**:
  Used for the frontend development of the interactive dashboard, providing a user-friendly interface for inputting queries, uploading datasets, and viewing results.
- **Machine Learning Libraries**:
  Libraries such as Scikit-Learn, Pandas, and NumPy are used for data analysis, preprocessing, and implementing various machine learning algorithms.
- **Data Visualization Libraries**:
  Tools like Matplotlib, Seaborn, and Plotly are used to create dynamic and informative visual charts and graphs.

4.1 System Architecture

The Block Diagram can be used to accurately illustrate the system's design as shown in Figure 4.1.1:

4.1.1 Block Diagram

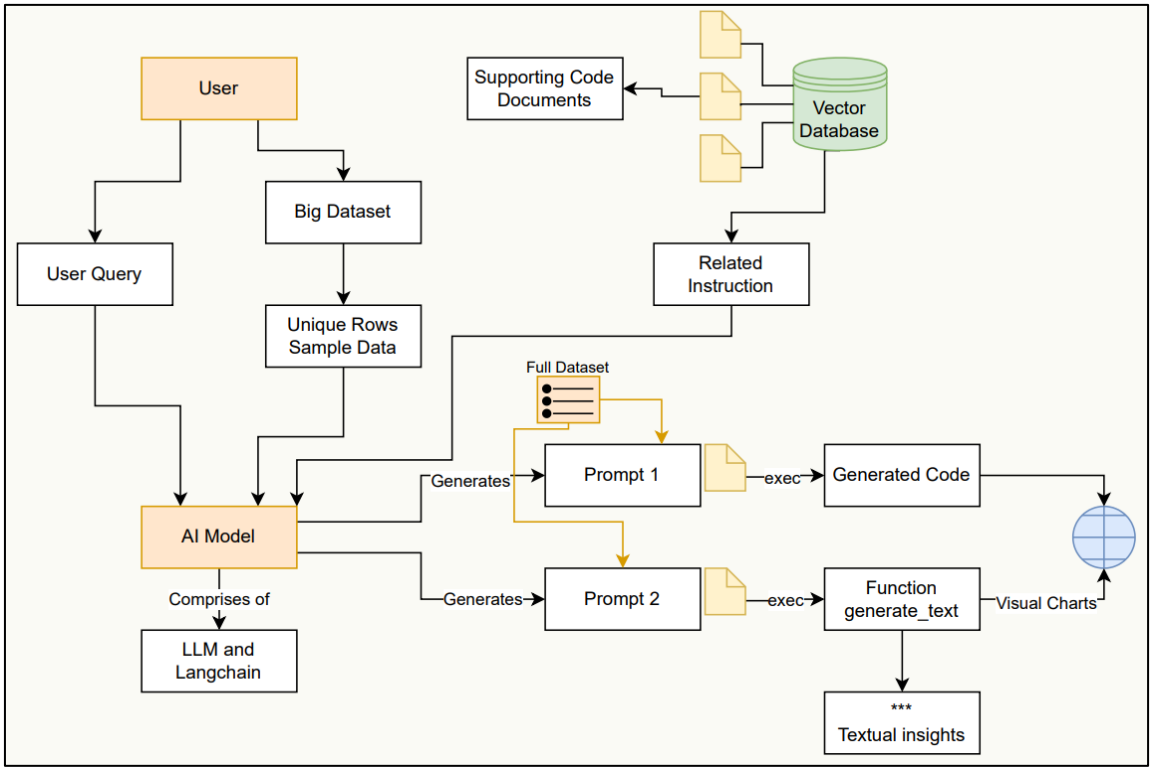


Figure 4.1.1 Block Diagram

Figure 4.1.1 refers to the block Diagram of the entire workflow and connected components of AnalytixAI: Automated Dashboard system. The user enters a user query and the corresponding dataset for analysis, further the model works on it for generating the appropriate prompt leading to output of the visual charts and textual insights.

4.1.2 Activity Diagram

An activity diagram depicts the sequential flow of control from an initial point to a final point with various decision points that arise during the execution of the activity.

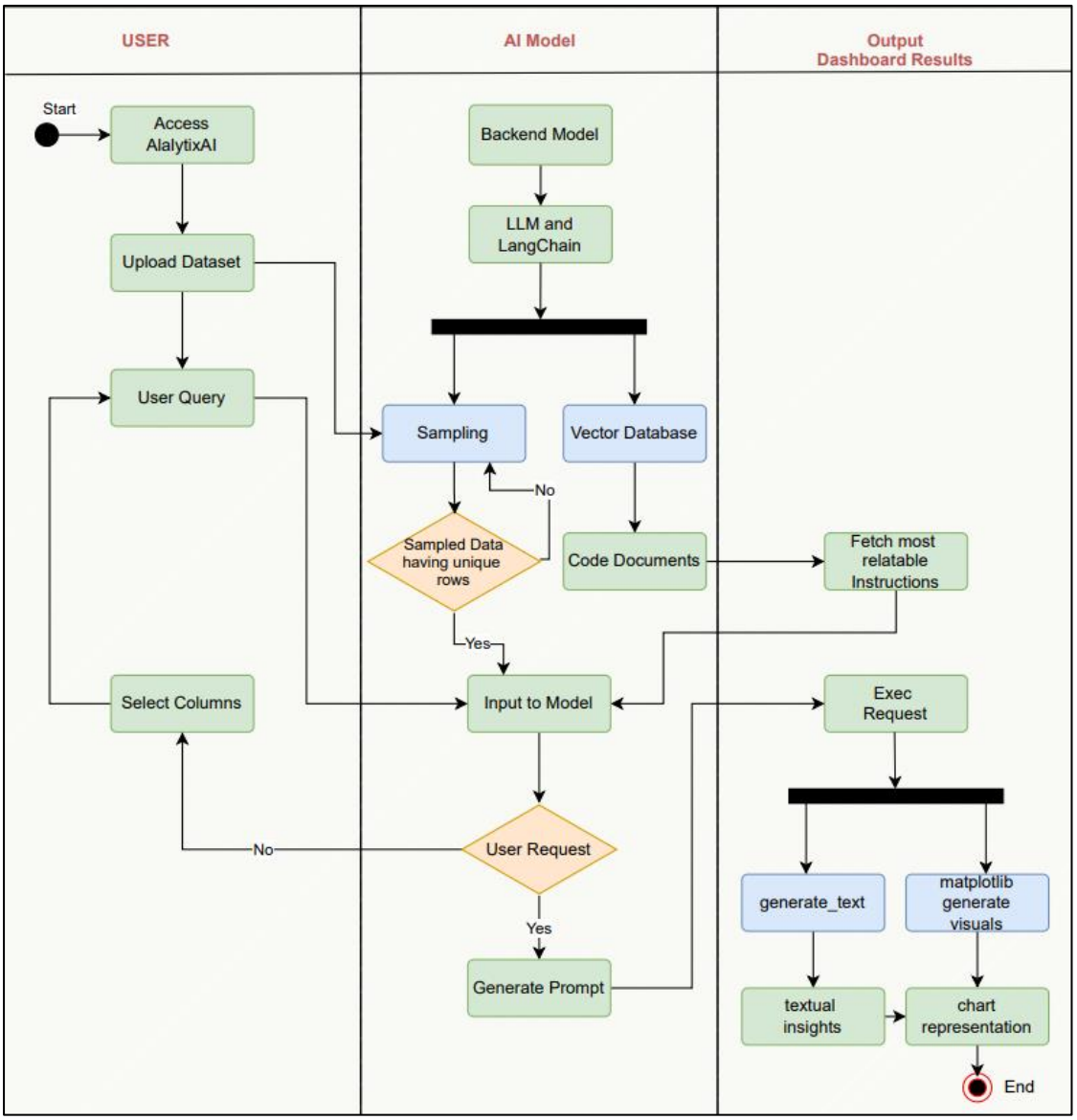


Figure 4.1.2 Activity Diagram

Figure 4.1.2 refers to the Activity Diagram of the project with the start node showing the initiation of the project. Further the functionality of the model is explained with the various decision points in the execution leading to the final results of automated Dashboard for corresponding user query and Data Analysis.

4.2 Design Level Diagrams

The following Data Flow Diagrams (Level 0 and Level 1) show the changes and transformations in data flows among all processes.

