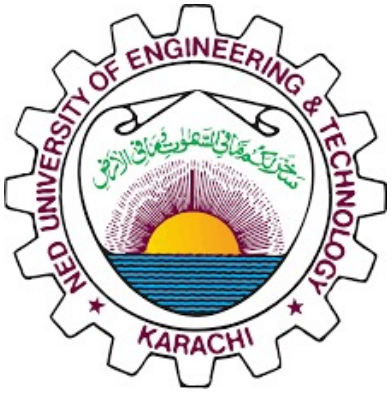
UNDERGRADUATE FINAL YEAR DESIGN PROJECT REPORT

Department of Software Engineering

NED University of Engineering and Technology



Agaahi

Group Number: SE-21008 **Batch: 2021-2025**

Group Member Names:

Khushbakht Khan	SE-21009
Sarah Sami	SE-21026
Syed Aun Muhammad	SE-21036
Moiz Naveed	SE-21048

Approved by:

. . . . . .

Engr. Asma Khan
Assistant Professor
Project Advisor

This page intentionally left blank

Author's Declaration

We declare that we are the sole authors of this project. It is the actual copy of the project that was accepted by our advisor(s) including any necessary revisions. We also grant NED University of Engineering and Technology permission to reproduce and distribute electronic or paper copies of this project.

Khusbakht Khan	*Sarah Sami*	*Aun Muhammad*	*Moiz Naveed*
.....	.....	.....	.....
Khushbakht Khan	Sarah Sami	Syed Aun Muhammad	Moiz Naveed
SE-21009	SE-21026	SE-21036	SE-21048
khan4401804@	sami4401662@	muhammad4403242@	naveed4403529@
cloud.neduet.edu.pk	cloud.neduet.edu.pk	cloud.neduet.edu.pk	cloud.neduet.edu.pk

Statement of Contributions

- **Ms. Khushbakht Khan** has been actively involved in data collection and has contributed to the development of the chat with database page, including API integrations, and implementation of Role Based Access Control (RBAC) to ensure privacy and security of organizations data. She has also worked on multiple static front-end pages and participated in compiling the project documentation.
- **Ms. Sarah Sami** has focused on active research, data preprocessing, training the language model (LLM), and establishing database connectivity. She has worked on integrating charts in chat with database page to ensure smooth communication. She has been managing UML related tasks as well as designs of frontend and contributed to the initial project documentation.
- **Mr. Syed Aun Muhammad** has been responsible for data collection, designing front-end interface components for Dashboard, User management, Role management, Agaahi Admin Panel, and developing preliminary visualization modules. He has also contributed to the creation of static pages of our system and in the preparation of the project report.
- **Mr. Moiz Naveed** has worked on data validation, training the chatbot using different models, and implementing preliminary back-end mechanisms through microservices. He used organization's data cleaned it and created schemas for robust back-end implementation. He has also contributed to the deployment of models and Agaahi interface, as well as compiling sections of the project report.
- **All team members** were actively engaged in analyzing the requirements provided by the industry advisor and ensuring the project aligns with the inventory management schema shared. Together, the team focused on research, design, and implementation of the foundational features, ensuring significant progress in the initial phases of the project.

Executive Summary

Organizations are producing data in phenomenal volumes, but are not getting the desired value out of it because of certain barriers. Barriers being technical complexity, reliance on nearest subject-matter expert, obscure tools with peculiar interfaces and the list goes on. Agaahi bridges the gap between non technical users and the insights hidden in the data. That could make a vast impact, if they were easily accessible.

Agaahi is an AI-integrated analytics automation web application optimized for businesses. It allows users to gain insights from their relational database in natural language, without the hassle of composing in SQL. Using LLM and LangChain, Agaahi autonomously builds, checks, and executes SQL commands, doing so with role-based data shielded privacy (RBAC) access controls to protect confidential data. Furthermore, customizable dashboards consisting of user required charts are prepared without any technical knowledge. Users only need to enter a prompt in natural language briefing the system about the type of chart required. Agaahi is not just a web application, it is a transformative link between data and meaningful insights [7]. By providing users with the capability to gain insights from databases using natural language, providing customizable dashboards and actionable summaries. A business analyst who needs assistance with understanding customer trends, or a team lead who needs operational summaries on the go. Agaahi enables every member of the organization to access required information without needing to know SQL.

Data is no longer constrained by an organizational tool; rather, Agaahi has transformed data accessibility by putting it into a new perception. It shifts the perspective by eliminating technical barriers and allowing users to pose questions in ways they understand. Passive data becomes active insights. Designed with a user first approach and an impenetrable roles-based security structure, Agaahi ensures decisions made are reliable, effortless, and grounded in data. Not only does it balances intuition and power, but marks the beginning of complete egalitarian access to the world of data. Now skilled technicians are no longer gatekeepers to analytics, as automation augments their user experience.

Acknowledgments

We are thankful to Allah for enabling us with the ability and capacity to think, act, and fulfill our responsibilities. Furthermore, we wish to express our gratitude to our supervisor, **Engr. Asma Khan**, and our external supervisor, **Mr. Abdul Basit**, for their continuous guidance and support during our project. Additionally, we acknowledge the significant contribution of our instructors who have provided teaching, mentoring, and assistance throughout our academic endeavors. Our appreciation extends to all the staff members of the department and university who have supported us during our tenure at the institution.

United Nations Sustainable Development Goals

The Sustainable Development Goals (SDGs) are the blueprint to achieve a better and more sustainable future for all. They address the global challenges we face, including poverty, inequality, climate change, environmental degradation, peace, and justice. There are a total of 17 SDGs as mentioned below. Check the appropriate SDGs related to the project.

- ☐ No Poverty
- ☐ Zero Hunger
- ☐ Good Health and Well-being
- ☐ Quality Education
- ☐ Gender Equality
- ☐ Clean Water and Sanitation
- ☐ Affordable and Clean Energy
- ☑ Decent Work and Economic Growth
- ☑ Industry, Innovation, and Infrastructure
- ☐ Reduced Inequalities
- ☐ Sustainable Cities and Communities
- ☐ Responsible Consumption and Production
- ☐ Climate Action
- ☐ Life Below Water
- ☐ Life on Land
- ☐ Peace, Justice, and Strong Institutions
- ☑ Partnerships to Achieve the Goals

F/SOP/FYDP 02/06/00

Similarity Index Report

Following students have compiled the final year report on the topic given below for partial fulfillment of the requirement for Bachelor's degree in Software Engineering.

Project Title Agaahi

S. No.	Student Name	Seat Number
1.	Khushbakht Khan	SE-21009
2.	Sarah Sami	SE-21026
3.	Syed Aun Muhamamd	SE-21036
4.	Moiz Naveed	SE-21048

This is to certify that Plagiarism test was conducted on the complete report, and overall similarity index was found to be less than 20%, with maximum 5% from a single source, as required.

Signature and Date

Engr. Asma Khan
Project Advisor

Table Of Contents

List of Tables

List of Figures

List Of Abbreviations

BI	Business Intelligence
SQL	Structured Query Language
NLP	Natural Language Processing
LLM	Large Language Model
ML	Machine Learning
AI	Artificial Intelligence
DSS	Decision Support Systems
SDG	Sustainable Development Goals
CPAs	Certified Public Accountants
PACE	Planning, Analysis, Communication, Execution
KPIs	Key Performance Indicators
SVM	Support Vector Machine
GPT	Generative Pre-trained Transformer
BERT	Bidirectional Encoder Representations from Transformers
API	Application Programming Interface
UI	User Interface

Chapter 1
Introduction

1.1 Background Information

In this digital era where information is available just a few clicks away, it has a way of deceiving individuals by appearing useful while in reality only a few can detect that the information they are presented with can not be used in its raw form. Data is very versatile; it contains all kinds of information for all kinds of people, that is, if they are able to transform it according to their planned use case. The world where information is power and the need for deriving insights from data is highly crucial for an organization to make swift data-driven decisions. This need leverages organizations to acquire a team of highly skilled and proficient data analysts. As there is an extensive amount of data, inherent in an organization, managing it and deriving insights becomes a tedious task. Hence the presence of data analysts becomes a constant need which comes with high expense. Automating this entire process eliminates the constant need of data analysts to get relevant information regarding organization's data and its trends, instead it provides a system which allows users to query about data in natural language according to their understanding. The system also reduces the requirement of domain knowledge including SQL and other query languages and also minimize the need of Data Analyst. Alongside it can also provide visualized results in the form of charts for better understanding. It also focuses on handling real time data. In the form of various analytics graphs to better understand business market trends to support effective decision making. The system also focuses on generating comprehensive excel reports to understand or view data in details with advance filters with the use of simple natural language prompts. As these models gain ability to understand more nuanced language inputs and produce structured outputs there are interesting possibilities for BI, analytics, reporting. Also, visual aid for data analysis such as graphs and dashboard are generally essential in bookkeeping. Statistical, visualization and/or machine learning analytics models [1].

1.2 Significance and Motivation

The purpose of this project is to make data agile and available for the masses especially non-technical users so they can contribute to give a voice through data. This natural language query vision, coupled with intelligence and dashboarding features (even out of the box) [2] could be transformative across many global verticals including finance, healthcare, retail and education. Here are a few reasons why this is an important project:

Eliminating the need to work with data experts or hard-coding, this system empowers users

to interact with and analyze data directly. This results in a more inclusive environment around business intelligence and analytics. Momently allows users to access structured insights faster than ever — which in turn increases the efficiency of decision-making processes within organizations. The project enhances productivity and usability by providing a human-like interface with tools for customizing the visualization. This prevents wastage of time and more important energy in performing data analysis manually. With Gemini's LLM [3], it can scale for numerous industry datasets — allowing a solution that scales flexibly across large or complex datasets. Since it works by primarily assessing the metadata, the complicated databases are not a problem.

1.3 Aims and Objectives

Main aim is to form an application that empowers the role of data analysts and eliminates the constant need of a specialized team that performs operations to extract insights. Instead it presents this application where anyone can query and ask questions and get natural language responses based on their role and quality of information suitable. The application caters the following features:

1.3.1 Natural Language Querying

Facilitate users in querying databases using natural language. Users receive easy-to-understand natural language responses and can send follow-up prompts to generate graphs and visualize insights, aside from dashboards.

1.3.2 Automated Data Analysis

Automate the process of data analysis to significantly reduce manual effort. The application identifies trends, insights, and patterns efficiently through advanced algorithms.

1.3.3 Real-Time Data Ingestion

Implement mechanisms to support real-time data ingestion, ensuring responses are highly accurate. Dashboards will dynamically display the latest data fields to reflect real-time updates.

1.3.4 Query Lens

Smart Query lens that provides insights in tabular form. We also have query builder that can provide raw query for users like data analyst to support it.

1.3.5 Customizable Dashboards and Visualizations

Provide informative dashboards to visualize key insights and data trends. These dashboards are regularly updated and offer customizable options for users to focus on metrics most relevant to their needs.

1.3.6 Automated Reporting

Generate reports highlighting anomalies, outliers, and critical insights. Reports focus on trends and patterns aligned with organizational objectives and requirements.

1.3.7 Voice Query Support

Integrate voice query support to allow users to interact with the application through speech, offering an alternative to typing and improving accessibility for all users.

1.3.8 Seamless Database Integration

Ensure compatibility with a wide range of organizational databases and data formats. The application provides secure, centralized data access and maintains smooth operations with minimal downtime.

1.4 Methodology

- The agile and incremental approach will be followed throughout the project. Where work will be divided and modules and revisited when a problem occurs. The organization we are collaborating with will be updated on the utilization of their data to diminish their privacy concerns.

- A thorough research will be conducted to understand the workings of natural language querying systems, data visualizations, comprehensive reporting, and generative AI [4]. This research aims to gather information on system architecture to generate valuable insights for organizations. Furthermore, past work related to Traditional Models, Machine learning models and deep learning models is analyzed to keep in view the innovation that needs to be made. Comparative analysis is made easier due to this and the workflow of the system will be finalised. This significantly help in preventing “Re-inventing the wheel” and allows us to use existing technology to build something bigger and better.

- Formal meetings will be organized to review the design mockups and discuss the logic implementations. And to confirm the validity of each step to prevent miswork. Extensive

Figure 1.1: PACE Stages

communication and collaboration will be done in order to eliminate vagueness inherent at any step.Basically following PACE workflow.

- The next step is Data Collection, which is the first step when it comes to training any kinds of Data-Driven Models. The data will be collected from our partner organization which contains an Ecommerce related Database. The collected data will then need to be preprocessing which brings us to the Data PreProcessing Stage. When it comes to data and dealing with data, this process has its own sets of steps that need to be fulfilled in order to move ahead. The collected data needs to be Cleaned which involves checking for inconsistencies, duplicates, null values, outliers etc. After Cleaning, comes transformation, the cleaned data will then need to be transformed into desirable format ADD preprocessing details here.

- The final version of data will be utilized in training the model.

- The Development of the system, as mentioned earlier, will be in modules, Model training will take place in steps like: “Text-to-SQL query” [5], “Databse Query response to Natural language simplified response”, “Data to actionable insights” and “Real time data ingestion”. These together will make up the entire model. Besides these, Front-End and Backend APIs will be developed according to the nature of workflow. Customizable Dashboards will be curated precisely allowing users to type prompts regarding the type of data that needs to be visualized. For Report Generation, the recent graphs on the dashboard will be examined and their insights will be added in the report’s content.

- All developed modules will be integrated into a backend server, and APIs will be created to facilitate seamless interaction with the frontend.

- Application Interface will be created and integration of modules will be done.

Year	2024 – 2025								
Months	Sept	Oct	Nov	Dev	Jan	Feb	Mar	Apr	May
Research									
Data Collection									
Front-end									
Back-end (API's)									
Model Training									
Integration									
Testing									
Documentation									

Figure 1.2: Gantt Chart

- Regression testing will be conducted on the system, and any issues discovered during the testing process will be addressed and resolved. Thorough testing will be conducted prior to the delivery of the project to leverage enough time for fixing the errors found during testing.

1.5 SDG Alignment

The project aligns with several sustainable development goals (SDGs)

1.5.1 DECENT WORK AND ECONOMIC GROWTH

It makes it easier for business owners and employees to make decisions based on data, ultimately helping them perform better by allocating their resources more efficiently, improving productivity with intelligence, and contributing to sustainable economic growth. This natural ability of the tool to democratize data access equalizes the opportunity between smaller businesses and entrepreneurs and their larger competitors, fostering equitable economic growth — and also contributing to better job quality by creating data-driven positions that can do so without fear.

1.5.2 INDUSTRY, INNOVATION AND INFRASTRUCTURE

Using state-of-the-art natural language processing models to automatically convert human-readable queries into SQL statements fosters technological innovation in the analytics and BI industries. It promotes the use of Gartner cool tools like advanced analytics tech, that would hopefully create industry standards for ease and efficiency with data. This in turn helps create resilient data infrastructure for companies, facilitating a wide range of other innovations across e-commerce, retail and beyond.

1.5.3 PARTNERSHIPS FOR THE GOALS

By providing new ways to work with Gemini's state-of-the-art language models within standard commercial SQL databases and visualization tools, this project is truly the embodiment of partnership between AI, data science, and business intelligence. It builds a partnership between the companies that provide technology and those who use it, providing shared value between both parties while developing new products to advance sustainability. The project also promotes cross-sectoral collaboration, with users across domains sharing knowledge

1.6 Target Audience

Agaahi is designed for **small and medium businesses**, and can be used by a wide range of users across the organization, from non-technical business users to data professionals. Its natural language interface allows managers, analysts, and decision makers, who otherwise would not possess any technical skills, to interact with meaningful insights derived from structured datasets without any need to execute SQL commands. At the same time, it offers a simplified means of data exploration to data engineers and analysts, streamlining processes for repetitive queries that would otherwise be cumbersome. Built for teams, institutions, and enterprises, Agaahi serves anyone seeking effortless access to intuitive data interactions that drive decisions without the need for dashboards, reports, or navigating through complex BI software.

1.7 Business Impact

This tool comes with the following set of powerful benefits that are designed for the world of retail and e-commerce:

- **Better Accessibility of Data for Decision Togging** The solution translates natural language queries to SQL, enabling retail and e-commerce professionals to autonomously derive insights that help them make product assortment, sales strategy and promotion decisions. By democratizing access to data, the need for specialized technical resources is

eliminated, thereby accelerating decision cycles and enhancing responsiveness to market trends.

- **Inventory operations optimization** This helps retailers and e-commerce businesses in gaining insights into the inventory levels, sales trends, and customer preferences. Such records allow for easy data queries to see stock availability, product performance reports, customer behavior and forecasts of products in demand leading the users to manage inventory proactively which not only help them mitigate the risks related with overstocking or stockouts but also save costs.

- **Enhanced Customer Insights and Personalization** The tool, by providing the customer data in a digestible form, also assists segmentation and personalizations. This helps companies to provide personalized marketing campaigns, increases customer satisfaction and loyalty which in turn improves customer retention and sales conversion rates.

- **Creating a Competitive Edge with Custom Dashboards** This allows business leaders to quickly see all of the most important KPIs through customizable dashboards created by this tool. As a result, companies in areas like retail and e-commerce gain quicker access to the important metrics surrounding sales growth, customer acquisition and profit margins for more rapid decision making than possible with graphs stored in CPAs or similar products—an edge for competitive advantage in fast-moving markets.

1.7.1 Report Outline

This report contains comprehensive documentation on the project titled as “Agaahi”, first chapter entails the Introduction to the project including background information (problem identification), objectives, scope, methodology including complete workflow. Second section is “Literature Review” Which also includes past/related work, comparisons and advantages. Third Section details the core features of Agaahi, all the functionalities and non-functional properties. The main focus of that section is to explain the system design using a Use Case Diagram and Data FLow Diagram.

1.8 Packages and assets used in project

1.8.1 Frontend Dependencies

No.	Name	Description
1	@ant-design/icons	A rich library of customizable icons designed for use with Ant Design components to enhance user interfaces.
2	@tanstack/react-query	A robust library for managing server-side state in React applications, enabling efficient data-fetching, caching, and real-time updates.
3	@tanstack/react-query-devtools	Developer tools to debug, monitor, and optimize the state managed by React Query in real-time.
4	antd	A comprehensive UI framework for React that provides pre-designed components for building professional web applications quickly.
5	apisauce	A lightweight wrapper for Axios that simplifies HTTP requests and offers built-in error handling and retry functionality.
6	dayjs	A minimalist alternative to Moment.js for handling date and time operations, offering a clean API and improved performance.
7	lodash.debounce	A utility function to delay the execution of a function, especially useful in handling frequent events like user input or resizing.
8	react	A JavaScript library for building fast, dynamic, and reusable user interfaces using a component-based architecture.
9	react-dom	Provides React bindings for rendering React components to the DOM and managing DOM interactions seamlessly.
10	react-drag-drop-files	A React library for implementing drag-and-drop file upload functionality with custom event handlers and styling options.