4.2.1 Data Flow Diagrams (DFD):

4.2.1.1 Level 0

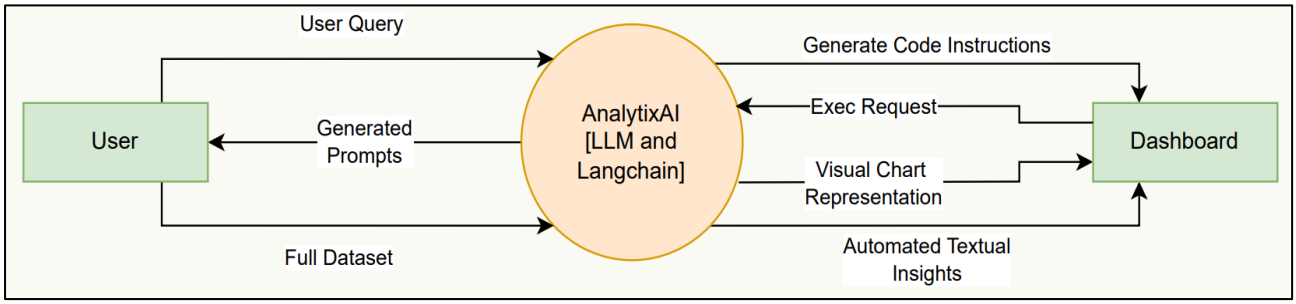


Figure 4.2.1.1 DFD (Level 0)

Figure 4.2.1.1 The Level 0 Data Flow Diagram (DFD) provides a high-level overview of the AnalytixAI system, illustrating the key components and the flow of data between them. The primary components in this diagram are the User, the LLM and LangChain System, and the Dashboard. The directed flow of data from the user to the model and the subsequent flow of output information to the final dashboard are depicted, ensuring a clear understanding of how the data analytics model operates.

4.2.1.2 Level 1

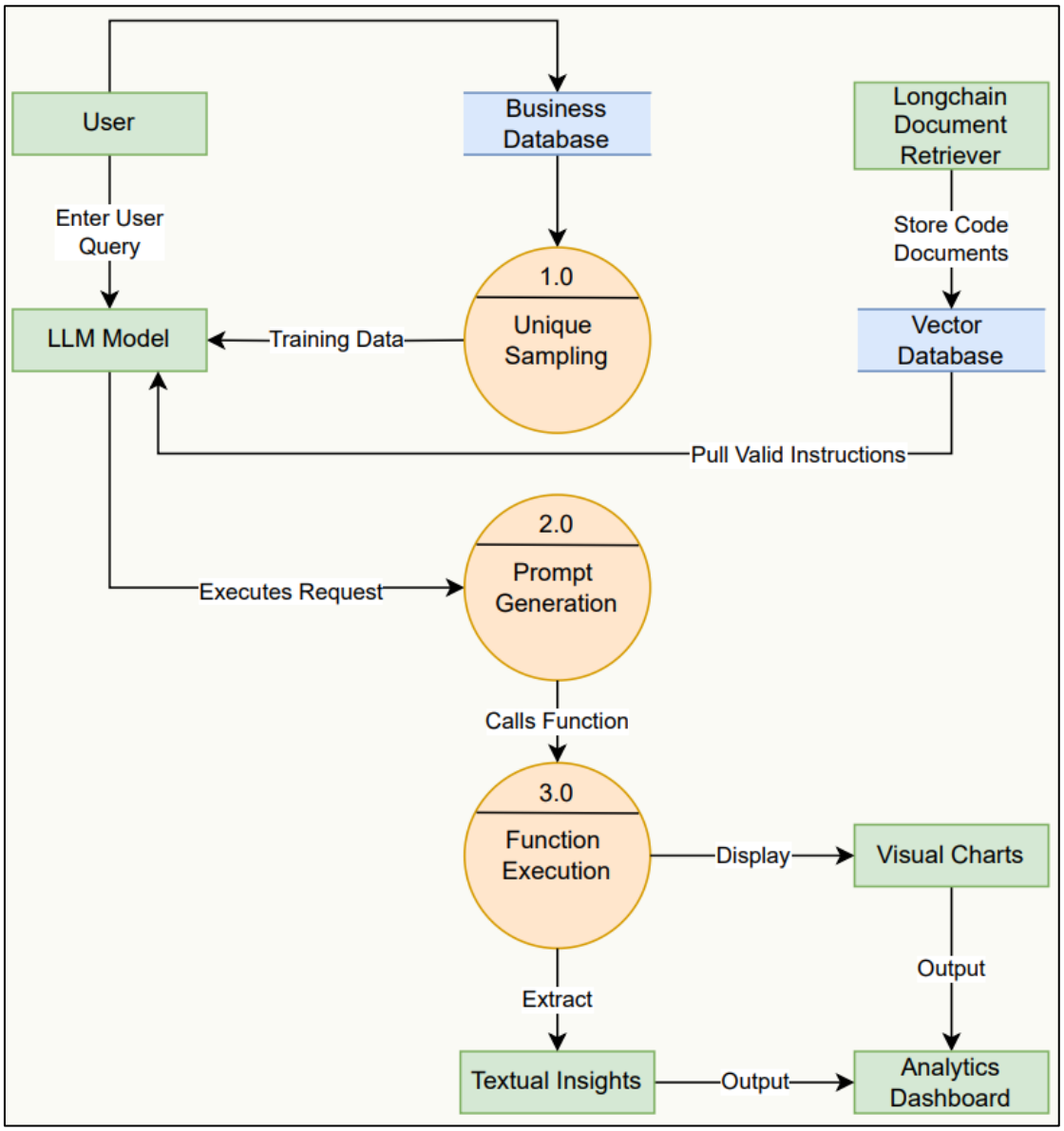


Figure 4.2.1.2 DFD (Level 1)

Figure 4.2.1.2 refers to the level 1 of the Data Flow Diagram which delves deep into the various operations that occur in each process breaking down the high level components and shows the data interactions of Sampling, Prompt Generation and further Execution transforming user queries to visual outputs.

4.2.2 ER Diagram

The given below ER Diagram shows the conceptual model of our Analytics Dashboard System defining the relationship between different entities of our project.

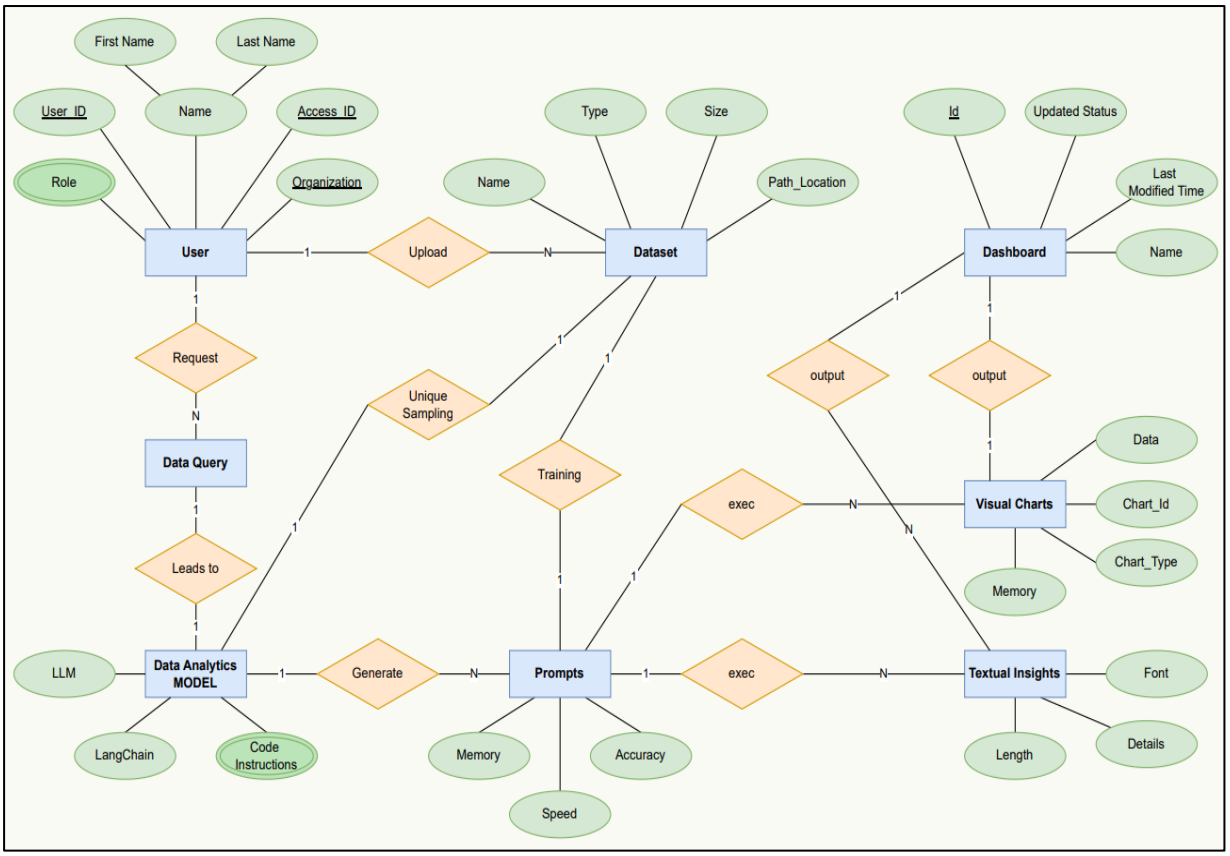


Figure 4.2.2 ER Diagram

Figure 4.2.2 refers to the ER diagram showing the connections between all entities. The entities, characteristic attributes and relationships interact to describe the entire system of the project. The diagram shows how the user request and data are processed through efficient system design and implementation of interrelated connections leading to desired output.

4.2.3 Class Diagram

The given Class Diagram represents the various aspects of the system and their corresponding functions.

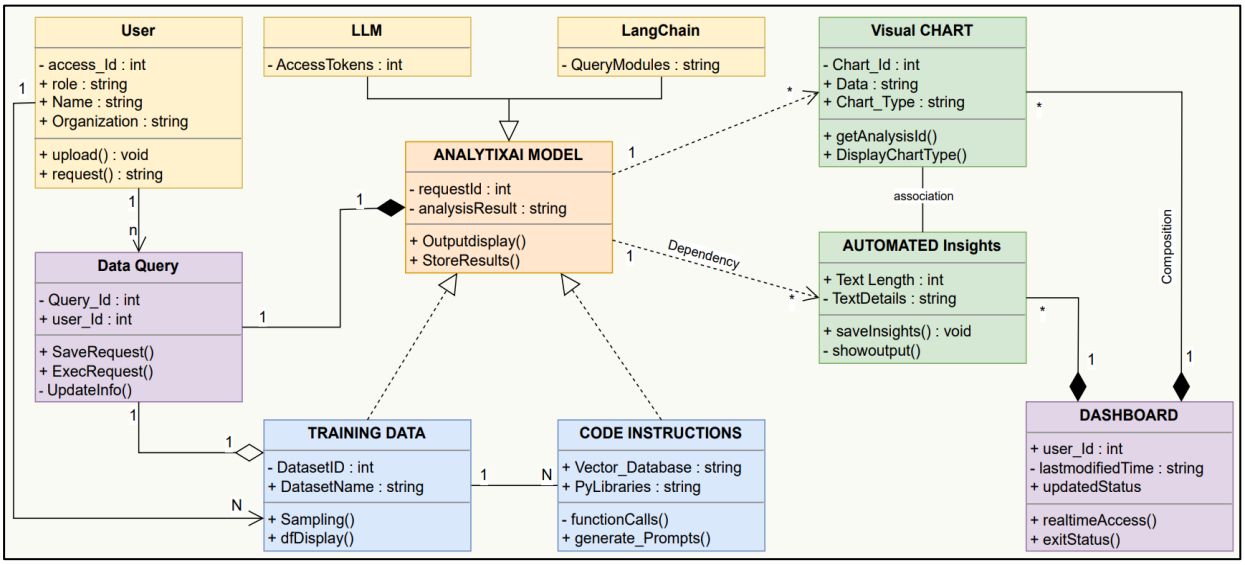


Figure 4.2.3 Class Diagram

Figure 4.2.3 refers to the Class diagram of the project. The dependencies at various levels are shown and through this visualization all the objects and classes used in the entire project are clearly depicted. The primary types of relationships in a class diagram include associations, generalizations, aggregations, and compositions. Each class has some attributes and related operations and each class possesses some relationship with other classes to maintain the sync between all components of the analysis model. The model is dependent on the training data and code instructions while the model has one to one complete relationship with the outputs (Visual Charts and Automated Insights) which are closely associated with the dashboard which is the final product of our AnalytixAI Model.

4.3 User Interface Diagrams

4.3.1 Use Case Diagram

The diagram illustrates the various ways in which use cases interact with the system.

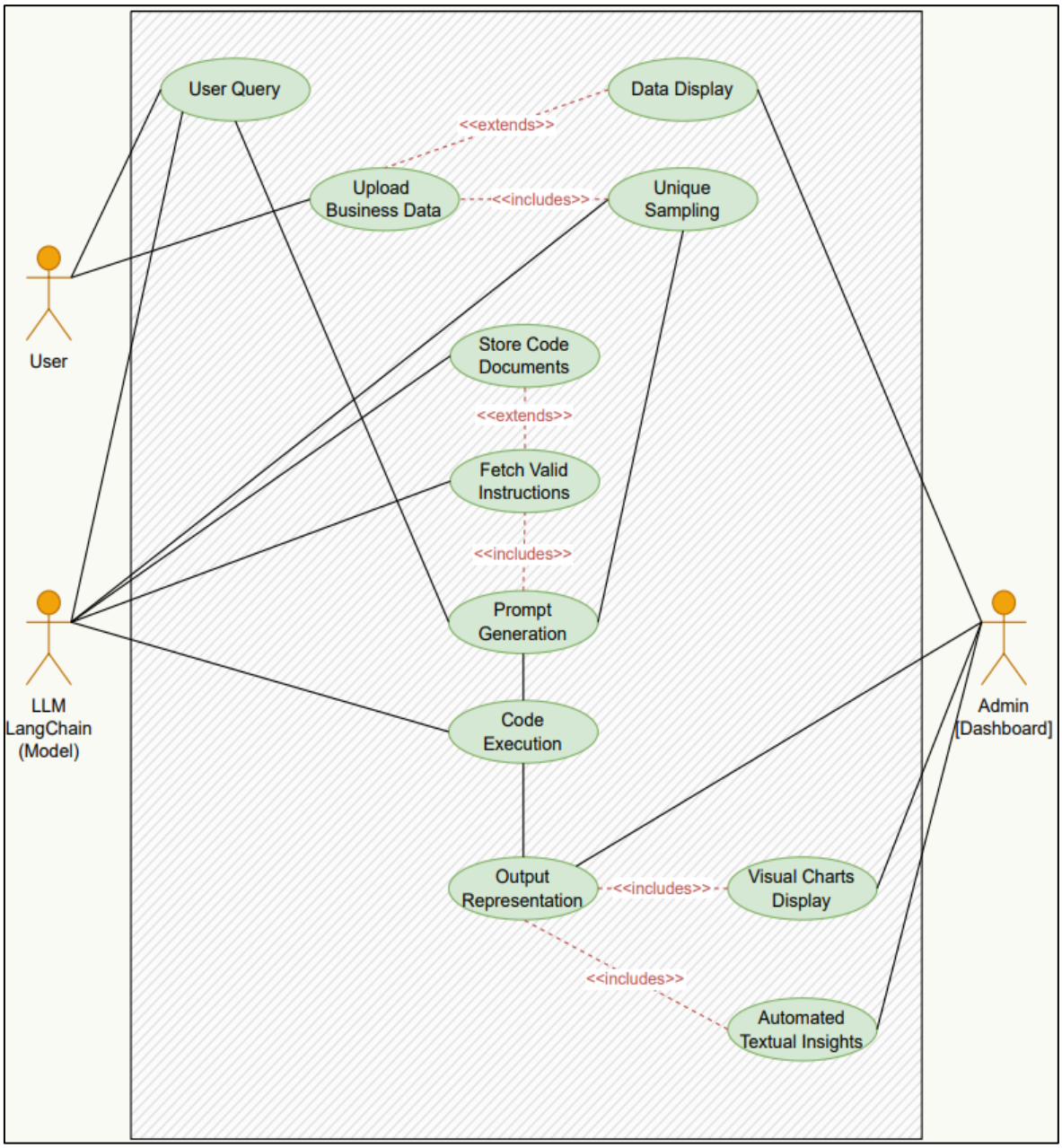


Figure 4.3.1 Use case Diagram

Figure 4.3.1 depicts the various functionalities and interactions for the project execution among the actors that is user and dashboard through the component analysis.

4.3.1.1 Use Case Templates

Use Case #1 (Upload Business Data and Id = 1)

Use Case ID	1.0
Use Case Title	Upload Business Data
Actors	User, Admin
Description	With this feature user can upload any dataset.
Precondition	User should have business analytics queries.
Task Sequence	1. User accesses AnalytixAI system. 2. Dashboard Interface provides dataset upload feature. 3. User uploads dataset with all metrics and parameters.
Postconditions	1. Dataset is processed in backend model. 2. Unique Sampling is performed.
Alternative Flows	1.Large Datasets might have resource limitation. 2.Multiple data might not be accessed simultaneously.

Table 4.3.1.1.1: Use Case template for Upload Business Data

Use Case #2 (Prompt Generation and Id = 2)

Use Case ID	2.0
Use Case Title	Prompt Generation
Actors	User, LLM (Model),Admin
Description	Generation of appropriate code instruction prompts.
Precondition	Execution of function calls in code prompts.
Task Sequence	1. Code prompts are executed. 2. Function Call1 generates desired visual charts. 3. Automated Textual Insights are displayed corresponding to the representations.
Postconditions	1. Interactive Dashboard for data analysis. 2. Easily explored charts for various purposes.
Alternative Flows	1.Additional instructions might be needed. 2.Multiple user queries generate multiple prompts.