Table 1.1: Frontend Dependencies

1.8.2 Backend Dependencies

No.	Name	Description
1	@nestjs/axios	Provides a seamless integration of Axios, a promise-based HTTP client, with the NestJS framework for handling API requests.
2	@nestjs/common	Contains core utilities, decorators, and services essential for building modular and maintainable applications in NestJS.
3	@nestjs/platform-express	Facilitates integration of the Express.js framework with NestJS for building HTTP servers.
4	@nestjs/schedule	A scheduling module for NestJS to implement cron jobs and task scheduling within the application.
5	@nestjs/swagger	Integrates Swagger for generating API documentation automatically based on decorators and metadata in NestJS.
6	bcrypt-nodejs	A password-hashing library that provides secure methods for generating and verifying hashed passwords.
7	cache-manager	A flexible and extensible library for managing in-memory and distributed caching to improve application performance.
8	class-transformer	A utility to transform plain objects into class instances and ensure type safety in data transformations.
9	class-validator	A library for validating object properties based on decorators, ensuring data integrity in class-based models.
10	csv-stringify	A library for converting objects or arrays into CSV format, commonly used for data exports.
11	dotenv	A lightweight module for loading environment variables from a ‘.env‘ file into ‘process.env‘ to manage configuration settings securely.
12	fs	Node.js module for handling file system operations such as reading, writing, and manipulating files and directories.

Table 1.2: Backend Dependencies

Chapter 2
Literature Review

2.1 Introduction

The core focus of this project is the innovation between Data Analysis [6] and Generative AI. It allows end-users, which includes members of that organization,to use natural language processing (NLP) to query the database and gain insights, allowing users to interact using either voice commands or handwritten text. The system is expected to generate accurate, relevant and easily understandable responses to user queries. System allows for the ingestion of real time data, so that results are updated with most current information. Furthermore, it provides the visual representation of insights and key trends through charts that assist organizations in making data-driven decisions. Reports involving summary, customer satisfaction and data patterns, are generated to document insights comprehensively.

2.2 Historical Approaches to Data Interaction

Take a journey through the evolution of models for interaction with data, from static reporting approaches to dashboards and then to truly interactive dashboards. Discussing how data interfaces are becoming more user-friendly especially for those who are not technically inclined so that anyone can get insights without specialized skills. Survey of state-of-the-art natural language (NL) querying techniques and their role as a forerunner to conversational data access paradigms like Agaahi.

2.3 Evolution of Data-Driven Models

Evolution of data driven models development over the years have been through several phases on this evolution path and we can trace these changes through different phases.

2.3.1 Decision Support Systems (DSS)

In the beginning, decision support systems were mostly manual and based on rules. These systems were designed for structured data and came with analytical tools to assist users in the decision-making process, but they did not offer flexibility nor any automation. These early systems included business intelligence (BI) tools to help organizations leverage structured data to drive decisions.

2.3.2 Transition to Machine Learning (ML)

Machine learning technologies started to change the game in late 1990s and early 2000s. Predictive analytics became an essential part of data-enabled decision-making. These models learned straight from data and provided more flexibility in decision-making compared to explicit rules. Pattern detection and Prediction capabilities of machine learning models on data, provide deeper insights and make predictions based on data more efficiently, hence transforming industries.

2.3.3 Progression of Large Language Models (LLMs)

In the last few years, large language based architectures have changed the way we acquire and interpret textual data in big language models like GPT, BERT et al [7]. By nature LLMs are built to comprehend and generate human language, so they deliver good results when working with unstructured data like text analysis, sentiment analysis, content generation and decision support tasks. In contrast to conventional ML models, the LLM approach allows for direct integration of knowledge into a LLM by leveraging various human languages and narratives.

2.4 Foundational Theories in Data-Driven Decision Making

Modern data-driven decision-making models are based on several theories:

2.4.1 Statistical Learning Theory

This theory is a foundation for modelling insights, providing stages to balance bias and variance into the models. While designing algorithms that generalise well from data and do not overfit/underfit, it helps in guiding the design of your algorithms such that they make accurate view points.

2.4.2 Cognitive Decision-Making Models

Based on cognitive psychology research, these models explain human information processing and decisions. Insights from these cognitive processes have guided the design of more human-centric, intelligible data-driven models allowing transparent and informed decisions within complex environments.

2.5 Review of Techniques and Methods in Data Analysis

In the data analysis domain, various techniques have become more popular recently. This increase is linked to a particular innovation called Large Language Models (LLMs):

2.5.1 Large Language Models (LLMs)

LLMS are a significant shift in natural language processing (NLP) [8] [9]. They assess big datasets to understand complicated language patterns and create connected, sensible text. LLMs can be used for many different kind of assignments:

- **Text generation:** They're good at writing content like reports and articles
- **Sentiment analysis:** Analyzing feelings behind reviews or social media posts
- **Question answering:** Finding suitable answers from large libraries of text
- **Text classification:** Grouping documents into set groups
- **Decision support:** offering supportive insights based on the boundaries of large loads of unorganized data

LLMs are beneficial for fields that rely on extensive text amounts. Industries such as marketing, customer service, law, and healthcare value them especially. They improve traditional ways of supporting decisions, allowing for more diverse and context-specific suggestions.

2.5.2 Natural Language Processing (NLP) for Business Intelligence

Moreover, while LLMs are the current top of the line for NLP, traditional NLP procedures continue to make a significant contribution to textual data examination. This includes techniques like splitting text into individual words, recognizing specific named entities, tagging words based on their function, among others. Powered by LLMs, NLP helps businesses pull important findings from unstructured data sources like customer feedback, emails, and social media to better decision-making [6].

2.6 Challenges in Implementing Data-Driven Models

Even with all these benefits, there are challenges when putting into practice data-dependent models, particularly those based on LLMs:

2.6.1 Data Quality and Consistency:

The quality of the information entered into data-led systems hugely influences the results they give. Large Language Models (LLMs) though strong, can struggle with flawed, biassed, or inconsistent info. This can tamper with results and cause wrong insights.

2.6.2 Data Privacy and Ethical Concerns:

LLMs raise serious moral concerns, especially about privacy and data security. They train better with loads of data—sometimes personal data. This stirs up debate about consent, who owns data, and the risk of creating harmful or bias content [10].

2.6.3 Model Interpretability:

Understanding how LLMs make decisions is another challenge—it's like trying to see into a black box. This is worrying in areas like health or finance, where it's vital to understand why a certain choice was made.

2.6.4 Handling Unstructured Data:

LLMs are very good at dealing with messy data, but ensuring this data is processed correctly, error-free, and varied is still a problem. They also need to be used with caution in sensitive or niche areas. In these situations, general training might not cover all the details.

2.7 Comparative Studies and Current Gaps in Research

New studies are comparing LLMs with older data systems and finding both strengths and gaps:

2.7.1 Comparison with Traditional BI and Machine Learning Systems

The older systems work best with neat data and struggle with messier, more common data. LLMs shine when it's about language and messy data but also have their struggles, like biases and being hard to understand.

2.7.2 Gaps in Research

As far as research goes, there are gaps:

- **Real time adaptability:** We need LLMs that can adapt better in real-time for dynamic choice-help.
- **Improved training methods:** Training methods must be improved to lessen the biases in LLMs and make them more specialised [11].
- **Ethical frameworks:** We need moral guidelines for using LLMs in sensitive areas (like health or finance).

Feature	Traditional Systems	Machine Learning Models	LLM-Based Approaches
Data Type	Structured data, like in databases or spreadsheets.	Structured and semi-structured data such as data from sensors or time series.	Structured, unstructured, and semi-structured data such as text, pictures, and sound.
Data Volume	Deals with small sets of data.	Handles large datasets but requires pre-processing.	Can effectively handle huge sets of data, especially unstructured text.
Decision Support	Focuses on reporting and providing descriptive analytics.	Predicts future trends, classifies data, and clusters similar data together.	Enables knowledgeable decisions, sentiment analysis, and impactful insights from text.
Adaptability	Static and usually demands frequent manual updates.	Can adapt if retrained using fresh data, but adaptability is still limited.	Highly dynamic and can adapt to new text data without explicit retraining.
Model Complexity	Relatively uncomplicated algorithms and provides detailed reports.	Uses complex algorithms (e.g., neural networks, SVMs).	Relies on exceedingly complex deep learning models (e.g., transformers) [21].
Automation	Limited automation, mainly for reporting.	Moderate automation for tasks like classification.	High automation in generating text, answering questions, and interpreting context.
Interpretability	High transparency in how decisions are made.	Varies; interpretable for models like decision trees, but deep learning models are less transparent.	Resembles ”black-box” behavior, making decisions difficult to explain.

Table 2.1: Comparison of Traditional BI Systems, Machine Learning, and LLM-Based Approaches

2.8 Summary

This chapter looked into the growth of data-led systems, particularly LLMs, and their big effects. We delved into historical approaches, evolution of data models, theories, methods, and the problems in this field, like data quality, moral issues, and understanding of models. Comparison of historical approaches and Current Gaps in Research were also discussed to help understand the need of our system, we reviewed some techniques for data analysis which includes NLP and LLMs. LLMs are a major step forward in studying messy/unstructured data and helping in making business decisions. But problems still exist like real-time adaptability and moral use; all these must be addressed for future growth in data-led models.

Chapter 3
System Design And Description

3.1 Introduction

The text provides an overview of Agaahi's key features and its advanced system. The aim is to showcase how modern technology bolsters traditional methods through the use of Agaahi, especially in areas such as real-time data consumption, generation of insights, personalization options , incorporation of language models [12] and report creation methodologies.

3.2 System Design

3.2.1 UseCase Diagram

- **Company Interface**

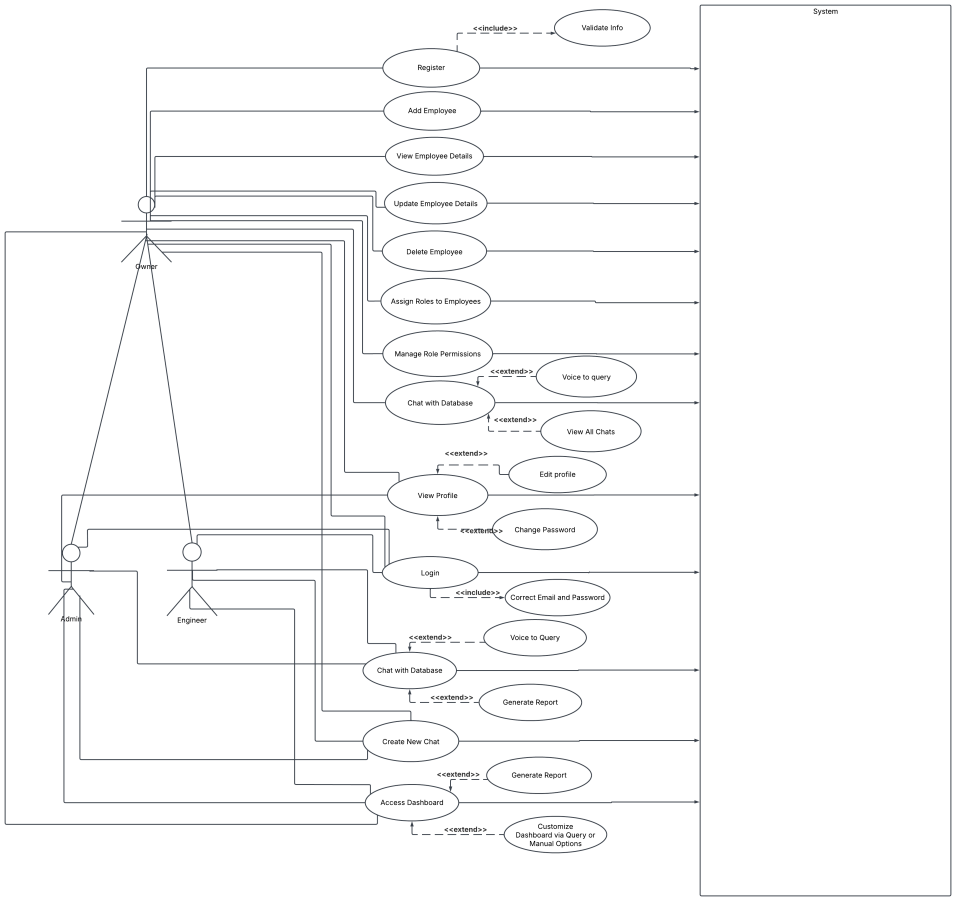


Figure 3.1: Company UseCase Diagram

The use case diagram illustrates the functional relations of the Company Owner, Admin, Employees/Engineers, and the Agaahi system. In this context, both the Company Owner, Admin, and Employees/Engineers interface with different aspects of the system under a hierarchical role-based access control framework. The Company Owner enjoys absolute control, including registering the organization within Agaahi by submitting verified company details, managing employee profiles (adding, deleting, updating, and reviewing personnel data), and bestowing role-based permissions by promoting some employees to Admin and others to Engineer. In addition, the Owner has sole control to monitor all chats of Admins Engineers.

Admins and Engineers gain system access after authenticating using recognized credentials. Everyone is authenticated through Agaahi which safeguards the privacy of sensitive data by ensuring information is concealed based on the user's role. Admins have permission to query advanced data tables, while Engineers are confined to a limited number of predefined tables pertinent to their roles.

Secure login, Chatting with database through text or voice prompts, navigation via dashboard, and report generation are some of the common features that all roles have. Everyone is able to get actionable insights by issuing dynamic text queries against enterprise data or creating visual analytics such as bar graphs and line charts from the summarized data. All actions taken by users are logged in a persistent chat history which can be referenced in the future. With the Generate Report feature, users are able to generate and download comprehensive reports which contain the summaries of the data they have queried. This helps the company make decisions effectively. This feature helps improve operational transparency along with structured security measures and enables collaboration across organizational levels.

Finally, the dynamic and customizable dashboard offers a range of facilities, by allowing users to enter natural language prompts about the nature of graphs they wish to be presented with. Upon prompting, the dashboard displays various graphs that best fits the data of the company.

- **Admin Panel**

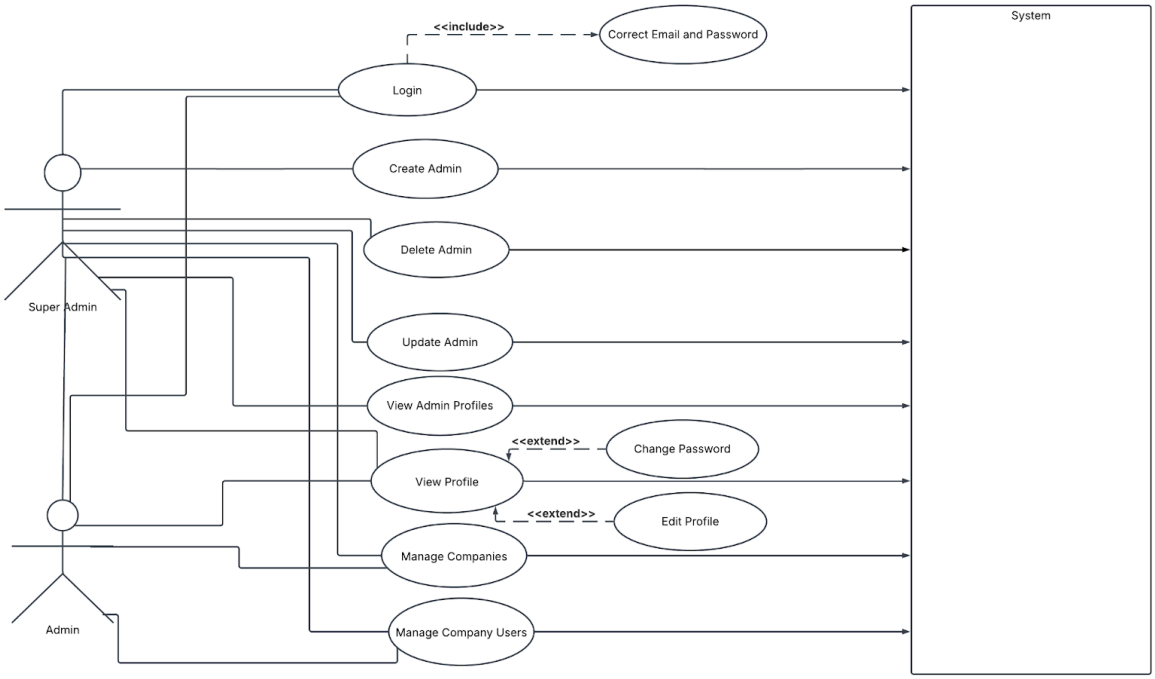


Figure 3.2: Admin Panel Use-Case Diagram

Within the framework of the organization, Super Admins and Admins have critical roles in managing companies and users. Both positions require proper authorization to access the Agaahi Admin Panel using a registered email address and password. The Super Admin possesses higher privileges, granting exclusive right to create, delete and modify Admin accounts and review profile details. Admins and Super Admins share some general functions: both may access and edit profile settings, change passwords, and manage all registered companies and users on the platform. This organizational structure provides proper control where Super Admins exercise control over Admins while allowing shared control over companies and users to maintain balance between functionality and security.

3.2.2 Data Flow Diagram

- **Agaahi Admin Panel**
  LEVEL 0

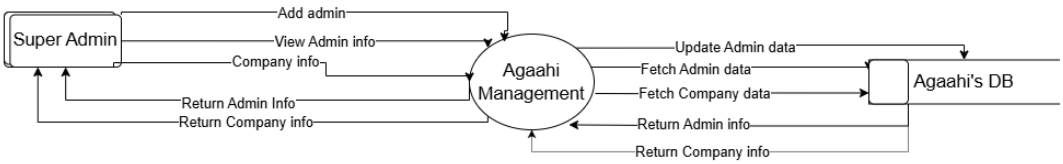


Figure 3.3: Level 0 -Admin Panel

Level 0 Data flow diagram has the highest level of abstraction and it shows the flow of data during the operations performed by Agaahi's Super Admin. A Super Admin can **"Add new Admins"** for which new data will be added to "Agaahi's Database" and if they want to view all Admin's information then that respective data will be fetched from Agaahi's Database and presented to the Super Admin. Similarly if they need to View Company Information (detail of each company and their employees) then Agaahi will fetch that information from Agaahi's Database and present it to the User. This is the basic flow of data for **Agaahi's Admin panel**.

LEVEL 1

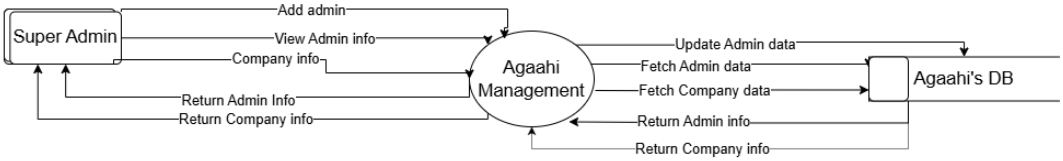


Figure 3.4: Level 1 -Admin Panel

Level 1 Data flow diagram represents the same flow in more detail, in this diagram it is visible that there are Two external entities namely, Super Admin and Admin. A Super Admin can do everything an Admin can, along with a few more operations. An admin can not Add or view "Admin Data" hence it does not have any relations to Admin Management. It can only access Company Management which allows an admin to View Company and Company User Information. All of that data is stored in "Agaahi's Database" and is updated and accessed from the same location.