Table 4.3.1.1.2: Use Case template for Prompt Generation

Use Case #3 (Output Representation and Id = 3)

Use Case ID	3.0
Use Case Title	Output Representation
Actors	User, LLM (Model), Admin
Description	Accurate Visual Representations and Textual Insights
Precondition	Retrieved code documents from Vector Database.
Task Sequence	1. User accesses model and uploads dataset. 2. LangChain retriever extracts appropriate code instructions. 3. Code prompts with functions are generated.
Postconditions	1. Function calls in prompts are executed. 2. Prompts perform analysis on entire dataset.
Alternative Flows	1. Multiple representations at concurrent queries. 2. Inclusion of predictive and more complex analytics.

Table 4.3.1.1.3: Use Case template for Output Representation

IMPLEMENTATION AND EXPERIMENTAL RESULTS

5.1 Experimental Setup

Data Collection and Preparation

- **Datasets**: Various datasets were collected from different business domains to test the system's versatility. These datasets include sales data, customer data, financial records, and operational metrics.
- **Data Preprocessing**: The datasets underwent preprocessing steps such as unique sampling to ensure high-quality data for analysis. This involved removing duplicates, handling missing values, and transforming data into a suitable format.

System Configuration

- **Backend Setup**: The backend was configured using LLM and integrating with the LangChain library for document retrieval and instruction handling.

Development Environment

- **Frontend Development**: The frontend was developed using streamlit to create a user-friendly interface for inputting queries and displaying results. The interface allows users to upload datasets, specify query parameters, and view visualizations and insights.
- **Database Setup**: A vector database was used to store and retrieve relevant code instructions. The database was populated with a diverse set of instructions to cover various data analysis and visualization needs.

5.2 Experimental Analysis

5.2.1 Data

The data from various business development datasets have been used for training so that the model becomes versatile for all use case scenarios.

The model is capable for working on pre uploaded datasets while also on randomly generated data. The data rows further undergo sampling to resolve redundancy and lead to unique rows and columns to generate prompts and function execution.

5.2.2 Performance Parameters

1. **Accuracy**

   The correctness of the generated visualizations and insights in reflecting the underlying data. Compare the outputs with known results or manually verified data.

2. **Response Time**

   The time taken from the user submitting a query to the display of the results.

   Measure the time intervals for different stages, including data preprocessing, query processing, and visualization rendering.

3. **Throughput**

   The number of queries the system can handle and process in a given time frame.

   Count the number of successful query responses over a specified period.

4. **Resource Utilization**

   The efficiency with which the system uses computational resources (CPU, memory, storage).

   Monitor resource usage during different operations and identify any bottlenecks or inefficiencies.

5. **Robustness**

   The ability of the system to handle errors, unexpected inputs, and edge cases without crashing.

   Introduce various error conditions and invalid inputs, observing how the system handles them.

5.3 Working of the Project

5.3.1 Procedural Workflow

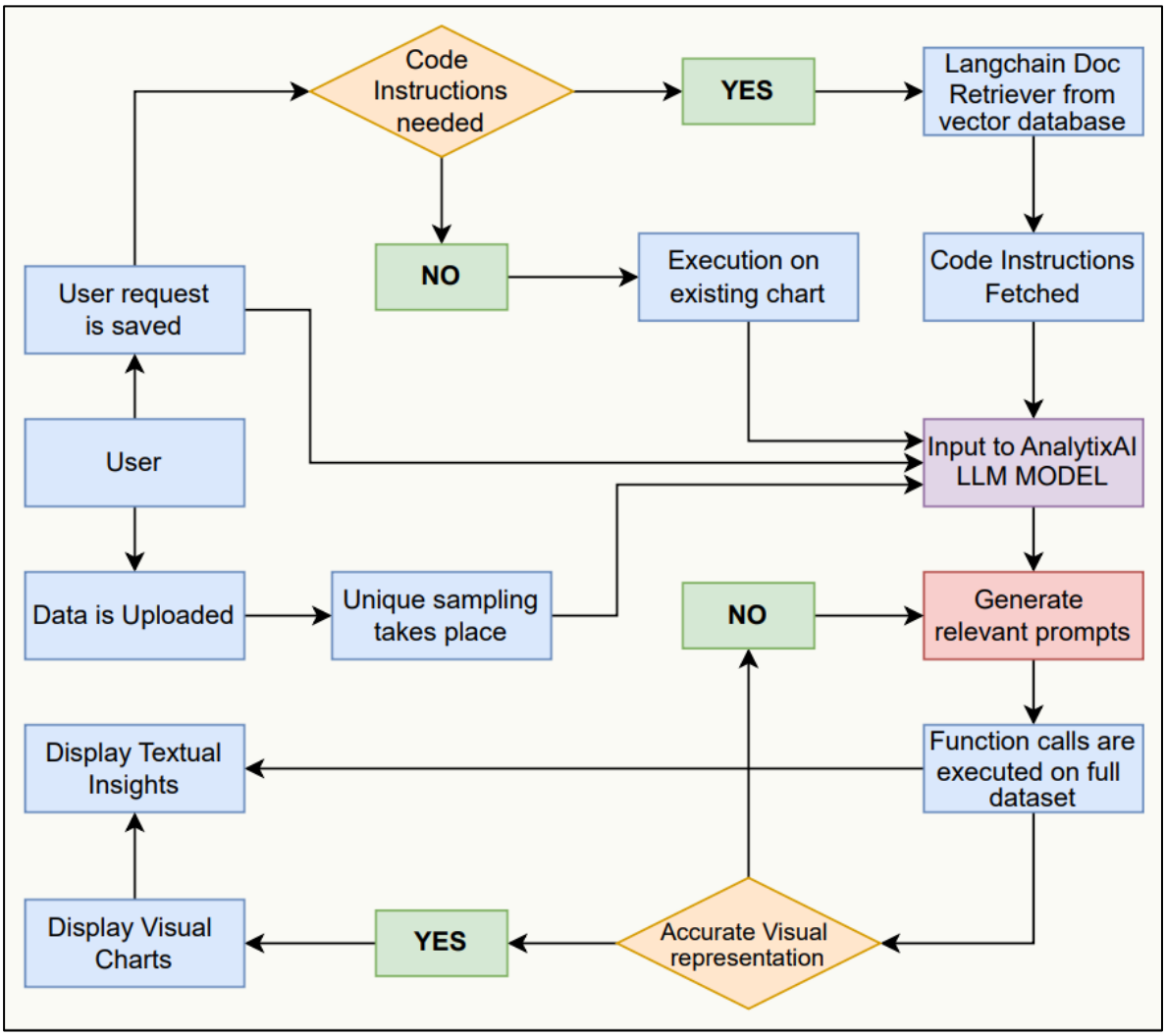


Figure 5.3.1 Procedural Workflow Diagram

Figure 5.3.1 depicts the flowchart of entire procedures carried out in the execution of the AnalytixAI Model to display the desired visual charts and automated textual insights for the user queries on the large business data and fulfill all the necessary requirements. The decision points are clearly depicted and the sequential workflow through the component entities like LLM model and vector database are implemented.

5.3.2 Project Deployment

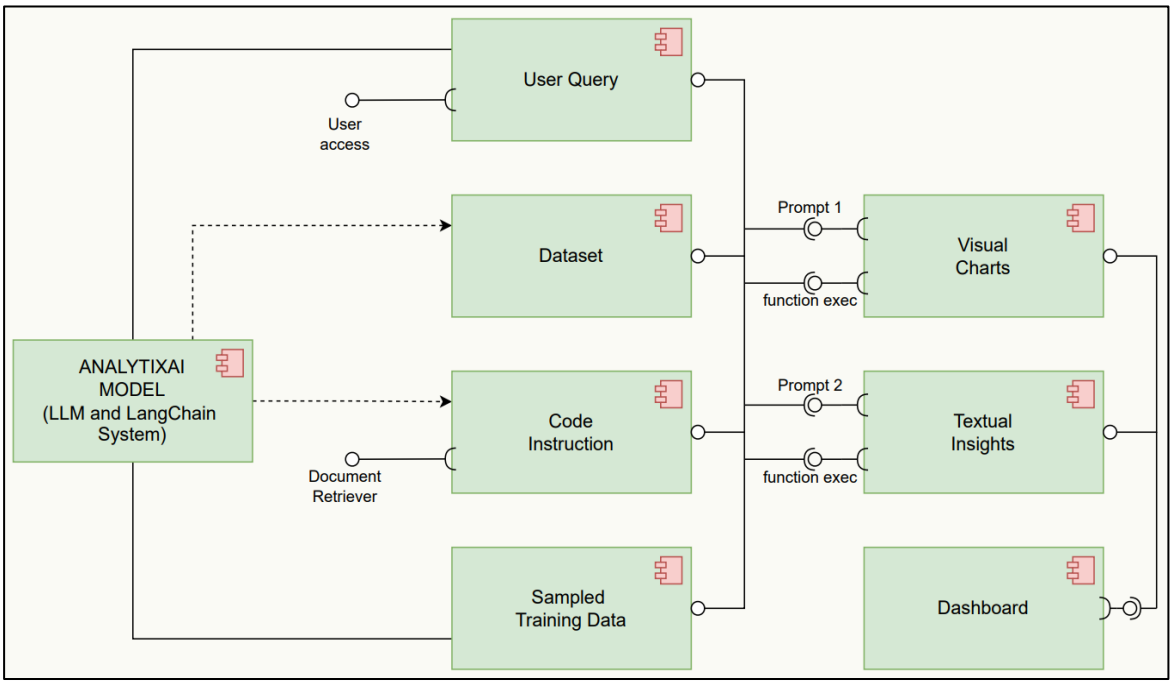
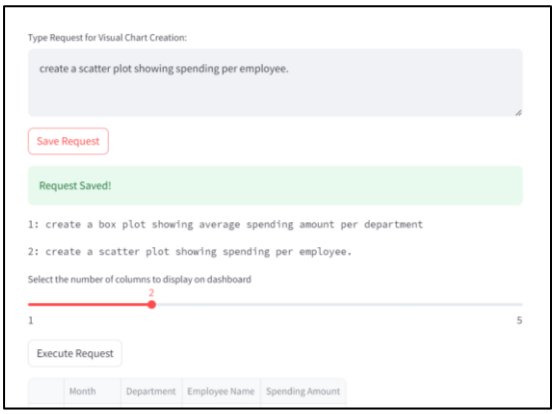