- **Agaahi Application Data Flow:**

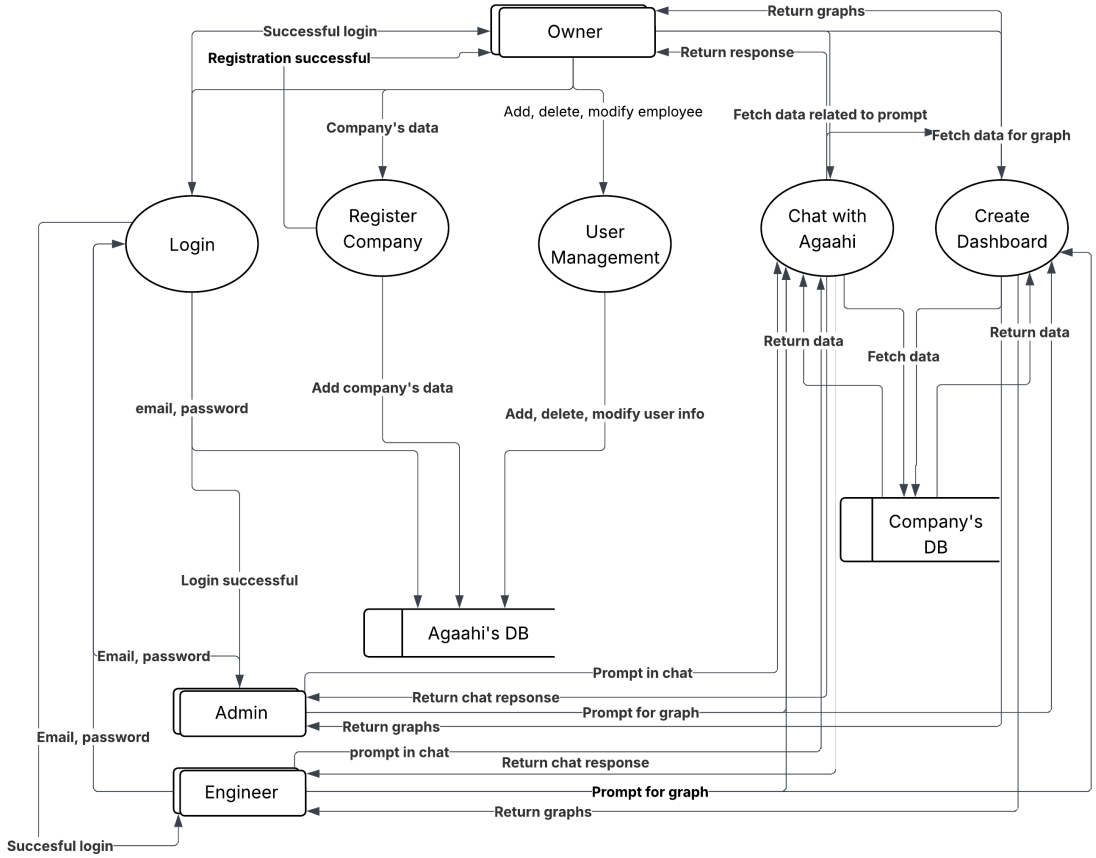


Figure 3.5: Application Flow

This Data Flow Diagram illustrates the flow of Data upon any user accessing the Agaahi application. There are three types of user Categorized on the basis of Role based Access Control, they have different rights. **Owner** being the most privileged and **Admin** and **Engineer** have fewer privileges. Initially, the company Owner is responsible for Registering the company, providing the "connection string" for the company's database and "Adding employees", which basically means the owner will create their accounts automatically when adding employees. The employees will have different roles based on what the owner decides. The employees will not need to Sign up, they can directly Login with the credentials provided by the owner. Upon registering a company, the data related to the company will be stored in "Agaahi's Database" (Agaahi DB).

The owner has access to **Manage Employees**, they can Add, Delete or Modify the employee role, and the data will be stored and updated in Agaahi DB.

Other than that, Chat and Dashboard feature is available for all users of all roles (Owner, Admin, Engineer). The user can prompt the chatbot to answer a certain question and

the model will fetch relevant data from the **"Company's Database"** to formulate a response. Similarly for Dashboard, user can also send a prompt about the data they need to visualize and the model will query the "Company's Database" to generate graphs.

3.3 Administrative Control and User Management

The system has a layered administrative structure to enforce accountability, security, governance, and management policies. At the top. Super Admins who are authenticated using their email and password possess rights to permit them to create, edit, or delete Admin accounts, scrub Admin activity logs, and in conjunction with Admins, control company-wide user and Administrator company-wide user accounts. Super Admins and Admins-aggregate who manage revolving registered users and companies user by viewing profiles.

On the other hand at the company side, the Company Owner has the unilateral right to register and verify the organization within Agaahi, manage employee profiles (creation, updates, and deletions), assign roles (Admin/Engineer), and oversee all activities of users which include chat and system interactions, Password reset, and permission setting at user level.

3.3.1 Security protocols

Security Protocols comprise identity verification employing centralized authentication of all users through Agaahi, tiered supervision (Super Admin oversight to Admins who manage users), and permanent records to audit administrative activity for verification and provide obligation.

3.4 Role-Based Access and User Management

The system employs Role-Based Access Control (RBAC) to limit data access and operational privileges. The Super Admins and Admins oversee user/company level management. Super Admins can modify all accounts and company settings. Company Owner manages admins and engineers, he can also query whole data tables and access company-specific datasets, then we have Admins who have access to query from advanced data tables while Engineers interact to retrieve insights from our system via text or voice prompts to role-specific tables. Authentication through Agaahi enforces dynamic data masking as well as the least privilege principle.

3.4.1 Core functionalities:

Core Functionalities are provided as secure logins per roles, fulfillment of dynamic text and voice queries, and dashboards set up for graph generation through natural language commands, to obtain reports which provide programmable insights and execute verifiable audits of all actions. Security measures put in place restrict role edges to mitigate the risk of privilege escalation, regulatory compliance through audit logs alongside RBAC standards and Encrypt Data at Rest (DAR) considered in future directions.

Role	**Permissions**	**Data Access Scope**
Company	Manage admin and engineer accounts, create admins, retrieve advanced table data.	Execute user management, query whole dataset and use all features.
Admin	Query advanced tables.	Company-specific datasets and settings.
User	Task execution via text or voice prompts.	Predefined role-specific tables.

Table 3.1: Role Definitions and Permissions

3.5 Smart Query Lens

To explore data and generated insights on a deeper level, Agaahi features Query Lens, with its dual interface it provides feasibility to both Technical experts and non technical users. Query Lens provides clear visibility into the queries being generated and executed. It also has prompt history to maintain and ease users by checking previous prompts and queries that were generated prior.

3.5.1 Smart Query:

Smart Query mode, the users can simply enter a question like “Show me the customers and their average order value” and the smart query will generate an output in table format showcasing the customers and their average order values. In short, smart query, provides results in a formatted table for quick and easy perception.

3.5.2 Query Builder:

For users with technical knowledge about querying languages, Query Builder generates proper contextually and syntactically correct SQL queries to a natural language prompt. After a prompt is submitted, the system displays SQL queries along with the prompt.

3.6 Real-Time Data Ingestion and Analysis

Agaahi facilitates real-time inference. This means it processes information as soon as it comes, cutting down on waiting time and letting you understand things immediately. This is super important for groups that have to make fast decisions in fast-changing situations.

3.6.1 Aspects of Approach for Real-Time Ingestion:

Aspect	**Agaahi's Approach**
Data Ingestion	Data is processed continuously as it arrives, with no waiting period.
Data Latency	Instantaneous data processing and real-time insights.
Scalability	Scalable and optimized for high-volume, continuous data streams.
Data Accuracy	Real-time checks ensure data integrity and consistency.

Table 3.2: Agaahi's Approach to Data Processing

3.6.2 Fast Data Handling and Insight Getting: Main Points

Agaahi processes data right away, with no delay. This lets us understand things instantly and in real time. Our system can handle a lot of data all the time and still work fast. We also check the data all the time to make sure it's correct and reliable.

Aspect	**Agaahi's Approach**
Data Analysis	Automated analysis through AI and machine learning models.
Insight Type	Predictive, prescriptive, and actionable insights.
Speed	Instantaneous insights, with the ability to act in real-time.
Accuracy	Highly accurate, data-driven insights using AI models.

Table 3.3: Aspects of Agaahi's Approach for Automated Insight Generation

3.6.3 Customizable Dashboard and Visualization

Agaahi has a special dashboard that you can change how you like. You can make it fit what your job needs or what your business is all about. This way, you get to see only the important numbers and details that really matter to you.

Aspect	Agaahi's Approach
Customization	Fully customizable, tailored to the user's specific needs.
Interactivity	Highly interactive, with drill-downs and real-time filtering.
Data Updates	Live data updates, ensuring that dashboards reflect the most current data.
Ease of Use	User-friendly, designed for non-technical users to create and adjust their dashboards.

Table 3.4: Aspects of Agaahi's Approach for Interactive and Customizable Dashboards

3.6.4 Language Model Integration for Natural Language Prompts

Main Points: Agaahi uses big smart systems like GPT-3 to let people ask questions in simple language and get clear answers from complicated data without having to use special computer languages like SQL [13].

Aspect	Agaahi's Approach
Query Language	Users query data using natural, conversational language.
Accessibility	Accessible to non-technical users, no coding required.
Flexibility	Dynamic and flexible, adapts to a wide range of user queries.
Error Handling	Low risk, as the system interprets and corrects queries.

Table 3.5: Aspects of Agaahi's Approach for Natural Language Querying

3.6.5 Report Generation and Voice-Enabled Prompts

Agaahi has cool tools that make reports by themselves and lets you talk to it so you don't have to use your hands. This makes creating reports super easy and helps people who need to work without using their hands or have special needs to use their devices better.

Aspect	**Agaahi's Approach**
Report Creation	Fully automated, with reports generated based on real-time data.
User Interaction	Voice-enabled prompts allow users to create and modify reports hands-free.
Report Customization	Highly customizable, with dynamic report generation based on user preferences.
Accessibility	Voice-enabled interaction makes it accessible for all users, including those with disabilities.

Table 3.6: Aspects of Agaahi's Approach for Automated and Voice-Enabled Reports

3.7 Summary

Agaahi's features show significant steps forward in working with data compared to old methods:

- Real Time Data Ingestion and Analysis provides more data accessibility and faster responses .
- Meaningful insights help companies in decision-making processes.
- Customizable Dashboards allow people to modify the look and feel of dashboards both manually and through queries.
- Asking questions in natural language enables everyone to access databases and query about data easily, with minimum tech skills required.
- Report Generation provides concise reports making things easier to analyse and help understand work flow better.

Chapter 4
Application Layer and User Interface

4.1 Introduction

This chapter will narrate the practical implementation of Agaahi, through its Frontend and Backend details. The previous chapter focuses majorly on the overall system architecture of Agaahi and its high-level designs, while this one will discuss the actual implementation including, overall UI designs, frontend technologies, backend implementation roadmap, and interaction of components. This section delves into the details of how these components were developed and integrated.

4.2 User Interface Design

4.2.1 Agaahi Web Application

4.2.1.1 Home Page

Agaahi's homepage fully captures the user's attention due to its aesthetic appeal while presenting the organization's vision, features, and team. Its business slogan states, "Pioneering Smarter Data Solutions," which echoes Agaahi's motto, "Transforming raw data into strategic business insights with AI and human intuition."

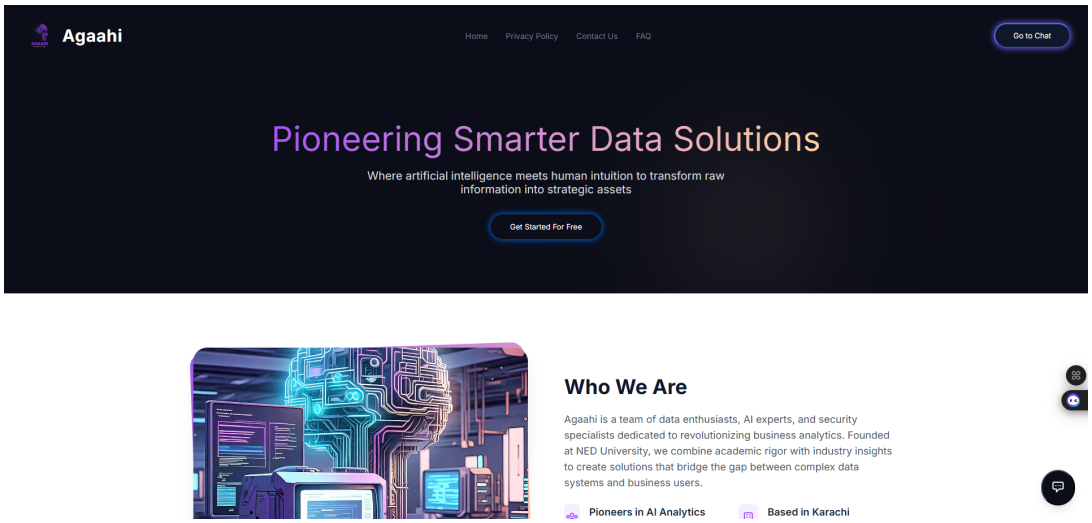
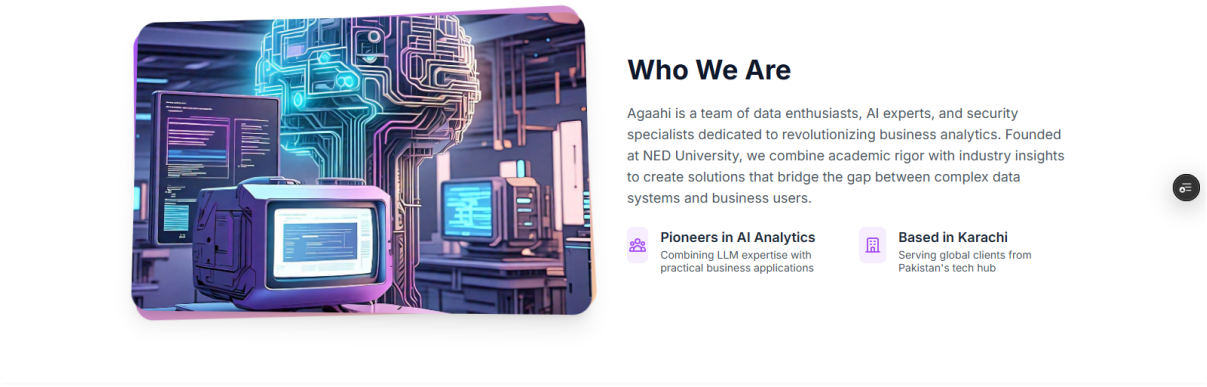


Figure 4.1: Home Screen (1 of 5)

Hero Section Includes a **CTA** – "Get started for free."

Briefly informs of the platform's ambition to integrate the human element with decision making and AI analytics [14].



Home Screen (2 of 5)

About Section – **"Who We Are"**

- Offers a brief recap of her professional biography and personal history as far as Agaahi is concerned.
- Calls out from NED University and emphasizes the focus on AI analytics and data systems.
- Calls out the Karachi based team and their dedication in transforming academic research into business solutions

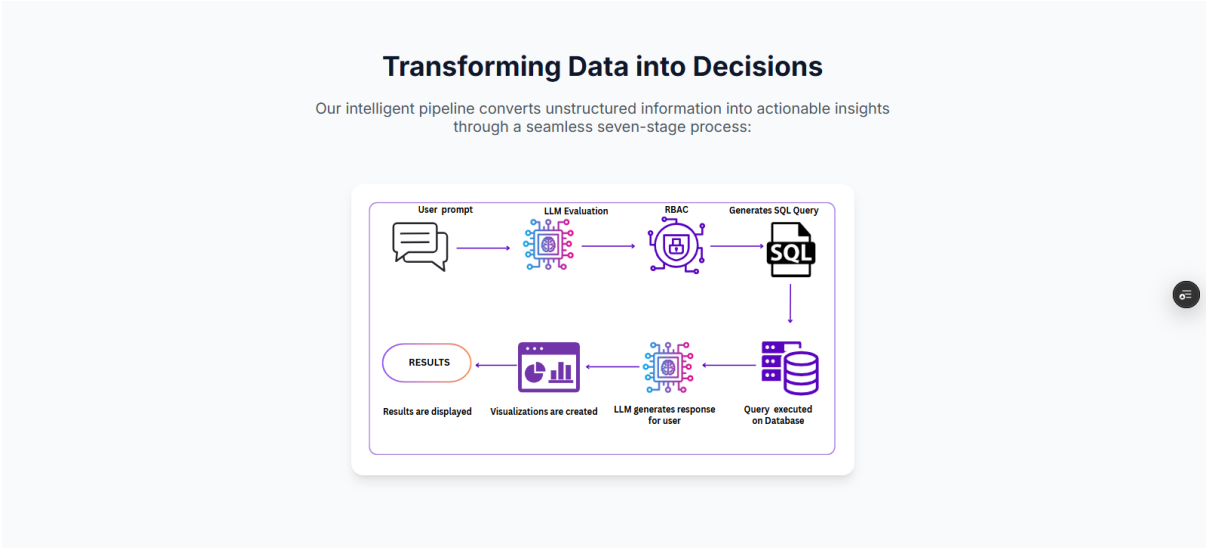


Figure 4.2: Home Screen (3 of 5)

Data Pipeline Overview: In Fig.4.2 Agaahi's intelligent seven stage pipeline is graphically best described by a data flow diagram which begins with unstructured inputs culminating in SQL queries and flows out as visualizations and insights.

The pipeline illustrates the operations of LLMs coupled with role based access control (RBAC) and query execution as they culminate in actionable outputs.

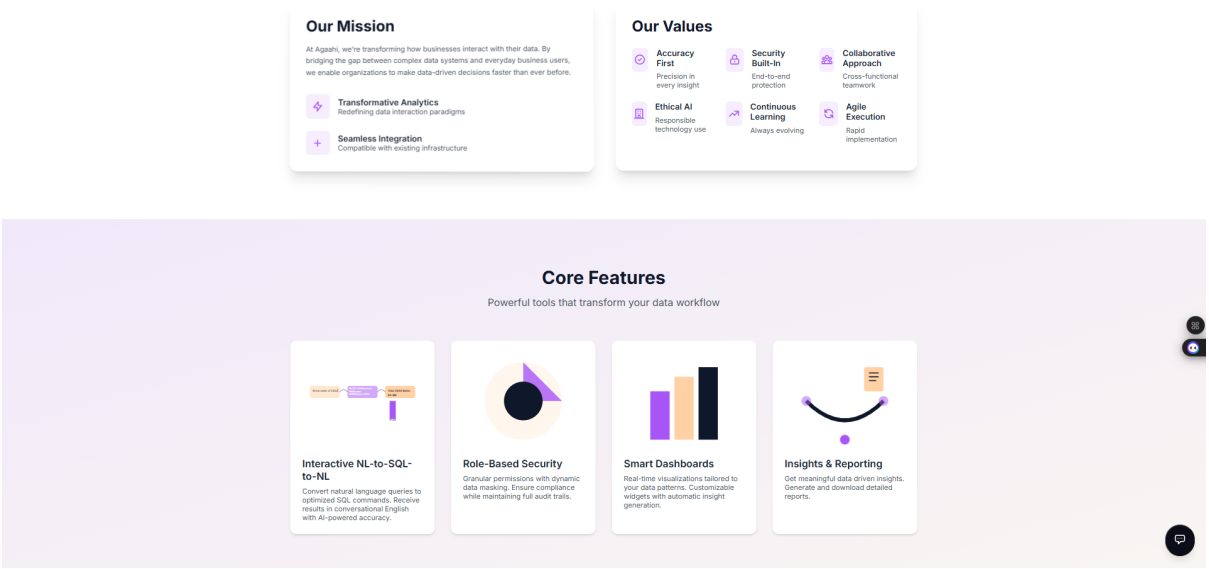


Figure 4.3: Home Screen (4 of 5)

Mission and Values

- **Our Mission** section communicates Agaahi's aim to enable organizations to make faster, more data-driven decisions.
- **Our Values** section outlines the principles behind the product: Accuracy, Security, Collaborative Execution, and Ethical AI.