Figure 5.3.2.1 Component Diagram (Model Design)

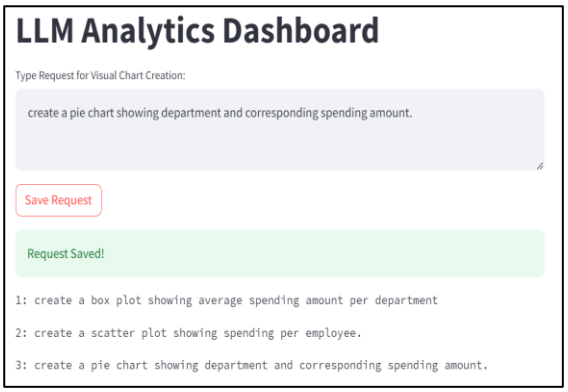
Figure 5.3.2.1 depicts all the components used for the project. The user access initiates the user query and code instructions are extracted using document retriever. All the required inputs are fed to the model for generating appropriate prompts and function execution to yield desired results specific to user and data.

5.3.3 System Screenshots

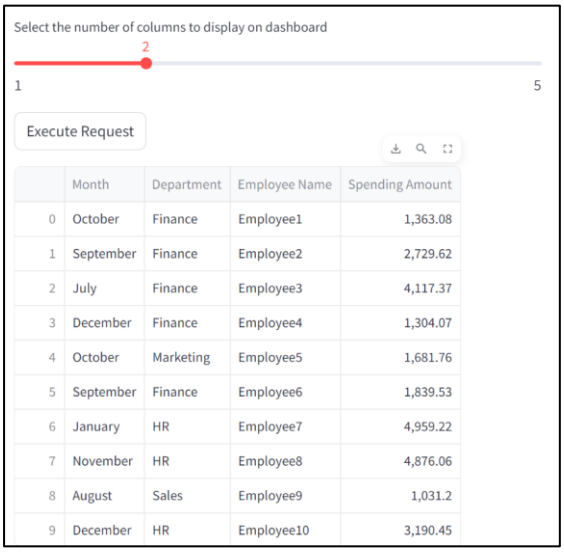
Following are the system software screenshots of the output dashboard and working of the project with all the features that are displayed in the final result.



(a) User Query Request



(b) Multiple User Queries at one time



	Month	Department	Employee Name	Spending Amount
0	October	Finance	Employee1	1,363.08
1	September	Finance	Employee2	2,729.62
2	July	Finance	Employee3	4,117.37
3	December	Finance	Employee4	1,304.07
4	October	Marketing	Employee5	1,681.76
5	September	Finance	Employee6	1,839.53
6	January	HR	Employee7	4,959.22
7	November	HR	Employee8	4,876.06
8	August	Sales	Employee9	1,031.2
9	December	HR	Employee10	3,190.45

(c) Selecting number of columns to be displayed

Figure 5.3.3.1 User Interaction to input customized Queries and requests.

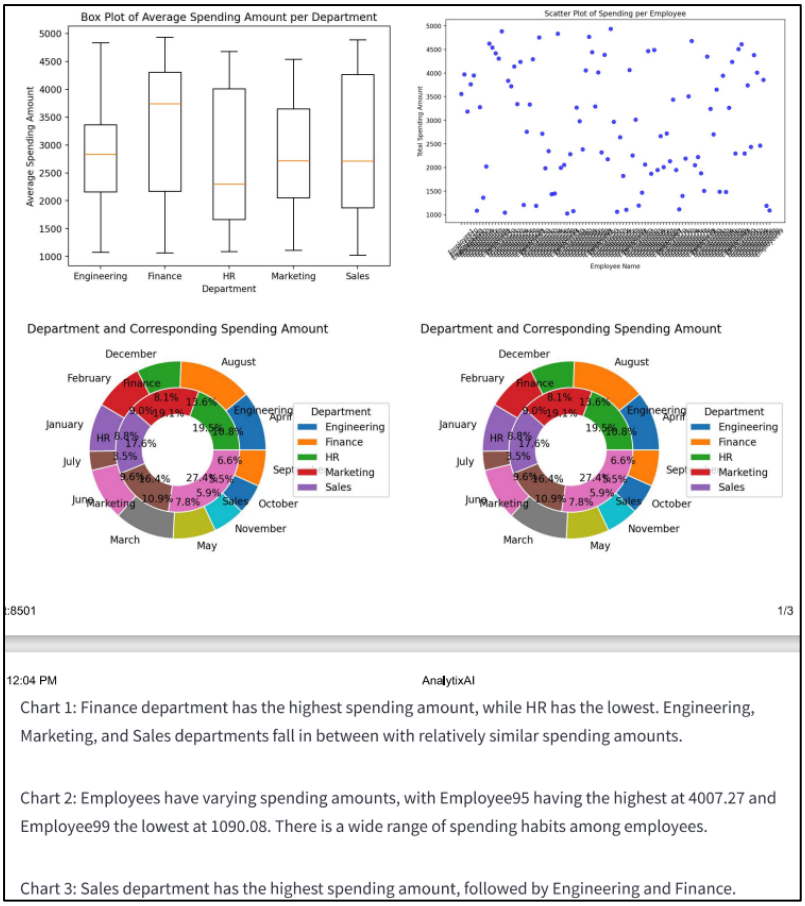
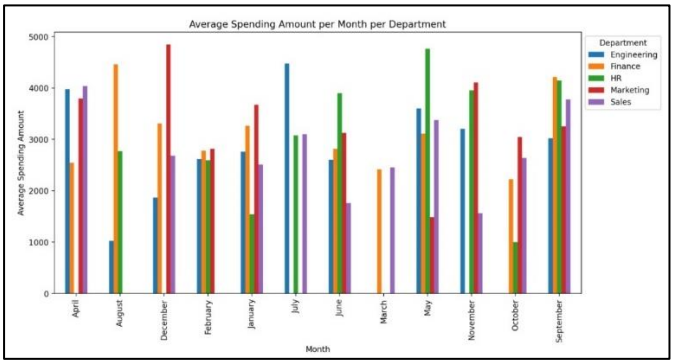
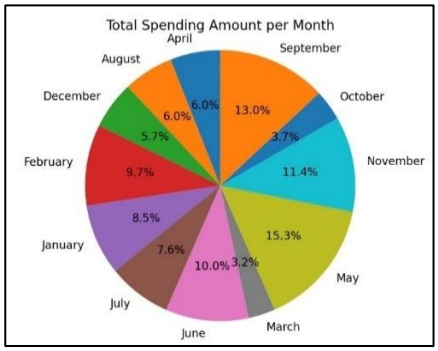


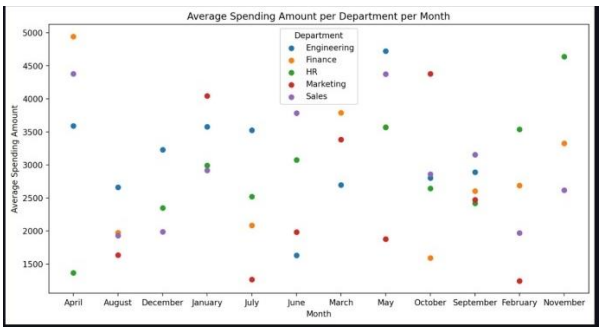
Figure 5.3.3.2 Desired Dashboard Output Display