Core Features Clearly laid out feature cards include:

- Interactive NL-to-SQL-to-NL
- Role-Based Security
- Smart Dashboards
- Insights & Reporting

Accompanied by performance stats like "70% reduction in IT dependency" and "1000+ daily insights generated."

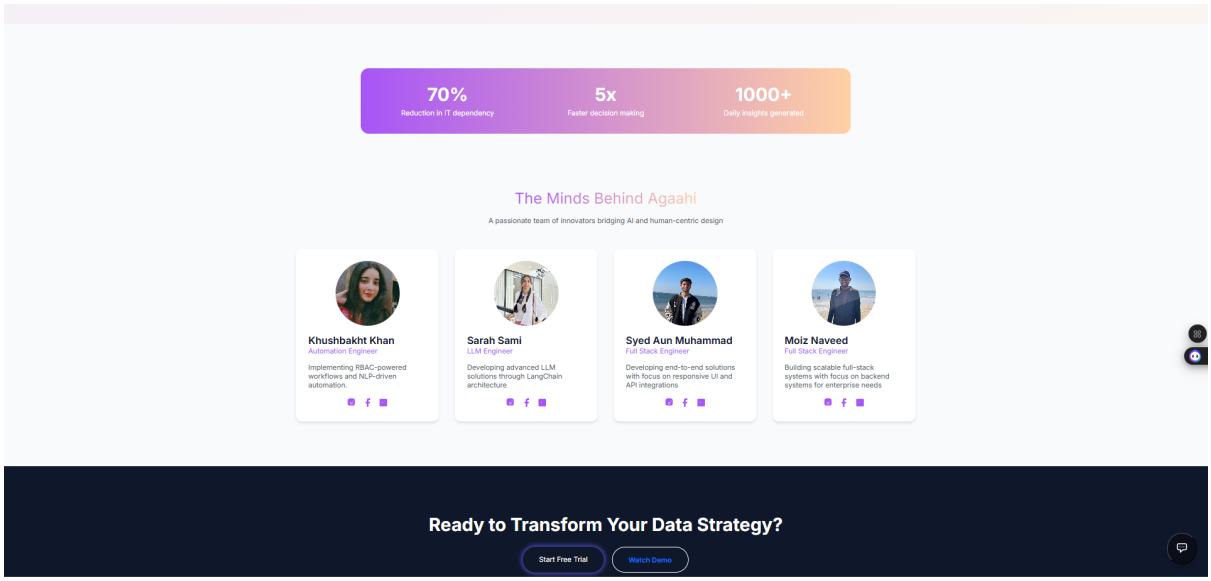


Figure 4.4: Home Screen (4 of 5)

Team Introduction

- The Minds Behind Agaahi section introduces key team members with profile images, roles, and areas of contribution
- Highlights the expertise across automation, LLM integration, API development, and full-stack engineering.

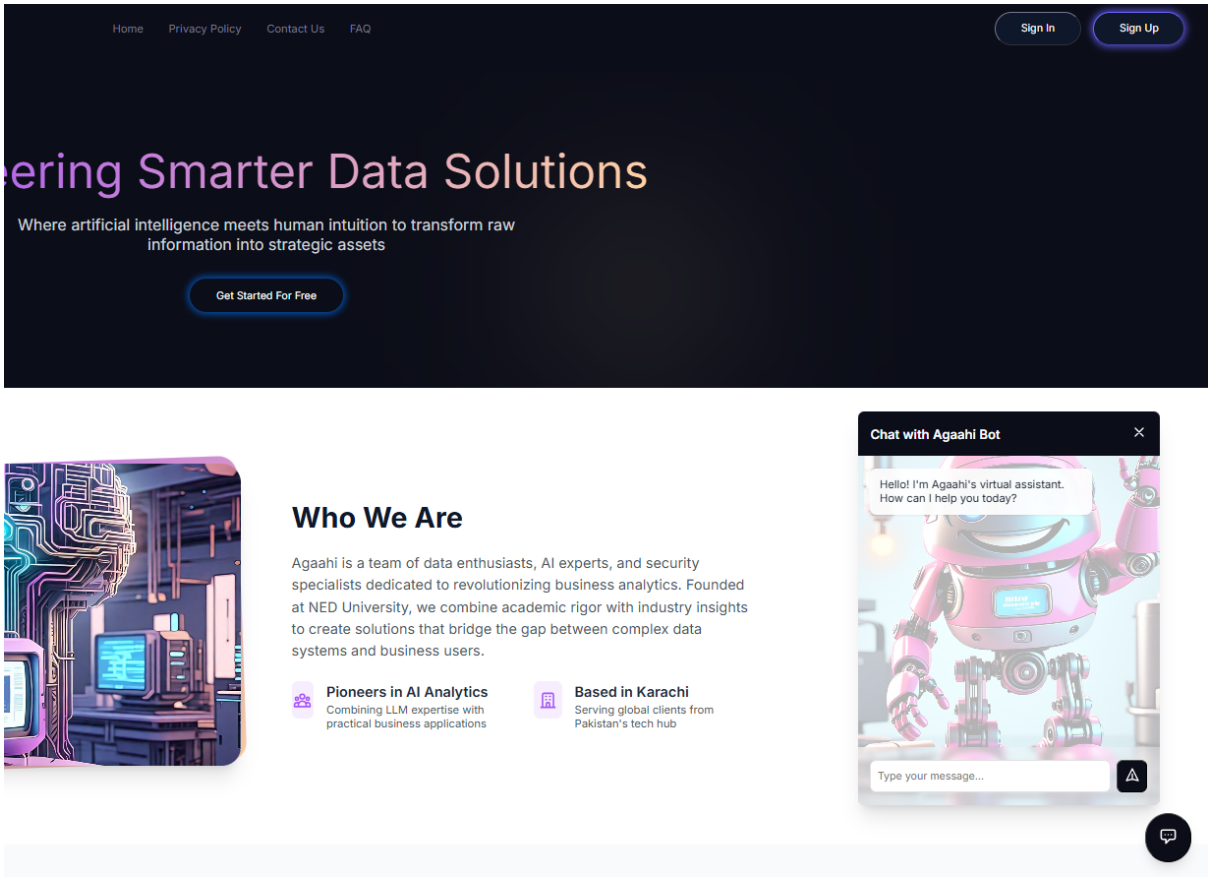


Figure 4.5: Home Screen Chatbot

Chatbot for Product Guidance A chatbot is integrated on the home screen to assist users with generic queries about the product. It provides immediate access to key information, including platform capabilities and usage guidance. Additionally, it offers contextual meta information to enhance user onboarding and engagement.

4.2.1.2 Company Registration

The registration interface allows businesses to create their accounts on the Agaahi platform. The form captures essential information including first name, last name, business name, email, country code, phone number and password. The interface features a clean, user-friendly design with clear input fields and validation. The left side of the page showcases various data analytics visualizations including bar charts, pie charts, line graphs, and progress indicators, highlighting the platform's analytical capabilities. This visual representation emphasizes the platform's core functionality of providing data-driven insights through prompts." This description covers both the functional aspects of the registration form and connects it to the broader purpose of your data analytics platform, while remaining concise and focused.

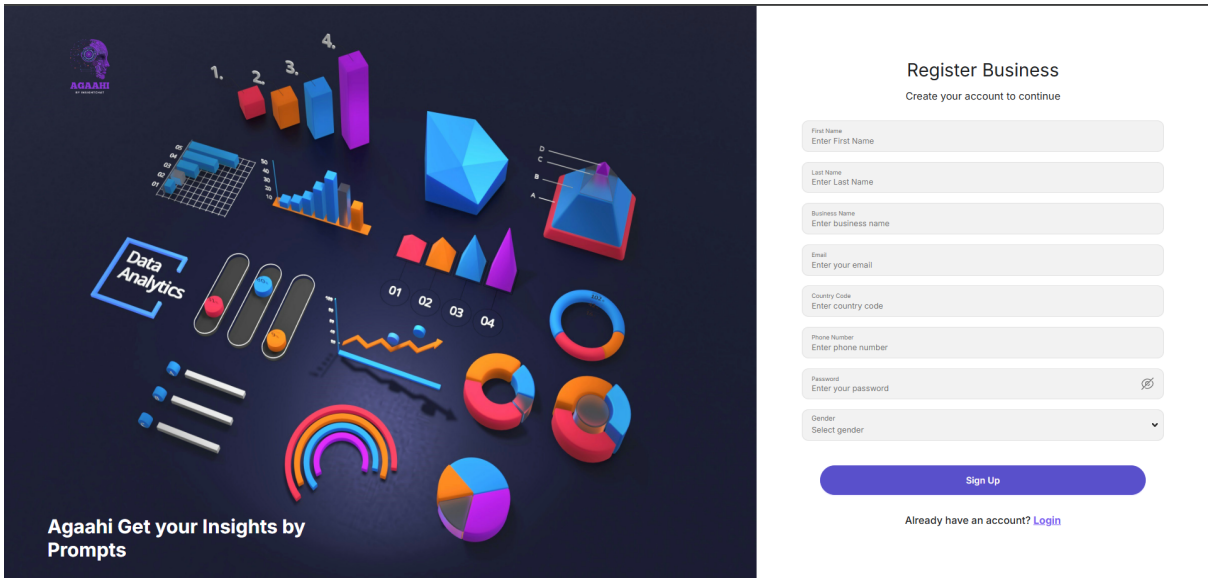


Figure 4.6: Company Registration Screen

4.2.1.3 Database Configuration

Agaahi currently offers support exclusively for MySQL and PostgreSQL databases. To successfully complete the registration and onboarding process, organizations are required to configure their preferred supported database and establish a secure connection with the Agaahi system. This ensures seamless integration, enabling the platform to query and analyze organizational data effectively.

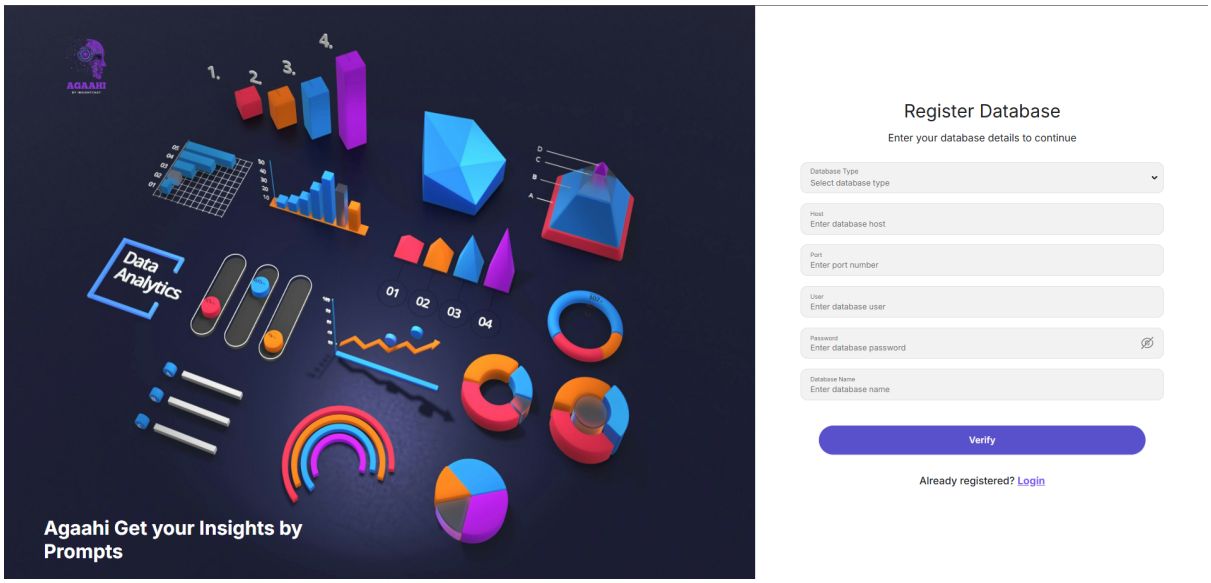


Figure 4.7: Database Configuration Form

4.2.1.4 Login Screen

Once a company is created, the individual who registers it is designated as the owner. The email address used during the registration process is automatically associated with the company and added as an employee account. The owner can then log in to the system using the same credentials via the login screen.

Figure 4.8: Login Screen

4.2.1.5 Forgot Screen

If the owner or any other employee of the company forgets their password, they can easily initiate the password recovery process. By selecting the ”Forgot Password” option, they will receive a verification code via email, allowing them to securely reset and create a new password.

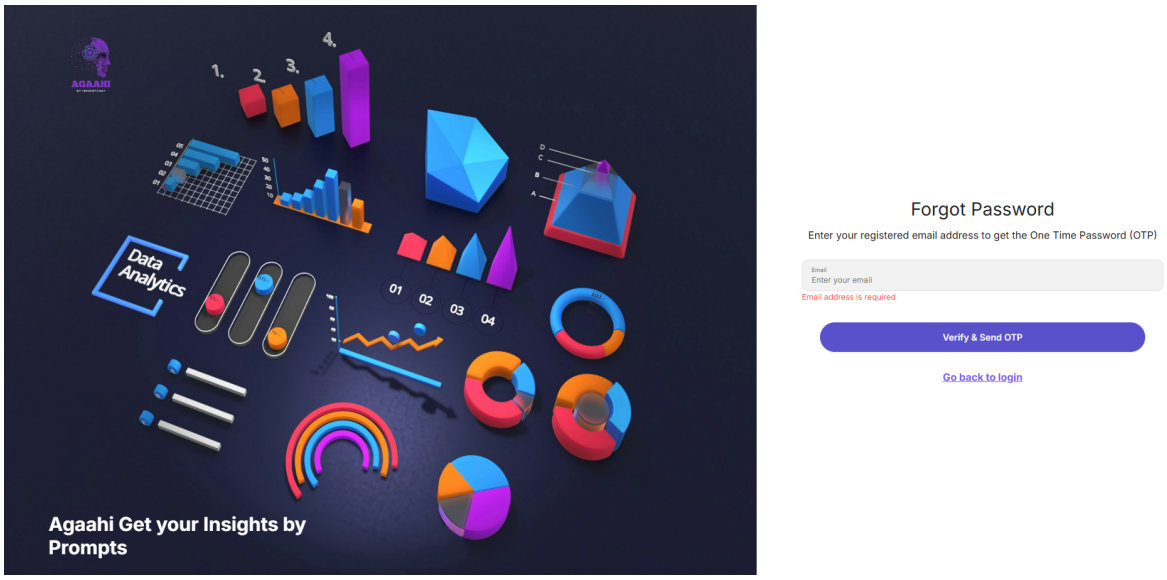


Figure 4.9: Forgot password Screen

4.2.1.6 Role Management

Once the company has been created, the owner must configure roles within the organization. By default, four roles are available: **Owner, Admin, Data Analyst, and Engineer**. The Owner role is pre-configured and cannot be modified, while the remaining roles are editable. The owner is responsible for assigning table-level permissions to each role, ensuring that every user has controlled access to data based on their designated role. This role-based access control enforces secure and differentiated visibility across the platform. From here Onwer can also update the role.

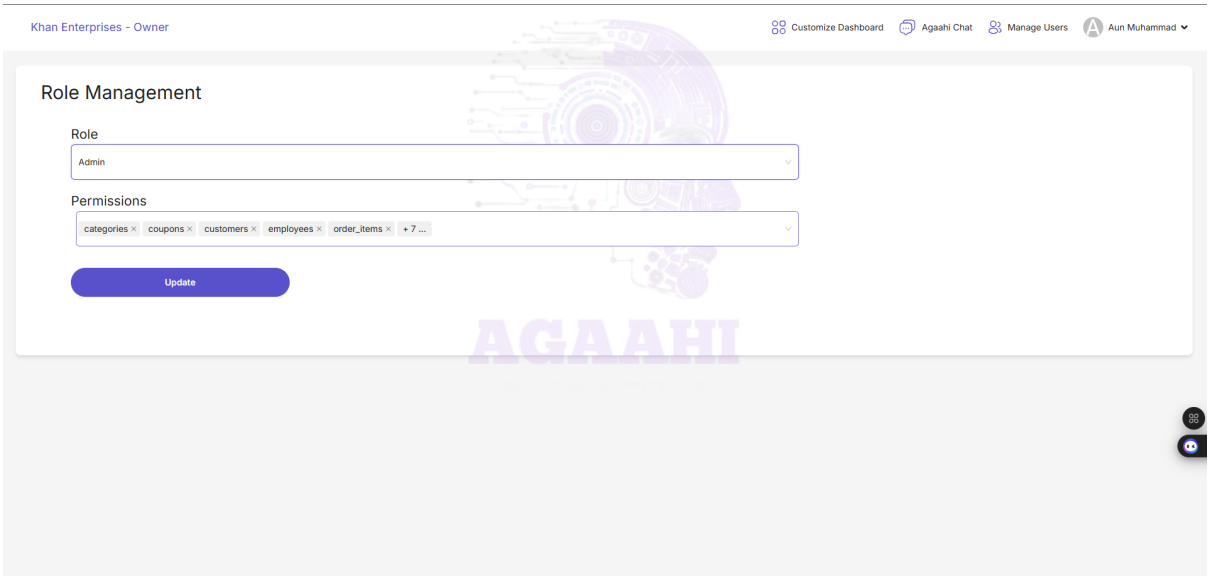


Figure 4.10: Role Management Screen

4.2.1.7 Employee Management

The **Employee Management** section displays a comprehensive list of all employees associated with a company. Only the company owner has the authority to create and manage employee accounts. This section provides a quick overview of each employee's general information, enabling efficient monitoring and administration.

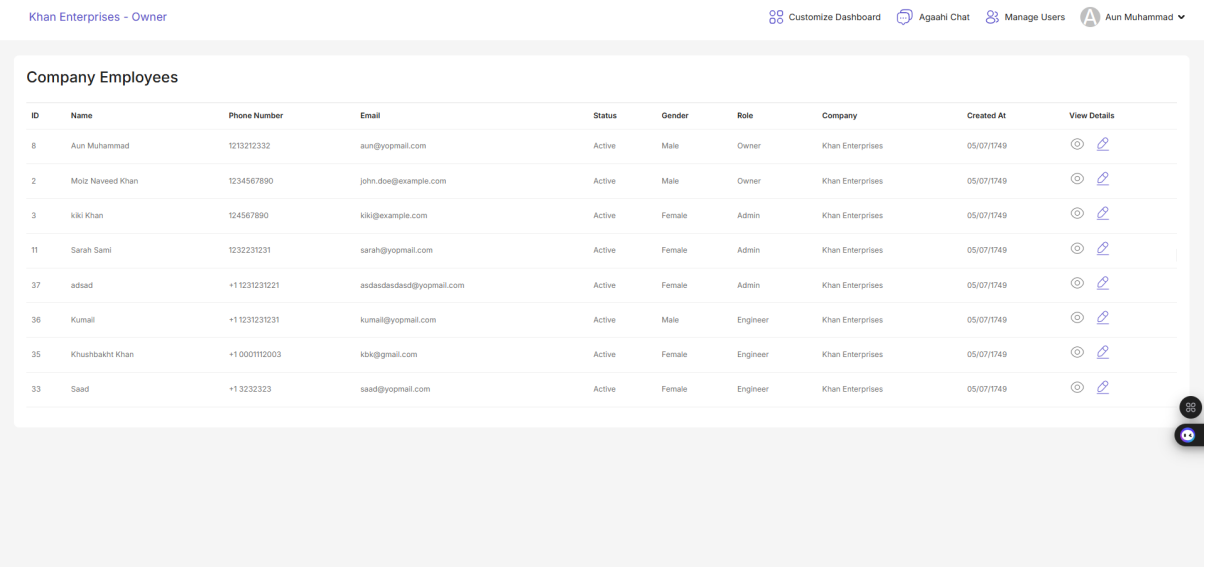


Figure 4.11: Employee Listing

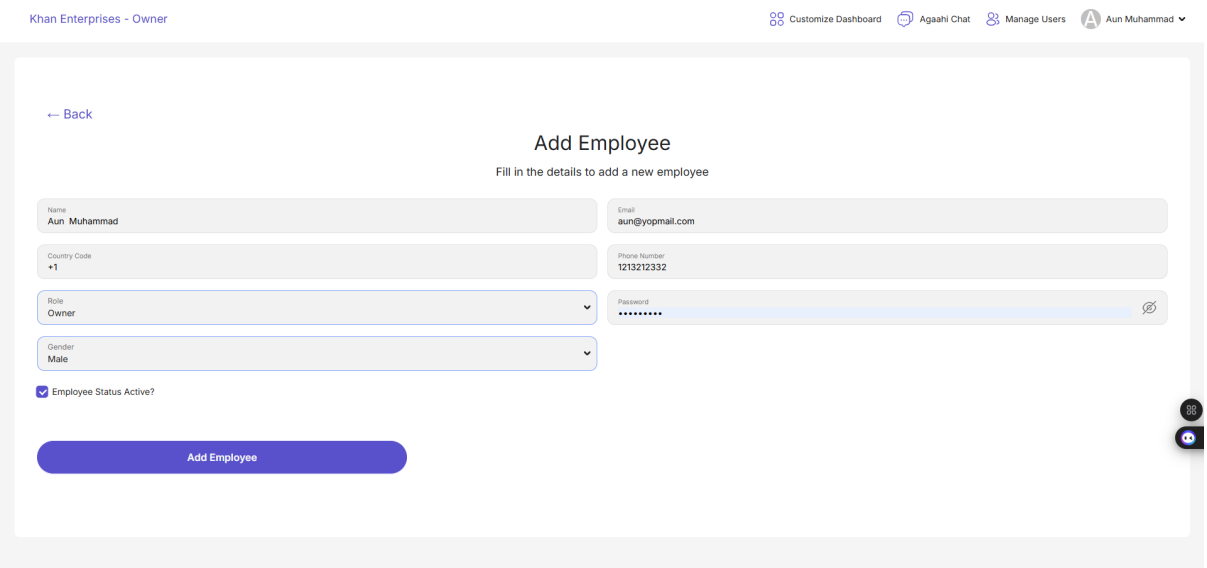


Figure 4.12: Add Employee

The owner/Admin can **Add employees** across the roles they have configured. The owner can assign multiple employees to the same role but cannot assign the owner role to anyone else. The owner has more privileges than other roles.

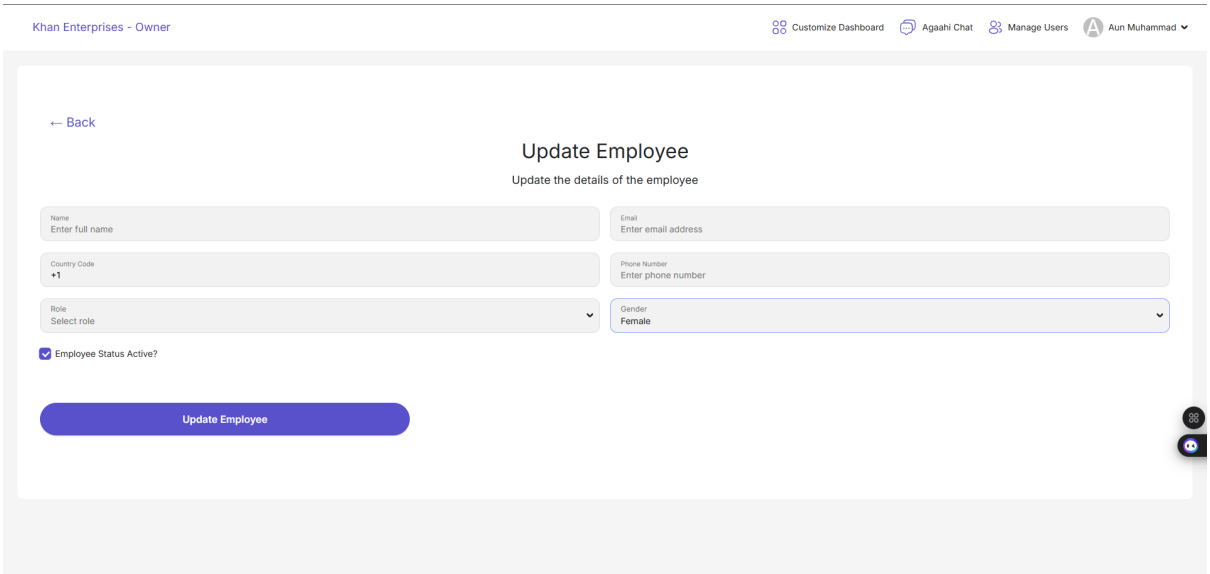


Figure 4.13: Update Employee

The user can update an employee's general information or password here.

4.2.1.8 My Profile View

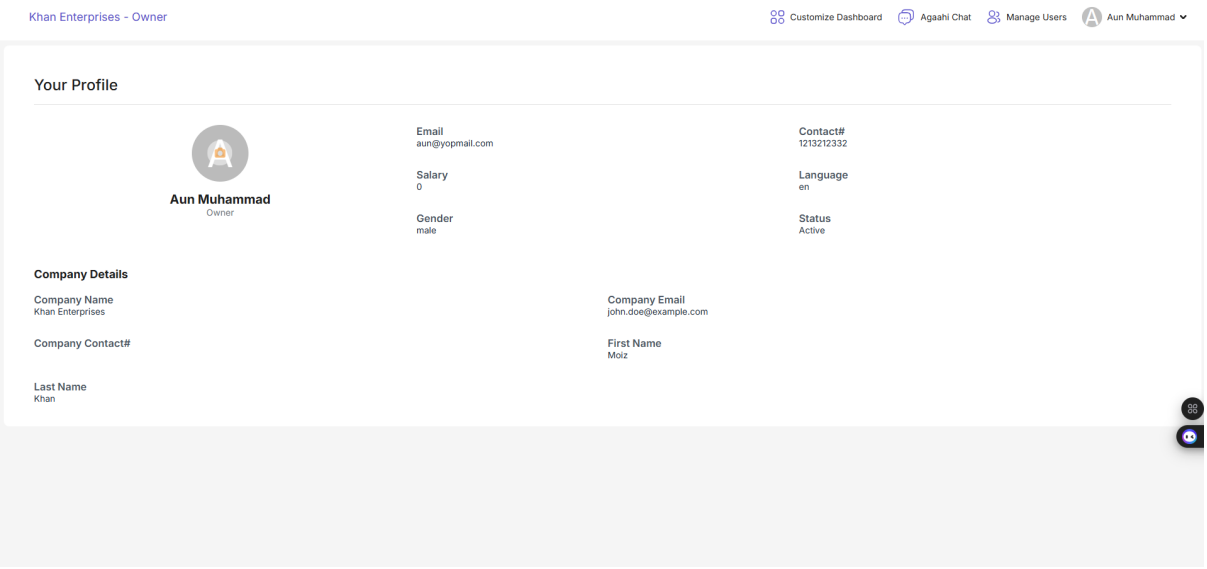


Figure 4.14: Profile View

Any user can view their **profile** by clicking the profile icon in the top-right corner, as shown in Fig. 4.14.

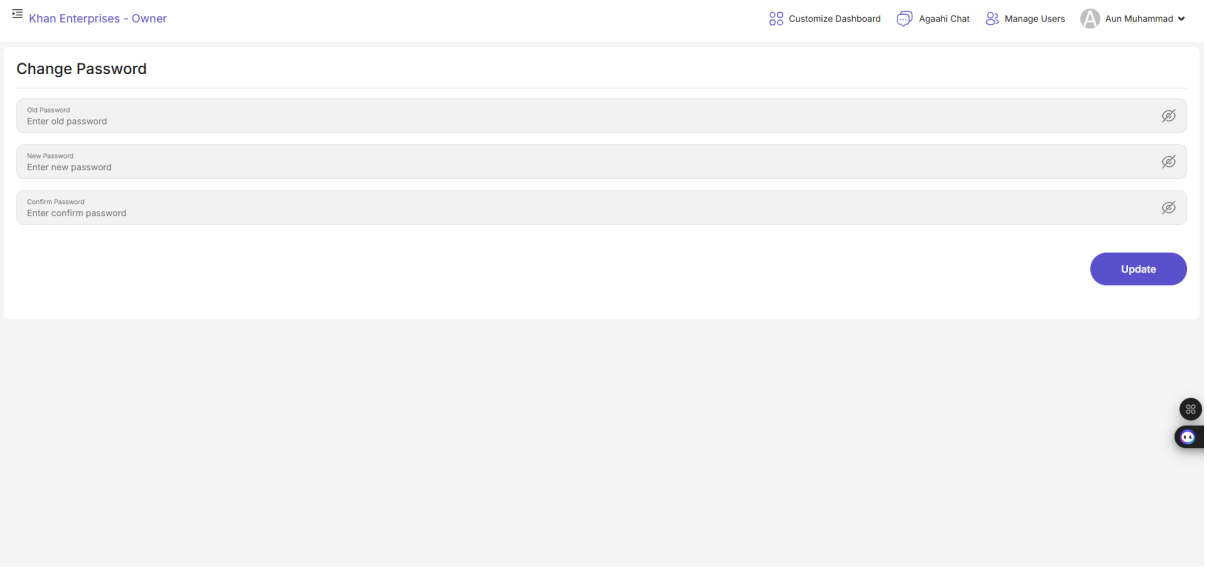


Figure 4.15: Change Password

4.2.1.9 Smart Query Lens

The Smart Query Lens feature allows users to input natural language prompts—such as **"Show me all the employees"**—and instantly receive both a table of results and the SQL query used to generate them. This bridges the gap between non-technical users and complex data by translating everyday language into actionable database queries. The system uses a language model to convert the prompt into SQL, executes the query on the database, and returns a structured table along with the raw query for transparency and validation.

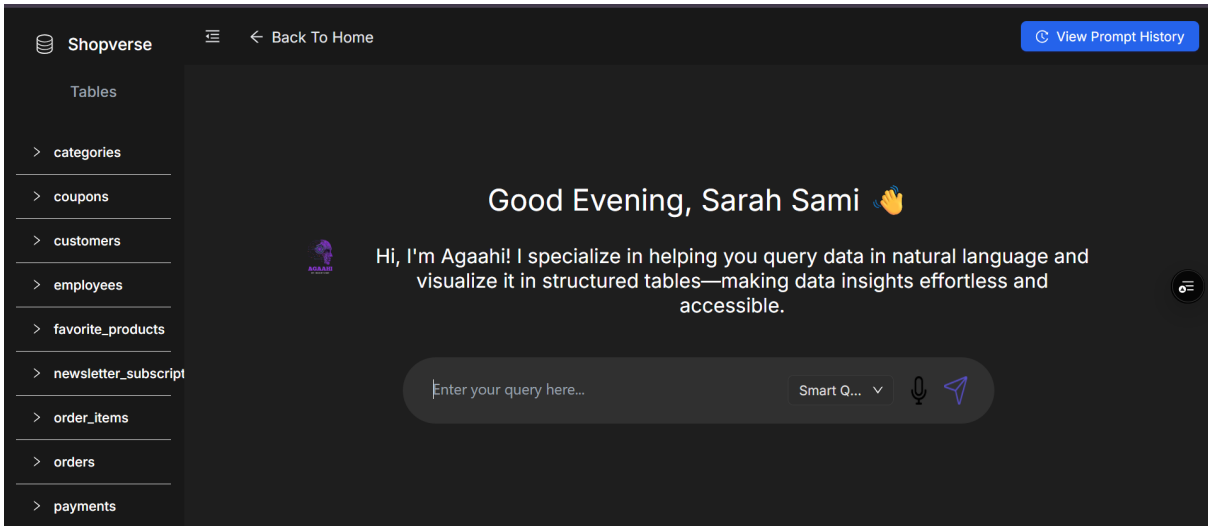


Figure 4.16: Smart Query Lens Landing Page

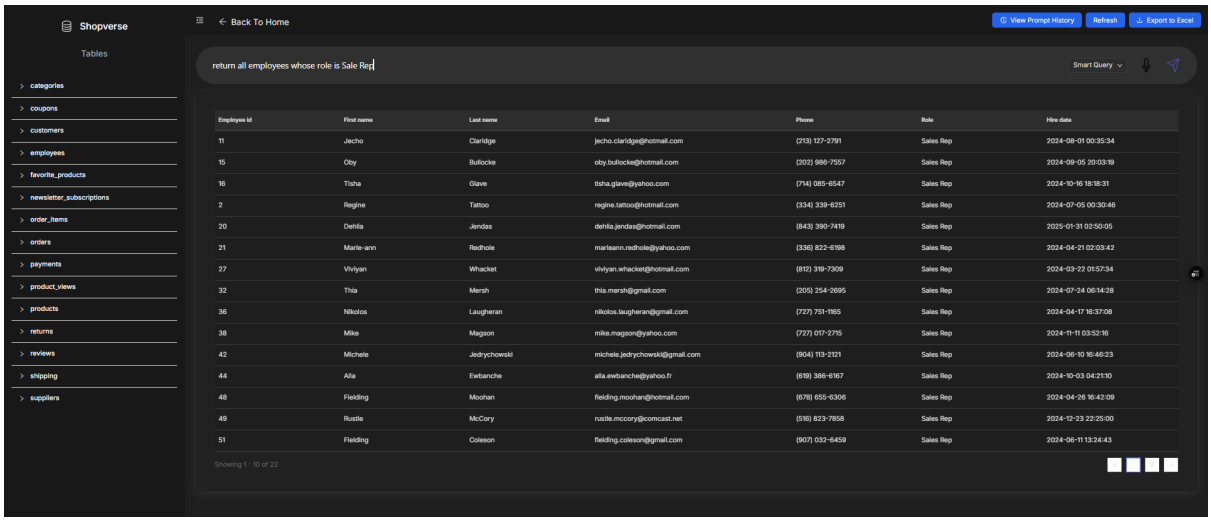


Figure 4.17: Smart Query Lense Result

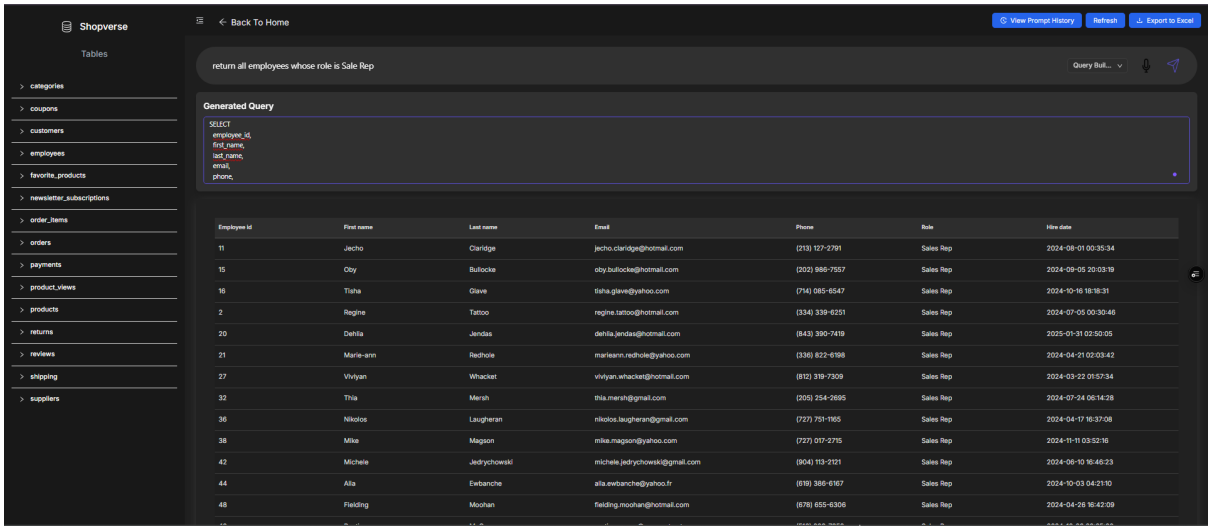


Figure 4.18: Query Builder Result

It provide feature of **Query Builder** in which user just can view the SQL statement and can modify the query and get results from it.

4.2.1.10 Chat Feature

The basic chat interface allows users to query data by typing in natural language or using voice recognition. Users can simply prompt the system and receive real-time insights instantly. However, it's important to note that users need to phrase their prompts according to their specific needs.