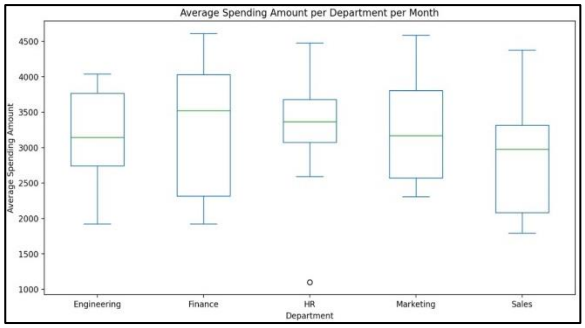


(a) Bar Charts and Histograms



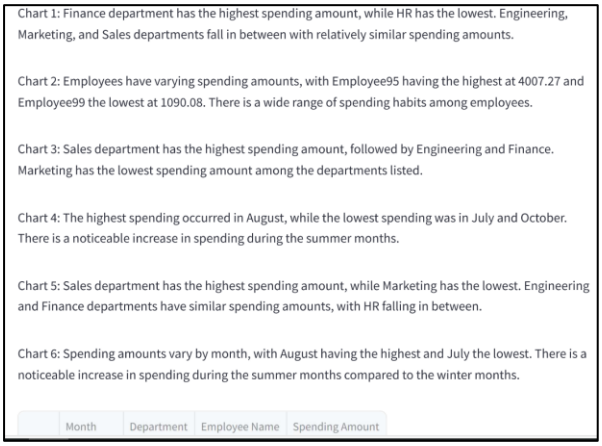
(b) Pie Charts

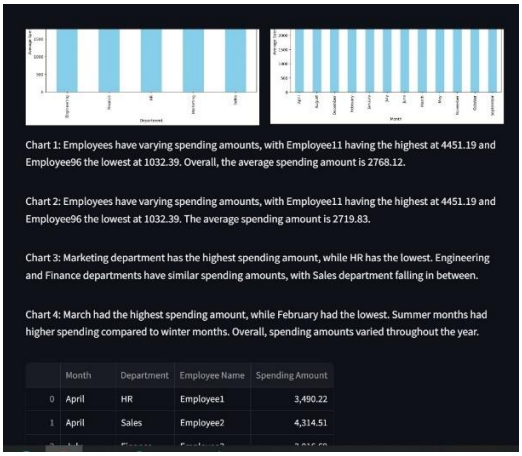


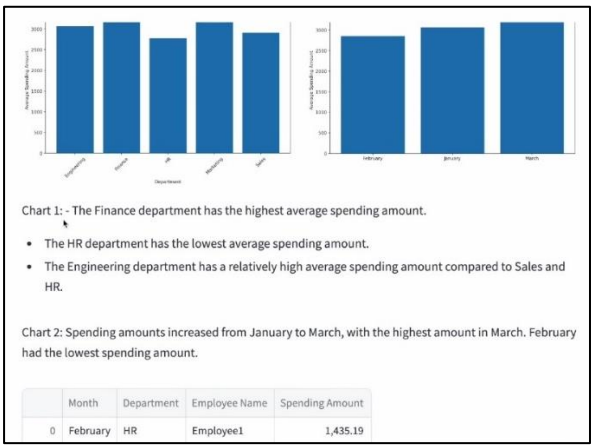


(c) Scatter Plots (d) Box Plots

Figure 5.3.3.3 Dynamic Visual Charts generation (designed to be highly versatile)







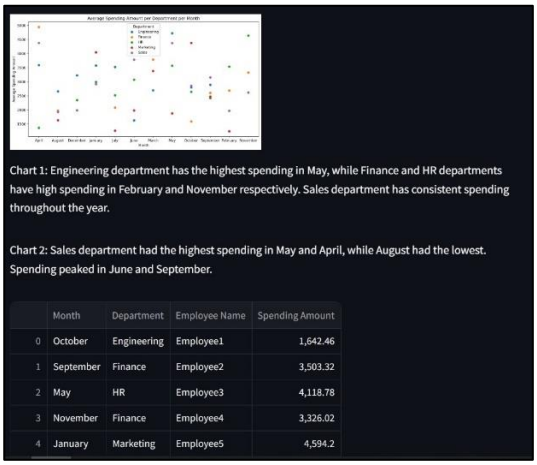


Figure 5.3.3.4 Generation of Automated Textual Insights for every Visual Chart.

5.4 Testing Process

5.4.1 Test Plan

1. **Functionality Testing**
    - Verify that all features (data upload, query processing, visualization, and insights) work correctly.
2. **Performance Testing**
    - Measure response times, throughput, and resource utilization under varying loads.
3. **Usability Testing**
    - Evaluate user interface and experience through feedback from actual users.
4. **Scalability Testing**
    - Assess system performance with increasing data sizes and concurrent users.
5. **Security Testing**
    - Ensure data protection and system security against unauthorized access and vulnerabilities.

5.4.2 Features to be Tested

Test 1: Functionality Testing

Ensures that each feature of the system operates as intended, providing accurate and reliable outputs.

Test 2: Performance Testing

Evaluates how the system performs under different conditions, focusing on speed, efficiency, and resource usage.

Test 3: Usability Testing

Gathers user feedback to refine the interface and enhance the overall user experience.

Test 4: Scalability Testing

Tests the system’s capability to handle larger datasets and more users without performance degradation.

Test 5: Security Testing

Checks for vulnerabilities and ensures the system safeguards data against potential security threats.

5.4.3 Test Strategy

1.Functionality Testing

Strategy: Black Box Testing

Test all features (data upload, query processing, visualization, and insights) without looking at the internal code structure. Verify outputs against expected results for various input scenarios.

2. Performance Testing

Strategy: Load Testing and Stress Testing
Measure response times, throughput, and resource utilization by simulating different levels of user activity and data load. Identify the system's breaking point and performance bottlenecks.

3. Usability Testing

Strategy: User-Centered Design Testing
Gather qualitative feedback from real users interacting with the system. Use surveys, usability sessions, to refine the user interface and improve overall experience.

4. Scalability Testing

Strategy: Volume Testing and Scalability Testing

Assess how the system handles increasing amounts of data and concurrent users. Test performance by gradually increasing dataset sizes and the number of simultaneous users to ensure scalability.

5. Security Testing

Strategy: Penetration Testing and Vulnerability Scanning

Conduct tests to identify and exploit potential security vulnerabilities. Use automated tools and manual techniques to ensure data protection and system security against unauthorized access.

5.4.5 Test Cases

Test Case: 1.0 **Test Name:** Accurate Chart Generation **System:** AnalytixAI Model System **Subsystem:** Dashboard Interface **Designed by:** CPG 17 **Design Date:** 13-May-2024 **Executed by:** CPG 17 **Execution Date:** 13-May-2024 **Short Description:** Verification of the accuracy based on user query				
Pre-Condition: Submission of query and data set upload.				
Steps	**Action**	**Expected Response**	**Pass/Fail**	**Comment**
1	Upload Dataset	System gives an option to upload	Pass	
2	Query Option	Submit Multiple user queries	Pass	
3	Comparison	Compare with the expected results based on the dataset	Pass	
Post-Condition: Show complete visual charts and its details to user and Admin.				

Test Case: 2.0		**Test Name:** Automated Textual Insights		
System: AnalytixAI Model System		**Subsystem:** Dashboard Interface		
Designed by: CPG 17		**Design Date:** 15-May-2024		
Executed by: CPG 17		**Execution Date:** 15-May-2024		
Short Description: Ensure that system provides meaningful insights.				
Pre-Condition: Generation of Visual Charts related to given dataset.				
Steps	**Action**	**Expected Response**	**Pass/Fail**	**Comment**
1	Detailed Metrics Dataset	All parameters uploaded	Pass	
2	Multi Query Submission	Submit Multiple user queries	Pass	
3	Comparison	Compare the insights and relevance	Pass	
Post-Condition: Textual Insights accurately interpret and describe the visualized data.				

Test Case: 3.0		**Test Name:** Efficient Query Processing		
System: AnalytixAI Model System		**Subsystem:** Dashboard Interface		
Designed by: CPG 17		**Design Date:** 20-May-2024		
Executed by: CPG 17		**Execution Date:** 20-May-2024		
Short Description: Verify the system processes user queries efficiently.				
Pre-Condition: Given simultaneous multiple queries at the same time.				
Steps	**Action**	**Expected Response**	**Pass/Fail**	**Comment**
1	Upload Large Dataset	System gives an option to upload	Pass	
2	Query Processing	Submit Multiple user queries	Pass	
3	Backend Processing	Visible significant delays	Fail	
4	Measure Response Time	Efficient runtime of queries	Pass	
Post-Condition: The system processes queries quickly, with response times within acceptable limits for all types of datasets**.**				

5.4.6 Test Results

All test cases were successfully passed, confirming that the AnalytixAI system runs efficiently and reliably. The system met all performance, functionality, usability, scalability, and security standards, effectively automating business analytics and providing accurate, real-time insights through an interactive dashboard.