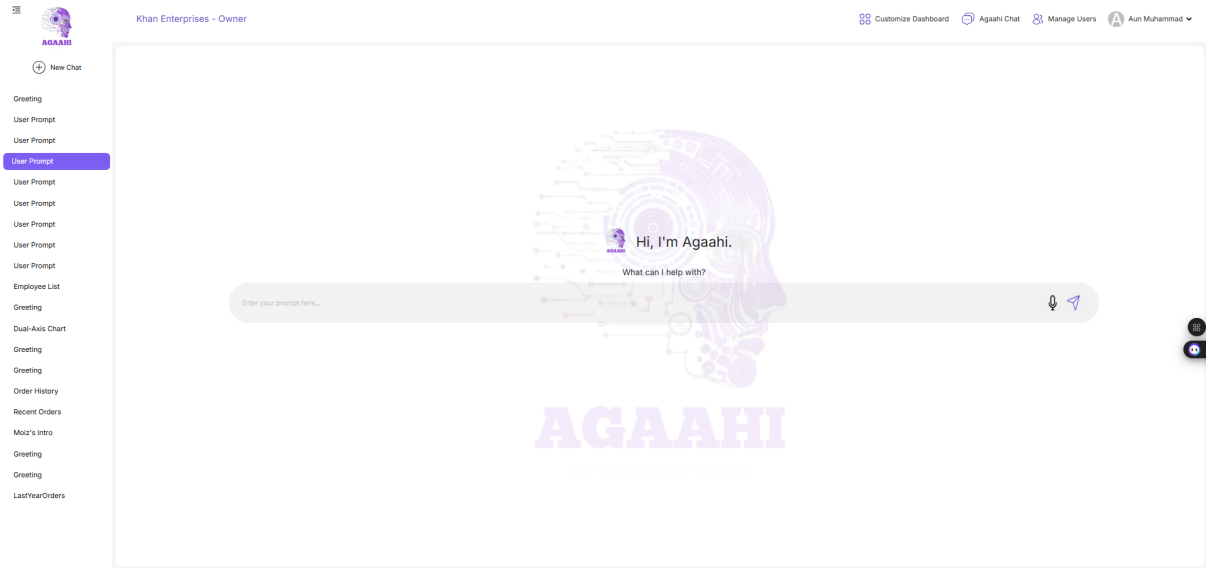


Figure 4.19: Chat Feature

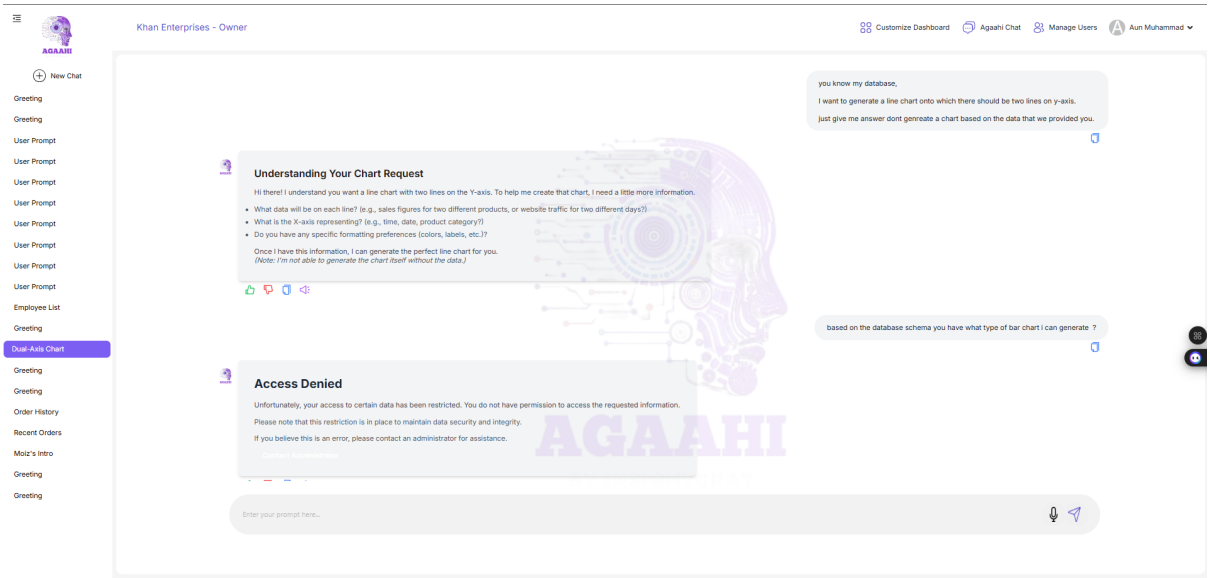


Figure 4.20: Chat with History

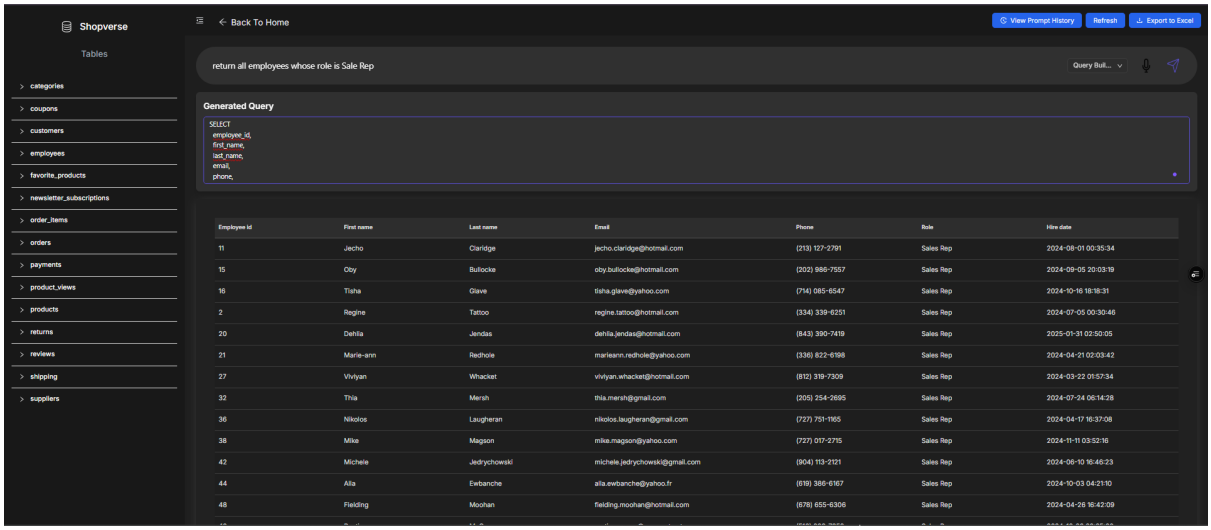


Figure 4.21: Chat Feature in Dark Theme

Chat Interface in Dark Theme

4.2.1.11 Report Generation Feature

Users can generate or export a CSV/Excel report for any type of information by entering a specific prompt in the chat interface, as demonstrated in Fig. 4.18. To do this, users simply need to include the keyword **"generate a csv"** within their query. Upon recognizing this keyword, Agaahi will process the request and provide a downloadable link containing the requested data in CSV or Excel format, allowing for easy offline access, sharing, or further analysis.for the report.

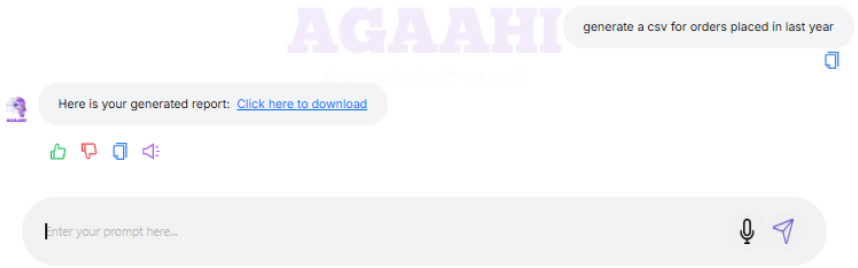
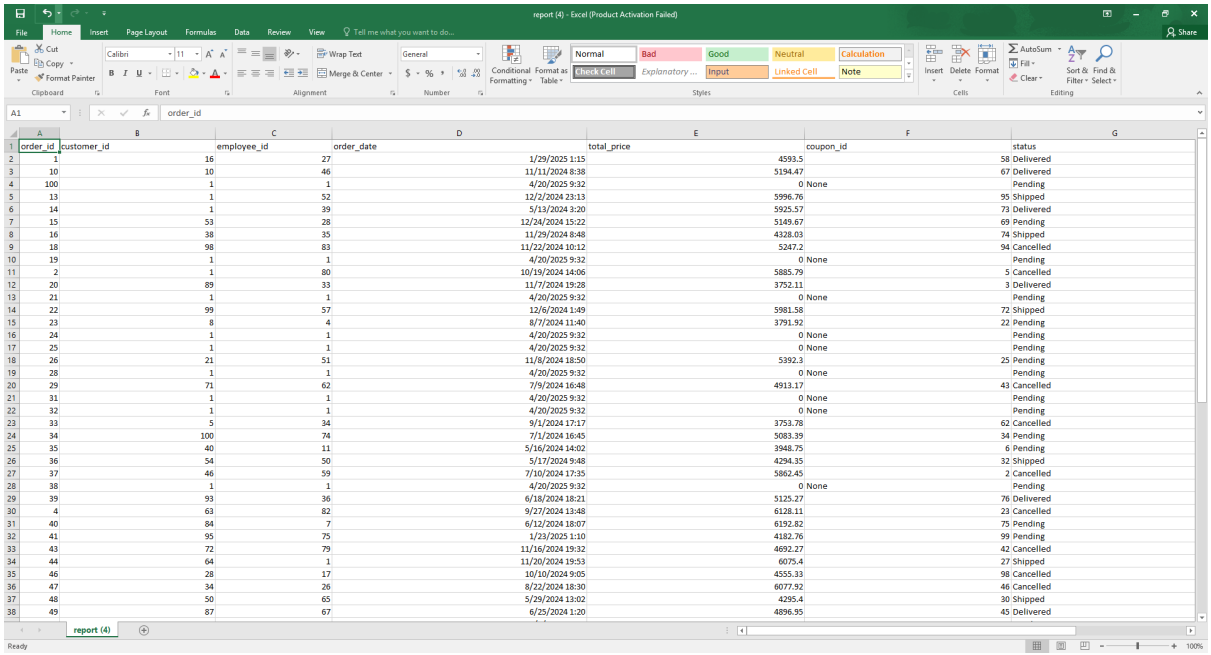


Figure 4.22: Prompt to Generate Report



order_id	customer_id	employee_id	order_date	total_price	coupon_id	status
1	16	27	1/29/2025 1:15	4593.5	58	Delivered
10	10	46	11/11/2024 8:38	5194.47	67	Delivered
100	1	1	4/20/2025 9:32	0	None	Pending
13	1	52	12/2/2024 23:13	5996.76	95	Shipped
14	1	39	5/13/2024 3:20	5925.57	73	Delivered
15	53	28	12/24/2024 15:22	5149.67	69	Pending
16	38	35	11/29/2024 8:48	4328.03	74	Shipped
18	98	83	11/22/2024 10:12	5247.2	94	Cancelled
19	1	1	4/20/2025 9:32	0	None	Pending
2	1	80	10/19/2024 14:06	5885.79	5	Cancelled
20	89	33	11/7/2024 19:28	3752.11	3	Delivered
21	1	1	4/20/2025 9:32	0	None	Pending
22	99	57	12/6/2024 1:49	5981.58	72	Shipped
23	8	4	8/7/2024 11:40	3791.92	22	Pending
24	1	1	4/20/2025 9:32	0	None	Pending
25	1	1	4/20/2025 9:32	0	None	Pending
26	21	51	11/8/2024 18:50	5392.3	25	Pending
28	1	1	4/20/2025 9:32	0	None	Pending
29	71	62	7/9/2024 16:48	4913.17	43	Cancelled
31	1	1	4/20/2025 9:32	0	None	Pending
32	1	1	4/20/2025 9:32	0	None	Pending
33	5	34	9/1/2024 17:17	3753.78	62	Cancelled
34	100	74	7/1/2024 16:45	5083.39	34	Pending
35	40	11	5/16/2024 14:02	3948.75	6	Pending
36	54	50	5/17/2024 9:48	4294.35	32	Shipped
37	46	59	7/10/2024 17:35	5862.45	2	Cancelled
38	1	1	4/20/2025 9:32	0	None	Pending
39	93	36	6/18/2024 18:21	5125.27	76	Delivered
4	63	82	9/27/2024 13:48	6128.11	23	Cancelled
40	84	7	6/12/2024 18:07	6192.82	75	Pending
41	95	75	1/23/2025 1:10	4182.76	99	Pending
43	72	79	11/16/2024 19:32	4692.27	42	Cancelled
44	64	1	11/20/2024 19:53	6075.4	27	Shipped
46	28	17	10/10/2024 9:05	4555.33	98	Cancelled
47	34	26	8/22/2024 18:30	6077.92	46	Cancelled
48	50	65	5/29/2024 13:02	4295.4	30	Shipped
49	87	67	6/25/2024 1:20	4896.95	45	Delivered

Figure 4.23: Excel Report Generation Result

4.2.1.12 Chat Graph Feature

Users can easily generate visual figures while participating in ongoing chat conversations by simply requesting them through natural language. At present, the system supports three types of visualizations: Line Charts, Bar Charts, and Pie Graphs. This feature enables users to quickly visualize data without leaving the chat interface.

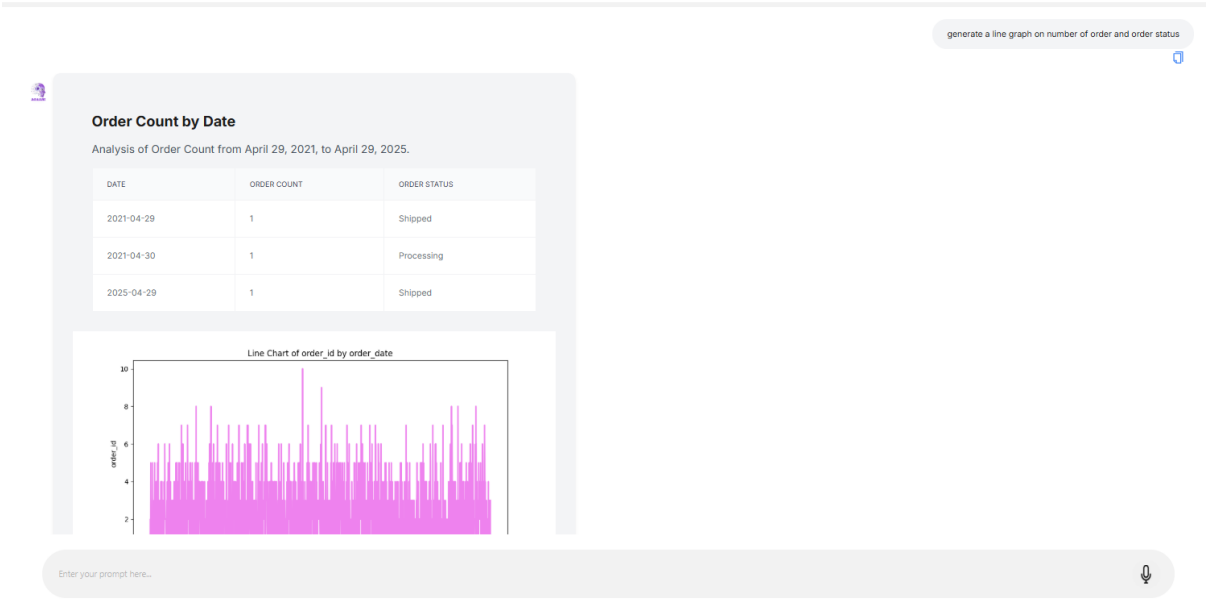


Figure 4.24: In Chat Graph Result

4.2.1.13 Dashboard Feature

Dashboard Preview Users have the flexibility to fully customize their dashboard according to the type of data they wish to monitor on the main screen of the application. This can be done either through manual configuration or by simply entering a prompt using natural language. The system responds in real time, instantly displaying insights based on the user's request. This interactive setup allows users to specify exactly what data they want to track—such as sales trends, stock levels, or employee performance—and how they want it visualized. The dashboard supports a wide range of visualization types including line charts, bar graphs, pie charts, cards, and more, enabling users to create a personalized, insight-rich view tailored to their operational needs.

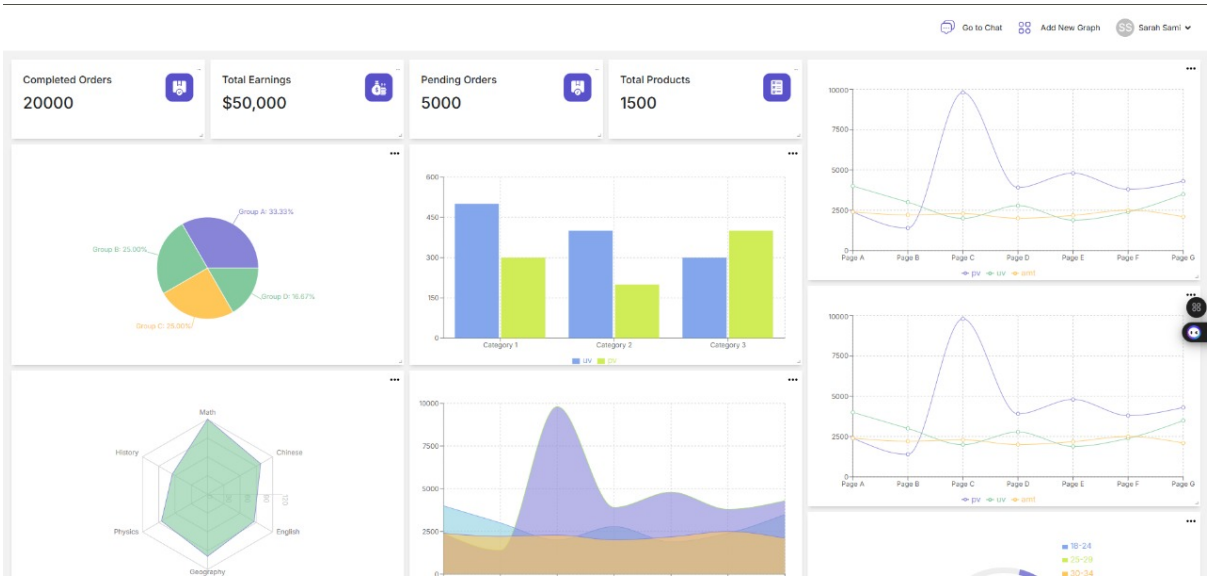


Figure 4.25: Dashboard Preview

Dashboard Setup

- **Supported Dashboard Graph Type:** First, the user needs to select the type of graph from the supported graph types. This selection helps the system understand how to visualize the requested data—whether it's a line chart, bar graph, pie chart, card, or another supported format—ensuring the insights are presented in the most effective and meaningful way.

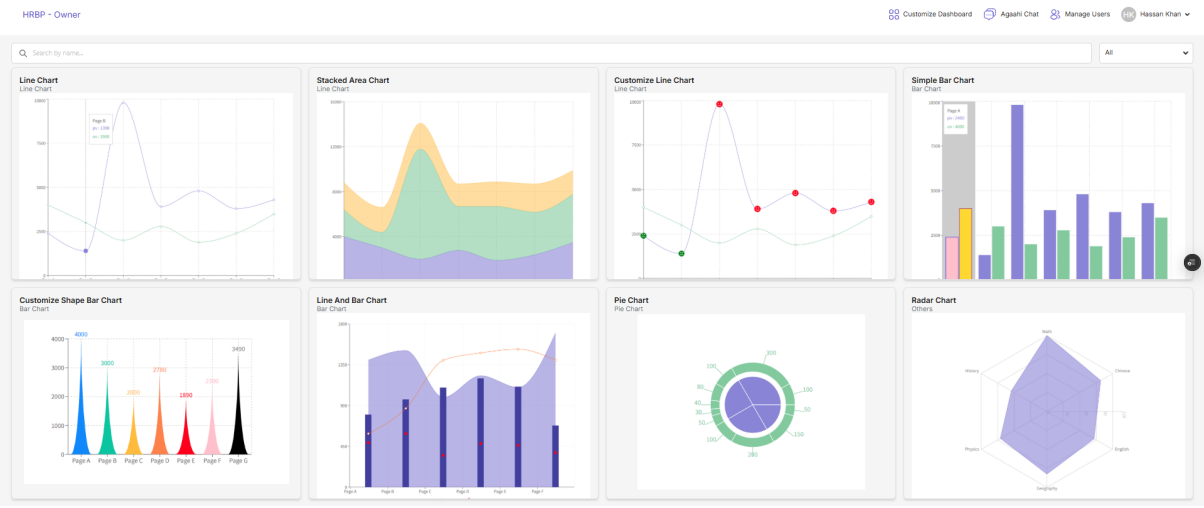


Figure 4.26: Supported Dashboard Graph Type

- **Set Graph with Prompt:** After selecting the desired graph type, the user needs to enter a prompt in the prompt box. This prompt should clearly describe the data they want to visualize. Based on the prompt, the system will fetch and display real-time insights using the selected graph format, providing an interactive and customized data view.

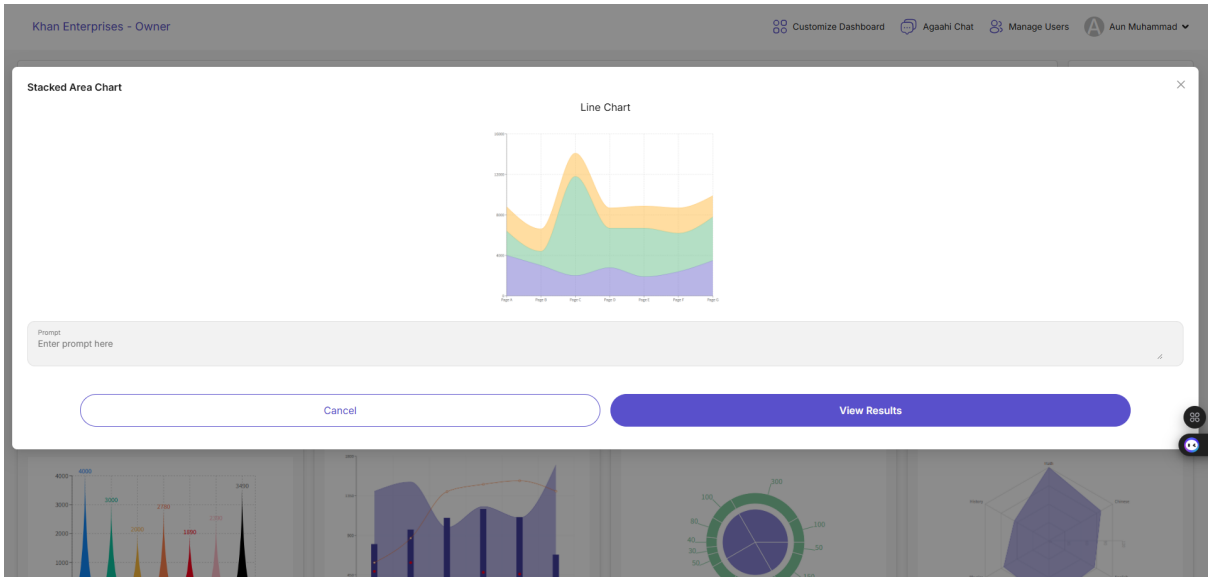


Figure 4.27: Set Graph with Prompt

- **Graph Preview:** It displays a preview of how the selected graph will look based on the user's prompt. This allows the user to review the visualization before finalizing it. If the user confirms by selecting "Yes, add to dashboard," the graph will be added to their personalized dashboard for ongoing access and monitoring.

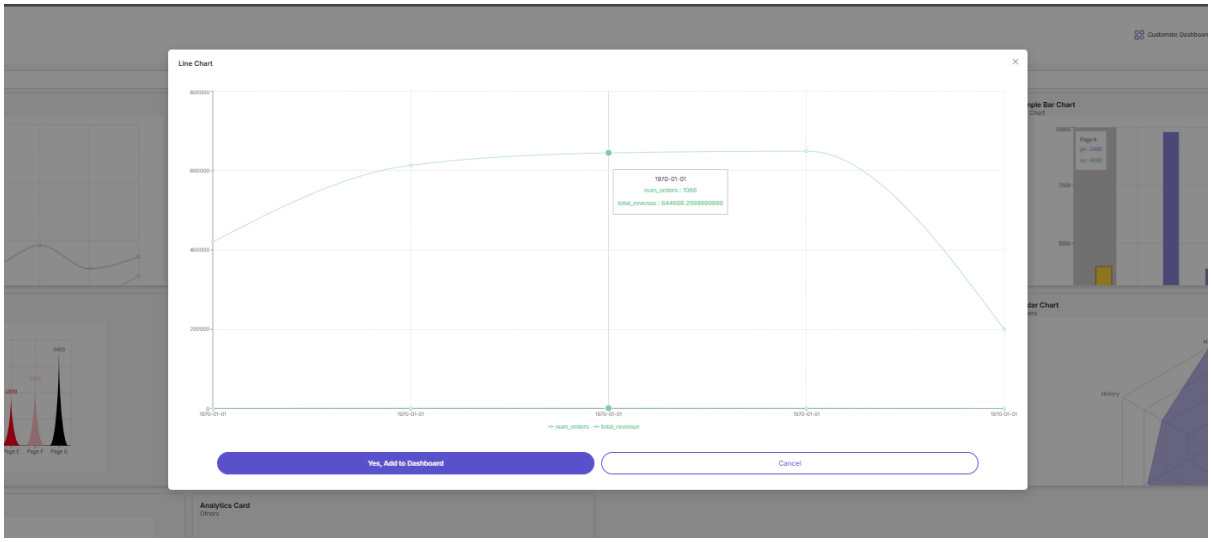


Figure 4.28: Graph Preview with Real Time Data

4.2.1.14 Contact Us

The Contact Us page of Agaahi serves as a centralized hub for fostering collaboration and communication between the platform and its users, partners, or prospective clients. Designed with a modern, user-friendly interface, the page features a structured inquiry form that collects essential details such as name, email, phone number, and a message field, enabling stakeholders to submit queries, feedback, or partnership requests efficiently. A prominent call-to-action button, “Send to the moon ”, aligns with Agaahi’s innovative and aspirational branding, encouraging engagement while maintaining a professional tone. The page also emphasize company’s operational presence and accessibility across key regions. A closing tagline, Ready to harness no-code AI analytics? reinforces the platform’s mission to democratize data-driven insights.

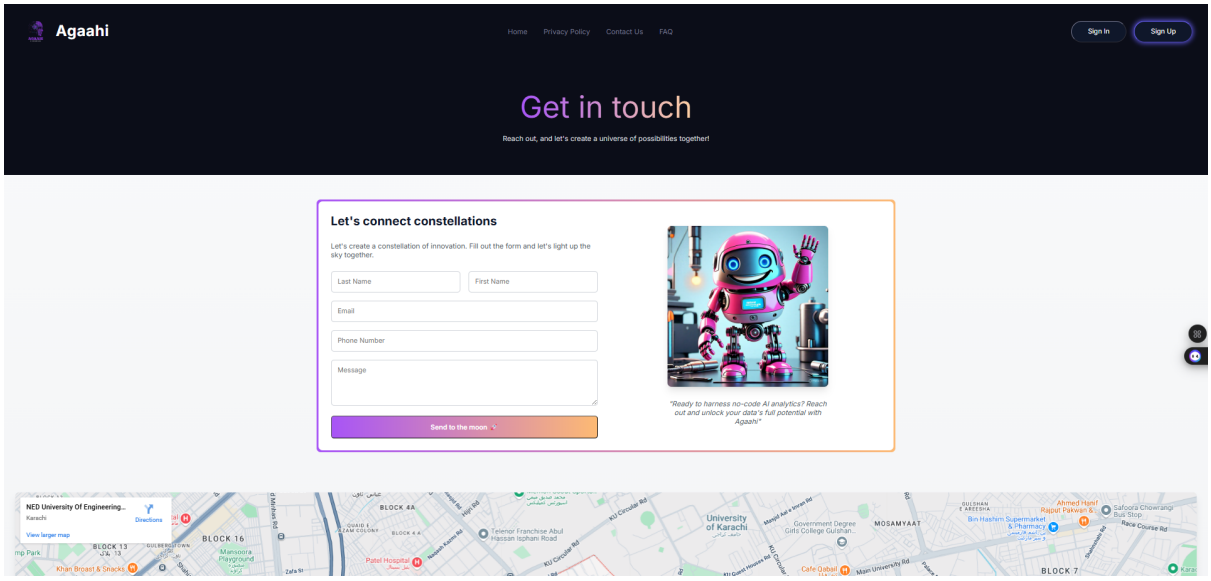


Figure 4.29: Contact Us Screen

4.2.1.15 FAQs

The FAQs page acts as a one-stop solution center for prospective and existing users looking for answers related to Agaahi features, use cases and security. At the page's beginning, the "Frequently Asked Questions" header features the "Book a Demo" and "Contact Support" making them readily accessible for active users. Below this information, each query appears in an accordion style format whereby a user clicks on a question and the answer appears without having to change the page. The questions are organized in both general "What is Agaahi, and who is it for?" to very specific workflows "How can I create dashboards using Agaahi?" and even policies "How does Agaahi ensure the data shared is secure?" ensuring that business users, analysts, and even decision makers instantly have the information they need in order to start off or resolve simple issues.

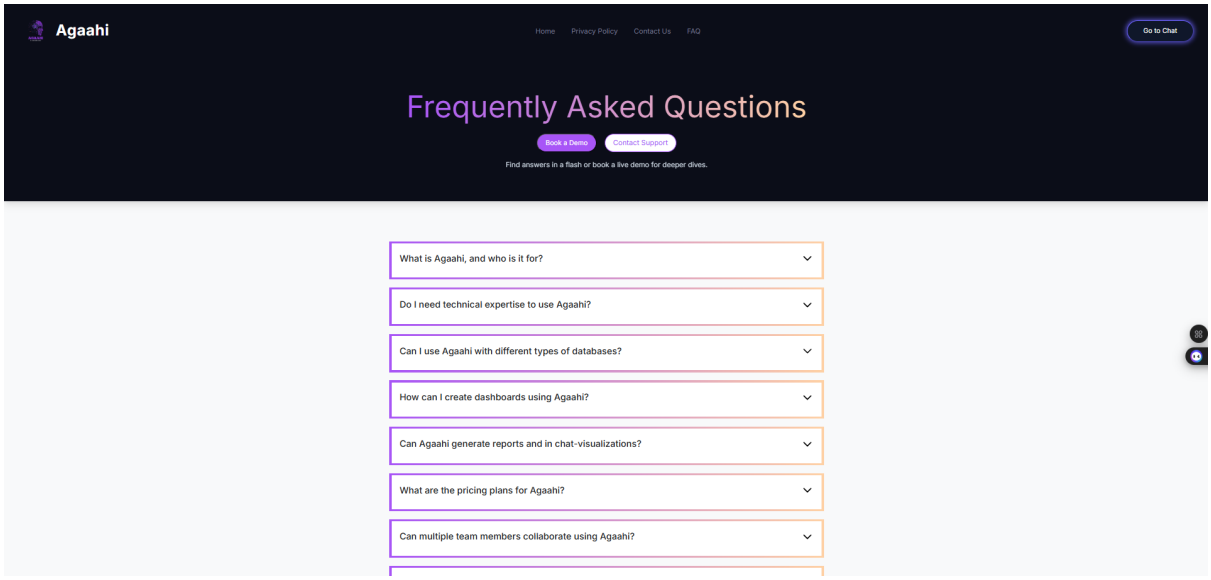


Figure 4.30: FAQs Screen

4.2.1.16 Privacy Policy

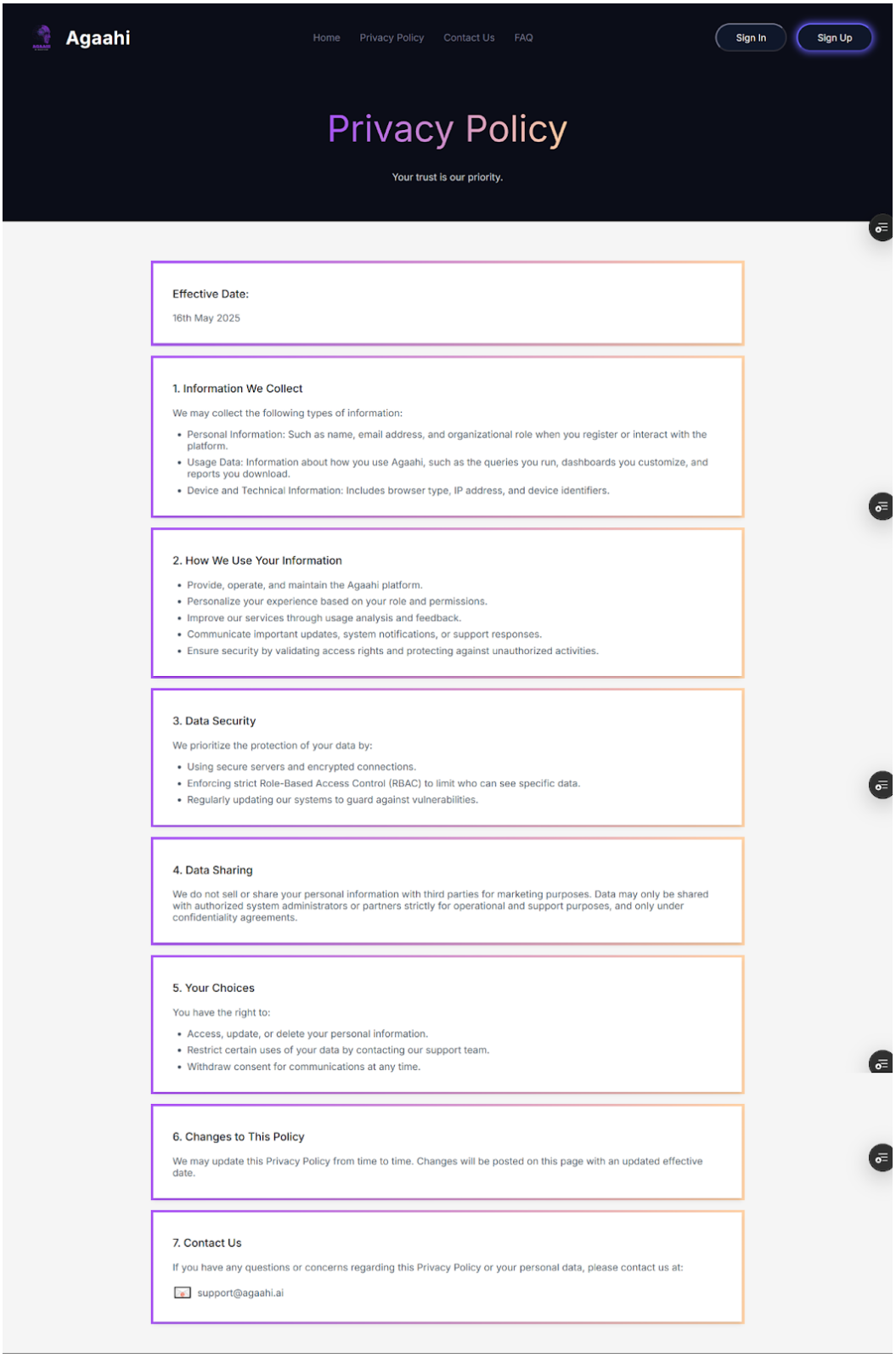


Figure 4.31: Privacy Policy

The **Privacy Policy** page highlights Agaahi's efforts towards user data protection and how they are committed to transparency regarding its collection and usage. Stating the effective date of the policy, it proceeds to describe its six sections. Section one describes the collected information which consists of account details and device metadata. Followed by information usage for platform operation and improvement in section two. Then data security measures in three which cover encrypted connection and escalation access control as shown in **Fig. 4.26.** Next is circumstantial data sharing with third parties. Last; users' data control rights and update communication section. Questions beyond this policy can be directed to listed contacts. This base structure preserves accuracy while allowing business users effortless accessibility to overarching data privacy principles.

4.2.2 Admin Panel

4.2.2.1 Login

This secure login screen provides authorized access to the Agaahi Admin Panel. Admins can log in using their registered email and password to manage and control the content, users, and overall system settings of Agaahi. The interface is simple, responsive, and designed to ensure fast access while maintaining high security standards.

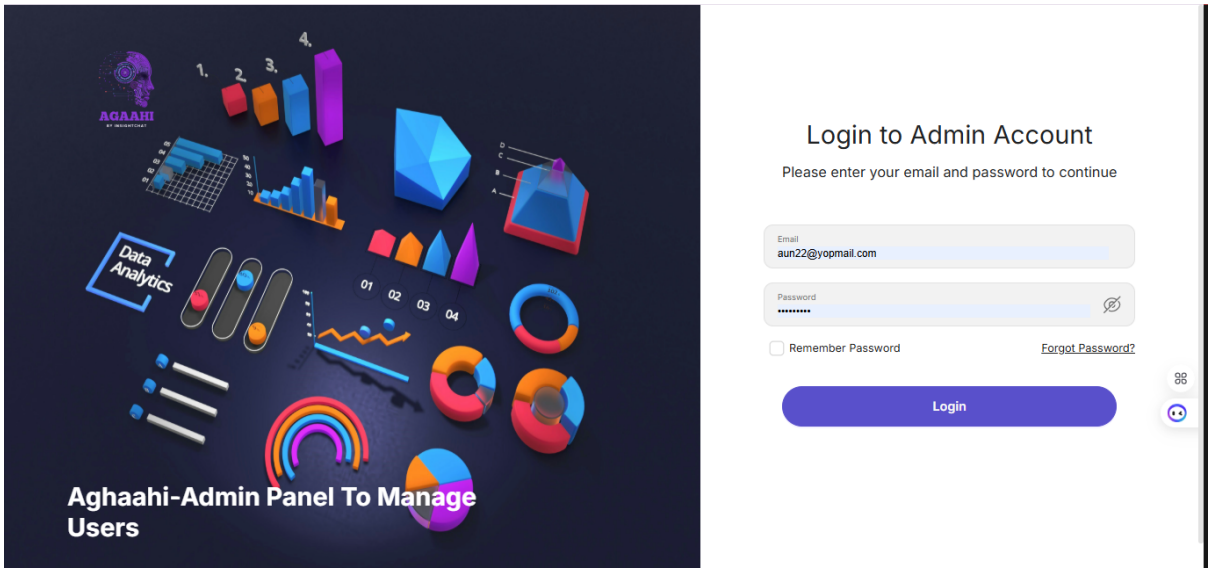


Figure 4.32: Admin Login Screen

4.2.2.2 Admin Management

This screen displays a complete list of all registered admins who have access to the Agaahi Admin Panel. Each admin entry includes their *name, email, assigned role, and status*. There are two types of roles: **Super Admin** (full access and permissions, including managing other admins) and **Admin** (limited access based on assigned responsibilities). From this screen, Super Admins can add new admins, edit existing roles, or deactivate accounts as needed.

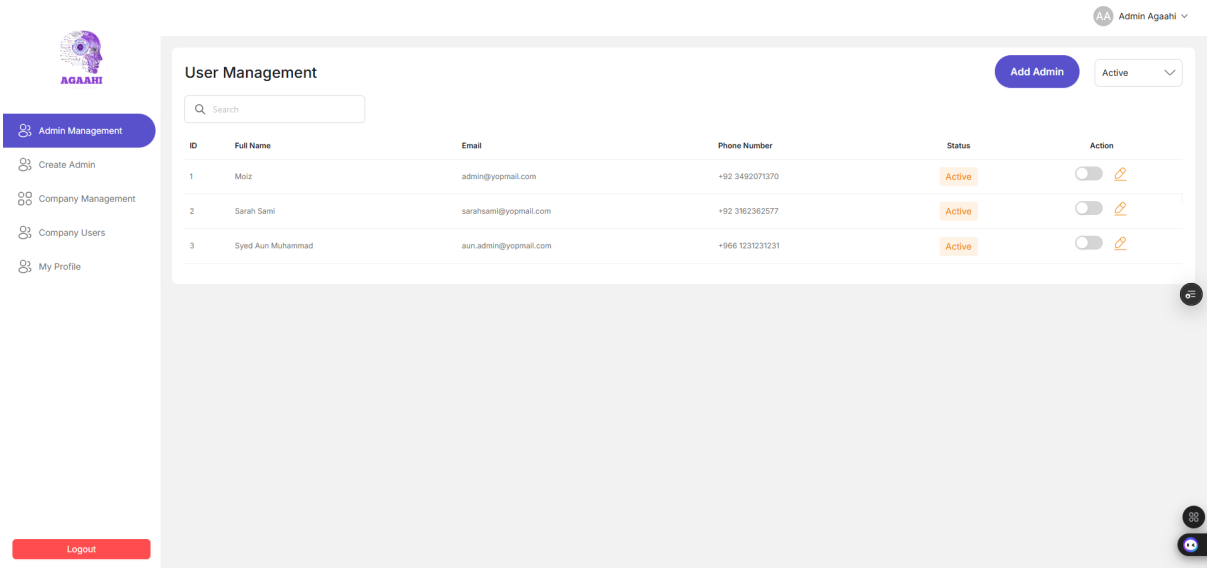


Figure 4.33: Admin Listing

Add Admin Feature: This screen allows Super Admins to create new admin accounts for the Agaahi Admin Panel. The form includes fields for the admin's name, email, password, and role selection (Super Admin or Admin). Upon submission, the new admin is added to the system with access based on the selected role. Validation checks ensure all required fields are completed correctly before account creation.