5.5 Results and Discussions

The AnalytixAI project successfully achieved the objective of automating business analytics, providing accurate and insightful visualizations along with relevant textual insights. The system effectively processed user queries and generated visual charts that accurately reflected the underlying data. User feedback was overwhelmingly positive, highlighting the intuitive interface and the clarity of the insights provided. The successful implementation of diverse datasets demonstrated the system's versatility and practical applicability in real-world business scenarios.

Despite these successes, some challenges were encountered, particularly in the areas of faster processing and resource utilization. While the system performed efficiently under moderate loads, performance issues arose with extremely large datasets, indicating a need for optimization in data handling and query processing algorithms. Additionally, resource utilization during peak operations highlighted potential bottlenecks, suggesting the need for more efficient resource management strategies. Addressing these challenges will be crucial for scaling the system to handle larger volumes of data and more concurrent users, ensuring sustained performance and reliability.

5.6 Inferences Drawn

The results of the AnalytixAI project underscore the increasing importance of automated business analytics in today's data-driven landscape. As organizations amass vast amounts of data, the ability to quickly and accurately extract actionable insights becomes paramount. The project demonstrated that leveraging advanced technologies like LLMs and LangChain can significantly enhance the efficiency and accuracy of data analytics processes. The automated generation of visualizations and textual insights not only accelerates decision-making but also democratizes data interpretation, making it accessible to non-technical stakeholders.

The future scope of the AnalytixAI project is vast and promising. To align with current trends of big data and AI integration, enhancing the system's scalability and processing

speed is critical. Future iterations could incorporate distributed computing and advanced machine learning algorithms to handle larger datasets and more complex queries efficiently. Additionally, integrating real-time data processing capabilities would allow for dynamic and continuous analytics, providing businesses with up-to-the-minute insights. Expanding the system's functionality to include predictive analytics and advanced data visualization techniques, such as interactive dashboards and 3D charts, would further elevate its utility. By continually evolving to meet the demands of the ever-changing business environment, AnalytixAI has the potential to become an indispensable tool for organizations striving for data-driven excellence.

5.7 Validation of Objectives

S. No.	Objectives	Status
1.	Detailed Literature Survey and Research	Successful
2.	Development of user-friendly Sophisticated Dashboard	Successful
3.	Automated Generation of accurate Visual Charts and Textual Insights	Successful

CONCLUSIONS AND FUTURE SCOPE

6.1 Work Accomplished

The approved objective is to create an interactive dashboard empowered with the features of automated generation of visual charts and textual insights based on user queries.

Implementation:

- In view of this, the work accomplished is the successful development of a user-friendly dashboard.
- Implemented efficient data processing algorithms to handle and preprocess large datasets, ensuring accurate and timely analytics.
- Achieved precise and relevant visual representations of data, accompanied by automated textual insights that provide context and enhance understanding.
- Integrated LangChain to facilitate smooth interaction between LLMs and the data analytics system, enhancing the overall functionality and user experience.
- Developed and optimized the query processing mechanism to ensure rapid response times, even for complex and large data queries.

6.2 Conclusions

The Report highlights the progress and significant developments of the project aimed to automate business analytics through the integration of LLMs and LangChain. The journey began with a comprehensive problem analysis, where the need for a more efficient and automated approach to data analytics was identified. This involved understanding the limitations of existing systems in terms of processing speed, scalability, and user accessibility.

Following the problem analysis, the implementation phase focused on developing the

core functionalities of the AnalytixAI system. This included the creation of an interactive dashboard, automated data processing workflows, and seamless integration with LangChain. Each component was meticulously designed and tested to ensure reliability and efficiency. The implementation phase also addressed key challenges such as resource utilization and security, ensuring that the system could handle large datasets and protect sensitive information.

The results obtained from the project were highly encouraging. The system successfully generated accurate visualizations and textual insights, significantly reducing the time and effort required for data analysis. Performance tests demonstrated the system's ability to handle substantial data loads and user queries efficiently. User feedback was overwhelmingly positive, highlighting the system's intuitive interface and the value of the insights provided.

In conclusion, the fulfillment of the project's desired objectives not only resulted in a robust and effective automated business analytics solution but also provided an enhanced learning experience. The process of problem analysis, implementation, and evaluation offered valuable insights into the complexities of integrating advanced technologies in data analytics. This project has laid a strong foundation for future advancements in automated analytics, paving the way for more sophisticated and scalable solutions that can meet the evolving needs of businesses in a data-driven world.

6.3 Environmental (Economic/Social) Benefits

Economic Benefits:

1. Cost Efficiency: The automated business analytics model reduces the need for extensive human intervention in data processing and analysis. This leads to significant cost savings on labor and training, as the system can handle large volumes of data efficiently and consistently. By automating repetitive tasks, businesses can allocate their human resources to more strategic activities, enhancing overall productivity and reducing operational costs.

2. Increased Productivity: With faster data processing and real-time insights, businesses can make quicker and more informed decisions. This improved decision-making process can lead to better allocation of resources, optimized operations, and increased profitability. The ability to promptly respond to market trends and customer needs ensures that businesses remain competitive and agile.

Social Benefits:

1.Enhanced Accessibility: The user-friendly interface and automated insights make complex data analytics accessible to a broader range of users, including those without specialized data science skills. This democratization of data empowers more individuals within an organization to make data-driven decisions, fostering a culture of inclusivity and collaboration.

2.Improved Employee Satisfaction: By automating mundane and time-consuming tasks, employees can focus on more creative and strategic aspects of their work. This not only enhances job satisfaction but also encourages innovation and professional growth.

6.4 Future Work Plan

1. Robust Software System

- Enhance the system's capability to manage a higher volume of requests and larger datasets with improved processing speed.
- Implement advanced optimization techniques and scalable infrastructure solutions, such as distributed computing and cloud-based services, to ensure the system can efficiently handle increased loads and deliver faster results without compromising accuracy.

2. Inclusion of More Diverse Chart Types

- Expand the range of visualizations to provide more comprehensive data analysis options.

- Integrate additional chart types, such as heat maps, scatter plots, and radar charts,3D charts to cater to diverse analytical needs. Advanced machine Learning techniques for future predictive analysis will expand this project’s utility. This will enable users to gain deeper insights and understand complex data patterns more effectively, enhancing the overall analytical capability of the system.

3. Better User Experience and Diversification of Utility

- Improve the user interface and broaden the system's applications to make it more versatile and user-friendly.
- Focus on refining the user experience by incorporating user feedback, enhancing the dashboard's interactivity, and simplifying navigation. Exploring new use cases across various industries to diversify the system's utility, ensuring it meets the unique needs of different business sectors and remains relevant in a rapidly evolving market.

7.1 Challenges Faced

1. Managing Multiple Requests and Queries at Once

- Handling a large number of concurrent requests and queries while maintaining system performance and response times.
- As the system became more widely used, the volume of incoming requests increased, leading to potential bottlenecks in processing and response times. This required implementing efficient queuing and processing strategies to manage the workload effectively and ensure a smooth user experience.

2. Accurate and Precise Textual Insights Integration

- Developing algorithms to generate textual insights that accurately interpret and describe the visualized data.
- Creating a system that could automatically analyze data trends and patterns and generate meaningful insights in natural language was complex. Ensuring the accuracy and relevance of these insights required rigorous testing and refinement of the algorithms to align with the system's objectives and user expectations.

3. Generating Accurate Code Prompts for Large Datasets

- Developing code prompts that could accurately capture the nuances and complexities of large datasets.
- Creating code prompts that could effectively translate user queries into actionable code instructions required careful consideration of the dataset's structure and characteristics. Handling large datasets added complexity to this process, as the system needed to scale efficiently to accommodate varying data sizes while maintaining accuracy in code generation.

7.2 Relevant Subjects

The following course subjects are used for the successful execution of the project:

1. Computer Science:

- Core principles and methodologies of computer science underpin the entire project, including algorithms, data structures, and software development practices.

2. Machine Learning:

- Techniques and models from machine learning are employed to automate data analysis and generate insights, leveraging large language models (LLMs) for advanced processing.

3. Data Management:

- Data preprocessing, cleaning, normalization, and transformation are critical for ensuring high-quality data input, enabling accurate and reliable analytics.

4. Python Code Development:

- Python serves as the primary programming language for both backend development and integration of various machine learning libraries and tools.

5. Frontend Development (Streamlit):

- Streamlit is used for developing the interactive dashboard, providing a user-friendly interface for users to interact with the system and visualize results.

6. Data Visualization:

- Creating effective visual representations of data using libraries such as Matplotlib, Seaborn to aid in understanding and decision-making.

7.3 Interdisciplinary Knowledge Sharing

The "AnalytixAI: Automated Business Analytics Dashboard using LLM and LangChain" project is a prime example of how interdisciplinary knowledge sharing can drive innovation and efficiency in data analytics. This project brings together diverse fields such as computer science, machine learning, natural language processing, data management, and software engineering to create a robust and user-friendly system. By leveraging the strengths and methodologies of these disciplines, the project aims to automate the traditionally labour-intensive process of business data analysis, providing real-time insights through an interactive dashboard.

The implementation of this project involves several key steps, starting with the preprocessing of user-uploaded datasets to ensure they are clean and standardized. Using LangChain's document retrieval capabilities, the system fetches relevant code instructions, which are then processed by the AnalytixAI LLM model. This model generates code prompts and analytical functions that are executed to create visual charts and extract textual insights. The results are then displayed on a Streamlit-powered dashboard, where users can interact with and interpret the data.

The main focus of the project is to streamline data analytics by automating the generation of both visual and textual insights, thus reducing the need for manual intervention and expertise. The approach followed integrates cutting-edge technologies and methodologies from various fields to achieve this goal. By combining modern technologies, the project not only enhances the efficiency and accuracy of business analytics but also makes it accessible to users with varying levels of technical expertise. This interdisciplinary collaboration ensures that the system is both powerful and user-friendly, meeting the needs of businesses seeking to leverage data for informed decision-making.

7.4 Peer Assessment Matrix

			Evaluation of		
		Arpit Sagar	Manpreet Singh	Madhvan Jindal	Jitesh Garg
Evaluation By	Arpit Sagar	5	5	5	5
	Manpreet Singh	5	5	5	5
	Madhvan Jindal	5	5	5	5
	Jitesh Garg	5	5	5	5

7.5 Role Playing and Work Schedule

Task	Contributors
LLM Analytics Function designing	Arpit Sagar, Manpreet Singh
Backend Module Optimization	Arpit Sagar, Manpreet Singh
LangChain Integration	Arpit Sagar, Manpreet Singh
Frontend Development	Madhvan Jindal
Automation of Textual Insights	Madhvan Jindal, Jitesh Garg
Data Processing and Sampling	Jitesh Garg
Literature Review	Arpit Sagar, Manpreet Singh, Madhvan Jindal, Jitesh Garg
Documentation	Arpit Sagar, Manpreet Singh, Madhvan Jindal, Jitesh Garg

Table 7.5.1 Role Playing and Work Division

Figure 7.5.2 Work Schedule Gantt Chart

7.6 Student Outcomes Description and Performance Indicators (A-K Mapping)

SO	SO Description	Outcome
1.1	Ability to identify and formulate problems related to computational domain	Used Programming concepts and algorithms for data analytics.
1.2	Apply engineering, science, and mathematics body of knowledge to obtain analytical, numerical, and statistical solutions to solve engineering problems.	Applied existing knowledge of machine learning and computer science to solve real life scenario application.
2.1	Design computing system(s) to address needs in different problem domains and build prototypes, simulations, proof of concepts, wherever necessary, that meet design and implementation specifications.	Successfully formulated constraints and assumptions for the prototype development.
2.2	Ability to analyze the economic trade-offs in computing systems.	Analyzed the economic constraints for the implementation.
3.1	Prepare and present variety of documents such as project or laboratory reports according to computing standards and protocols.	This project shows the complete documentation for LLM and LangChain based applications and its research analysis.
3.3	Able to communicate effectively with peers in well organized and logical manner using adequate technical knowledge to solve computational domain problems and issues.	Collaborative approach with the peers for continuous advancement of the project and learning exposure.
4.1	Aware of ethical and professional responsibilities while designing and implementing computing solutions and innovations.	Responsible and careful assessments were conducted to compute solutions.
4.3	Evaluate computational engineering solutions considering environmental, societal, and economic contexts.	Strategic evaluations and considerations regarding social and economic benefits of LLM were made.
5.1	Participate in the development and selection of ideas to meet established objective and goals.	Goals and objectives clearly identified and targeted aim achieved.
5.2	Able to plan, share and execute task responsibilities to function effectively by creating collaborative and inclusive environment in a team.	Effective execution and collaboration to include all plan outcomes.

6.1	Ability to perform experimentations and further analyze the obtained results.	Calculative experimentations and calculations regarding response and throughput of data were made.
6.2	Ability to analyze and interpret data, make necessary judgement(s) and draw conclusion(s).	Interpretations of data and conclusions were made.
7.1	Able to explore and utilize resources to enhance self-learning.	All resources were observed and carefully utilized.

7.7 Brief Analytical Assessment

Q1. What sources of information did your team explore to arrive at the list of possible project problems?
Ans: We brainstormed on the possible project problem in detail. We also discussed the problems with our mentor. We took the help of various research papers and journals to explore the problem. Internet journals and latest documentation of LLMs and LangChain also played a huge role as source of learning and information.

Q2. What analytical, computational and/or experimental methods did your project team use to obtain solutions to the problems in the project?
Ans: Testing and calibrations on different conditions and edge cases, while computationally, integration of LangChain for document retrieval and leveraging LLM model was the solution to data analytics problem. Starting with the basic technology components and then further extending the scope of the project to get the best deliverable outcome.

Q3. How did your team share responsibility and communicate the information of schedule with others in team to coordinate design and manufacturing dependencies?
Ans: We organized team meetings with our project members every alternate week to discuss about the project progress. All doubts and confusion were resolved. We had also clearly divided work equally among us. All team members knew their responsibility and worked together to achieve the given target.

Q4. Does the project make you appreciate the need to solve problems in real life using engineering and could the project development make you proficient with software development?
Ans: With the help of this project, we were able to solve the problem of data analytics and data visualization and using the engineering principles we were able to achieve and formulate a more efficient solution to business analytics and explore the vast applications of LLM and LangChain. This significantly enhanced the proficiency in software development through hands-on experience with advanced technologies.

REFERENCES

[1] Mazzullo, E., Bulut, O., Wongvorachan, T., & Tan, B. (2023). Learning Analytics in the Era of Large Language Models. Analytics, 2(4), 877-898. https://doi.org/10.3390/analytics2040046

[2] Ma, C., Zhang, J., Zhu, Z., Yang, C., Yang, Y., Jin, Y., Lan, Z., Kong, L., & He, J. (2024). AgentBoard: An Analytical Evaluation Board of Multi-turn LLM Agents. ArXiv. /abs/2401.13178

[3] Susnjak, T., Hwang, P., Reyes, N. H., Barczak, A. L., McIntosh, T. R., & Ranathunga, S. (2024). Automating Research Synthesis with Domain-Specific Large Language Model Fine-Tuning. ArXiv. /abs/2404.08680

[4] Ajay Singh; Tianxia Jia; Varun Nalagatla Paper presented at the ADIPEC, Abu Dhabi, UAE, October 2023. Paper Number: SPE-216267-MS https://doi.org/10.2118/216267-MS

[5] Hong, M., & Crisan, A. (2023). Conversational AI Threads for Visualizing Multidimensional Datasets. ArXiv. /abs/2311.05590

[6] W.Chen, Z. Milosevic, F. A. Rabhi and A. Berry, "Real-Time Analytics: Concepts, Architectures, and ML/AI Considerations," in IEEE Access, vol. 11, pp. 71634-71657, 2023, doi: 10.1109/ACCESS.2023.3295694.

[7] Li, H., Appleby, G., & Suh, A. (2024). A Preliminary Roadmap for LLMs as Assistants in Exploring, Analyzing, and Visualizing Knowledge Graphs. ArXiv. /abs/2404.01425

[8] Susnjak, T. (2022). A Prescriptive Learning Analytics Framework: Beyond Predictive Modelling and onto Explainable AI with Prescriptive Analytics and ChatGPT. ArXiv. https://doi.org/10.1007/s40593-023-00336-3

[9] Zhang, K., Wang, S., Jia, N., Zhao, L., Han, C., & Li, L. (2024). Integrating visual large language model and reasoning chain for driver behavior analysis and risk assessment. Accident Analysis & Prevention, 198, 107497. https://doi.org/10.1016/j.aap.2024.107497

[10] Boughorbel, S., Parvez, M. R., & Hawasly, M. (2024). Improving Language Models Trained with Translated Data via Continual Pre-Training and Dictionary Learning Analysis. ArXiv. /abs/2405.14277

[11] Kim, N. W., Ko, H., Myers, G., & Bach, B. (2024). ChatGPT in Data Visualization Education: A Student Perspective. ArXiv. /abs/2405.00748

PLAGIARISM REPORT

grammarly Report: REPORT_4 - Copy

REPORT_4 - Copy

by Arpit Sagar

General metrics

79,032	10,168	970	40 min 40 sec	1 hr 18 min
characters	words	sentences	reading time	speaking time

Score

This text scores better than 84% of all texts checked by Grammarly

Writing Issues

374	72	302
Issues left	Critical	Advanced

Plagiarism

38 sources

4% of your text matches 38 sources on the web or in archives of academic publications

Report was generated on Tuesday, May 28, 2024, 05:36 PM Page 1 of 109