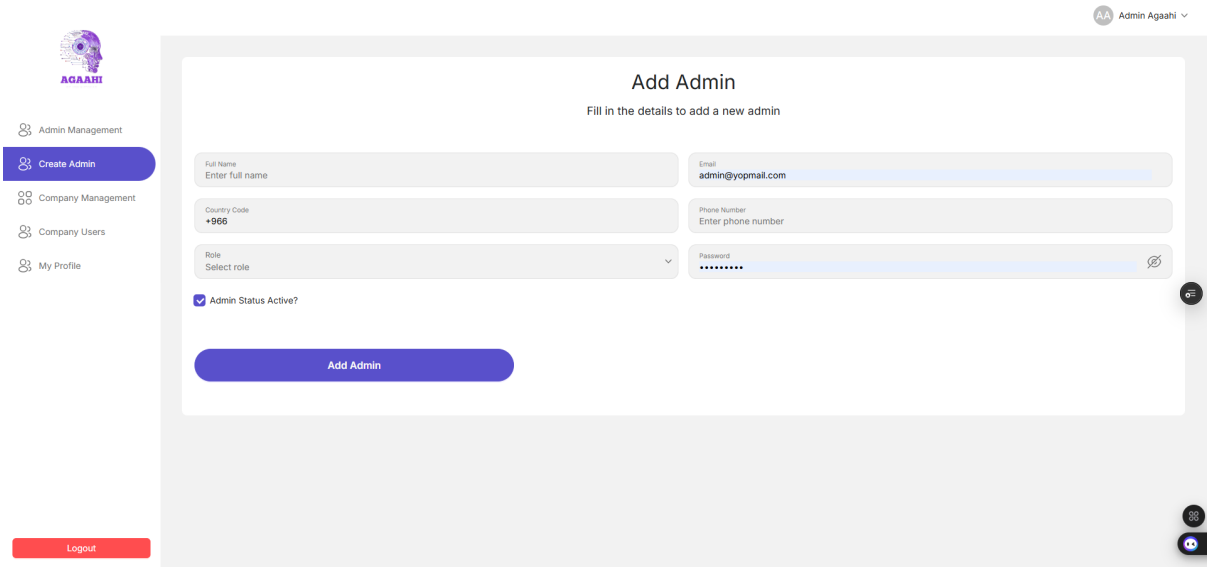


Figure 4.34: Add Admin Screen

Update Admin Feature: This screen enables Super Admins to edit the details of an existing admin. Admin information such as name, email, and role can be updated. Super Admins can also reset the admin's password or change their access level between Admin and Super Admin. All changes are saved securely and reflected instantly in the Admin Management list.

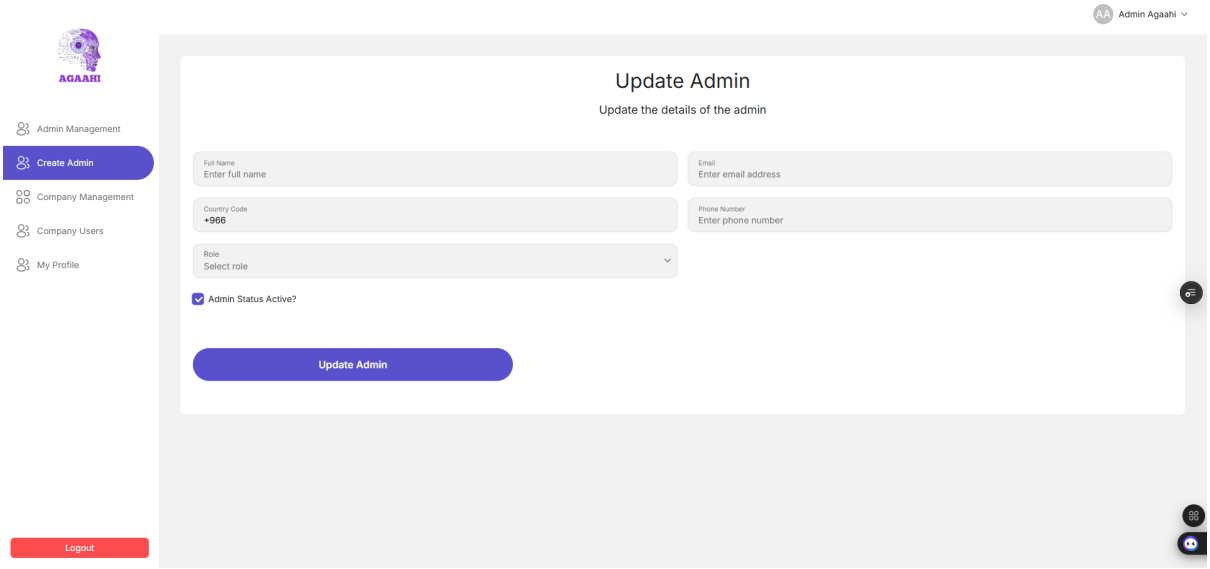


Figure 4.35: Update Admin Screen

4.2.2.3 Admin Profile View

This screen allows admins to view and manage their personal profile within the Agaahi Admin Panel. It displays their name, email, role (Admin or Super Admin), and other relevant details. Admins can update their basic information and change their password from this screen. It provides a simple and secure way for each admin to manage their own account settings without affecting system-wide data.

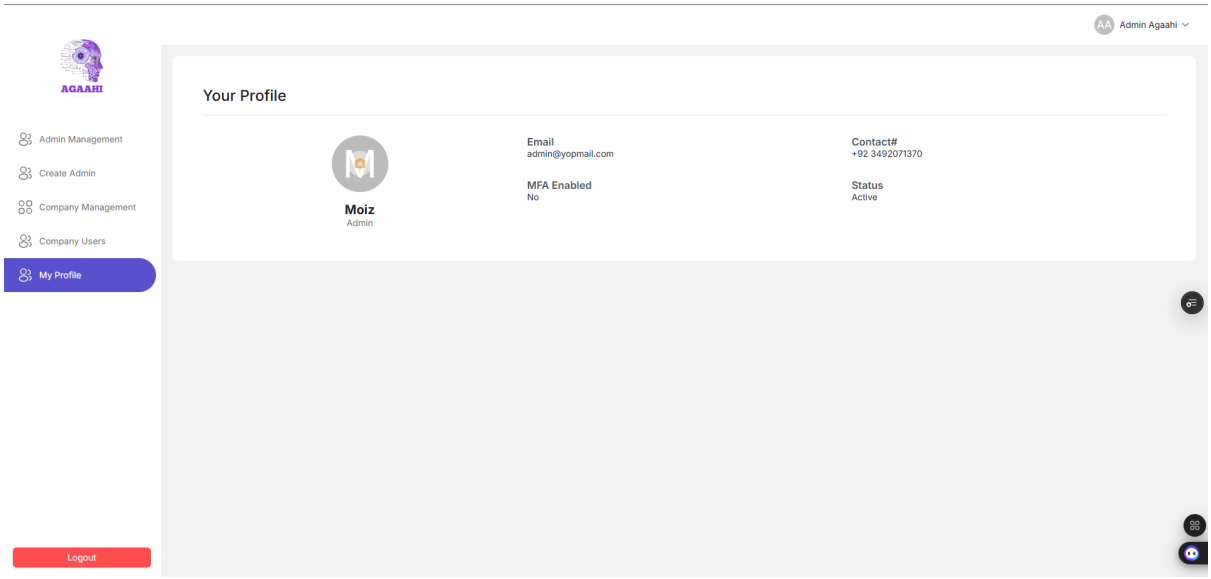


Figure 4.36: Admin Profile View

4.2.2.4 Change Password Feature

This screen allows admins to securely update their account password. To change the password, admins must enter their current password, followed by the new password and its confirmation. Real-time validation ensures that the new password meets security requirements. Upon successful update, the system confirms the change and enhances account protection.

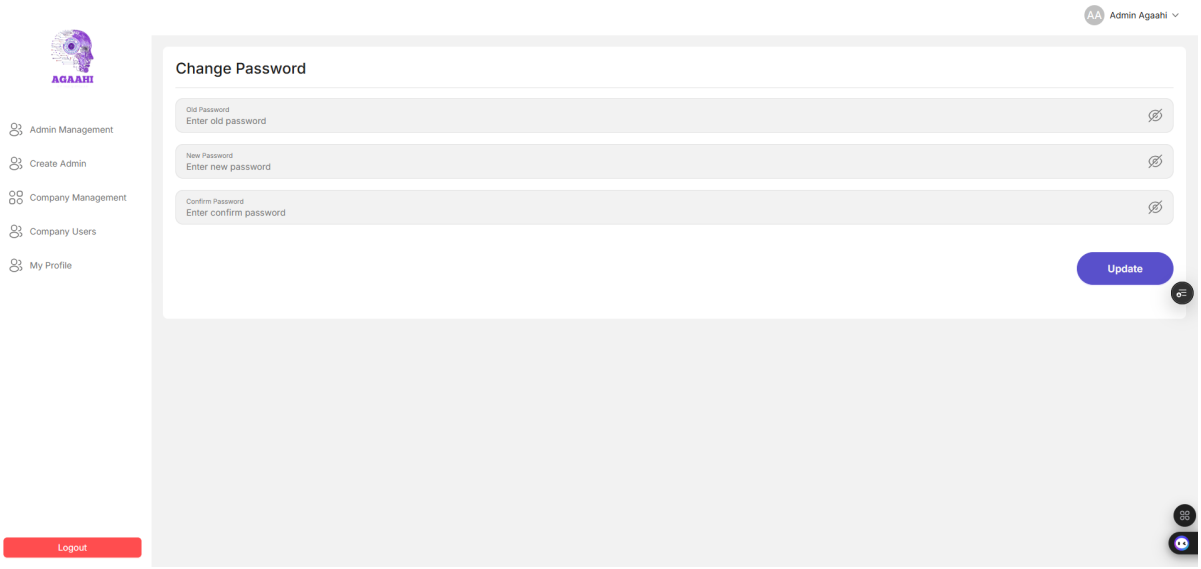


Figure 4.37: Admin Change Password

4.2.2.5 Company Management

This screen displays a list of all onboarded and registered companies within the Agaahi platform. Each entry includes key company details such as name, registration date, contact information, and current status (active/inactive). Admins can view, search, and filter the list for easy management and also number of employee present in each company. This screen serves as a central point to monitor and manage all company accounts using Agaahi.

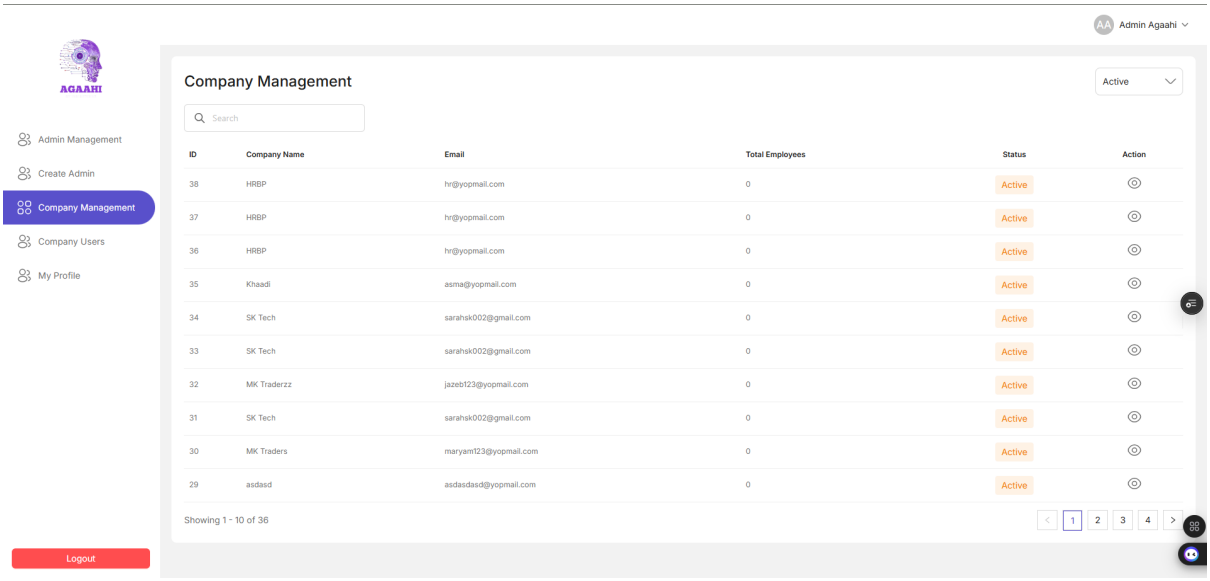


Figure 4.38: Registered Company listing

4.2.2.6 Employee Management

This screen displays a list of employees associated with each registered company on the Agaahi platform.The list can be filtered by company or searched by employee name for quick access. This screen helps admins monitor and manage the workforce structure across all onboarded companies.

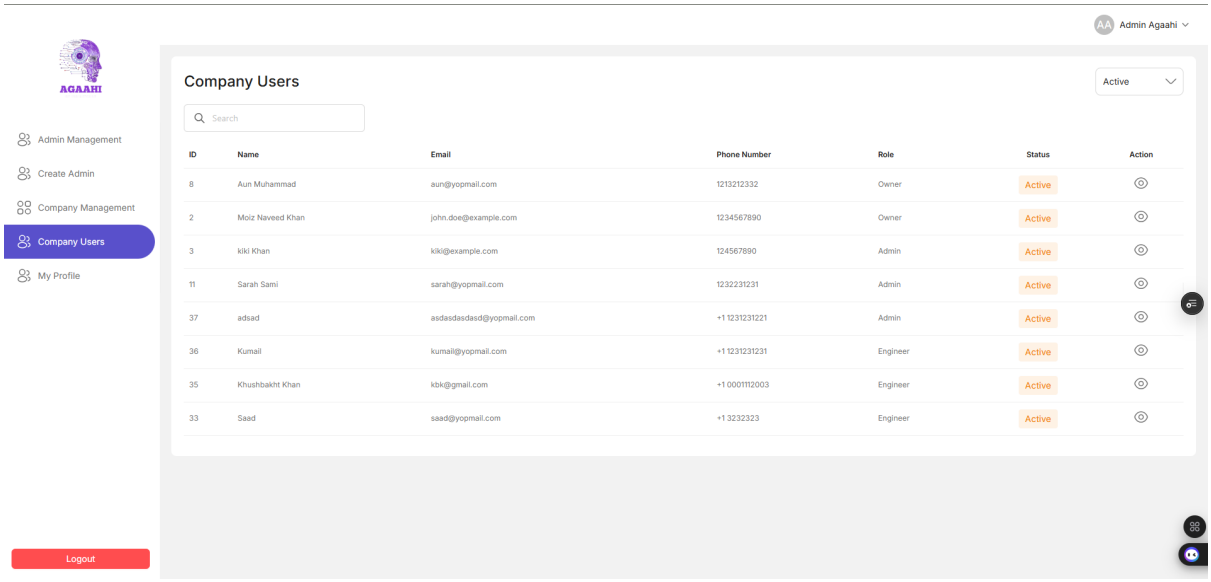


Figure 4.39: Users Against Company listing

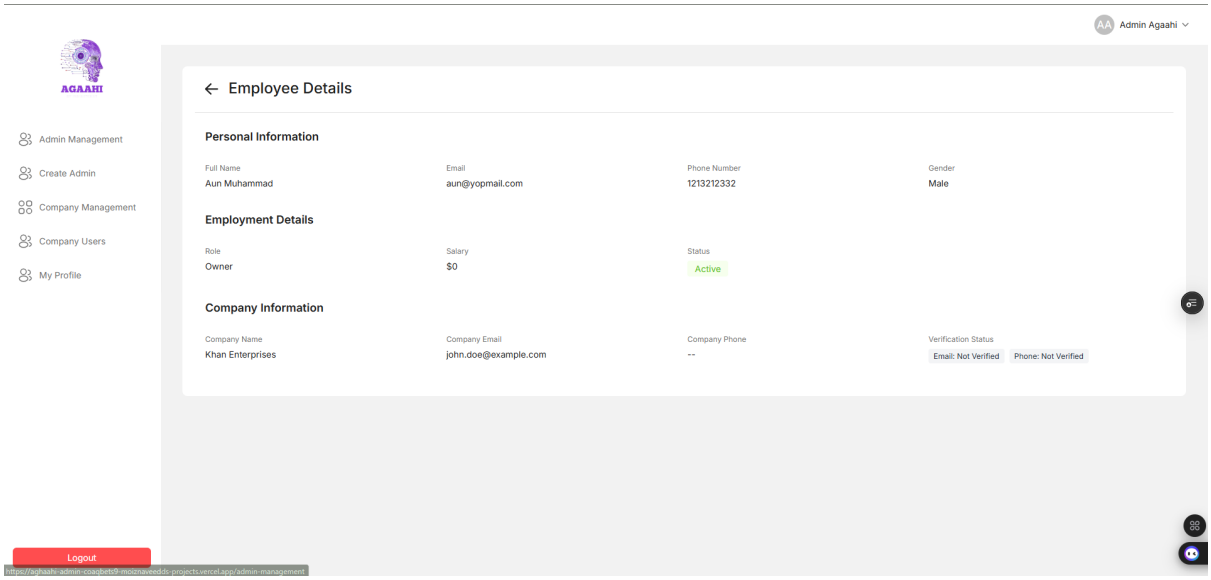


Figure 4.40: Users Against Company listing

4.3 Frontend Implementation

4.3.1 Technology Stack and Architecture

4.3.1.1 Core Technologies

- **React with TypeScript**: The frontend is built using React with TypeScript, providing robust type safety and enhanced development experience.
- **Vite**: Utilized as the build tool for faster development and optimized production builds.
- **Tailwind CSS**: Implemented for responsive and utility-first styling approach.
- **SCSS**: Used for custom styling and component-specific style definitions.

4.3.1.2 Project Structure

The frontend implementation follows a modular and scalable architecture with the following organization:

```
src/
├── assets/ ..............................Static assets (images, icons)
├── components/ ..................................Reusable UI components
├── pages/
│   ├── AuthScreens/ .........................Authentication-related pages
│   └── AppScreens/ ..............................Main application screens
├── services/ ..................... API and external service integrations
├── stores/ ..............................................State management
├── hooks/ ..........................................Custom React hooks
├── utils/ ..............................................Utility functions
├── constants/ ..................................... Application constants
├── styles/ .....................................Global styles and theme
└── navigation/ ............................Routing and navigation logic
```

4.3.2 Key Modules and Components

4.3.2.1 Authentication Module

- Implements user authentication flows
- Handles login, registration, and password recovery
- Manages authentication state and token management
- Protected route implementation for secure access

4.3.2.2 Navigation System

- Centralized routing configuration using React Router
- Role-based access control
- Navigation guards for protected routes
- Responsive navigation components for different screen sizes

4.3.2.3 State Management

- Centralized state management for application data
- Efficient data caching and persistence
- Real-time state updates and synchronization
- State isolation for better performance

4.3.2.4 API Integration Layer

- Structured API service implementation
- Request/response interceptors
- Error handling and retry mechanisms
- Data transformation and normalization

4.3.3 User Interface Components

4.3.3.1 Layout System

- Responsive layout components
- Consistent spacing and alignment
- Dynamic sidebar and header components
- Mobile-first design approach

4.3.3.2 Form Components

- Reusable form elements
- Form validation
- Error handling and user feedback
- Accessibility considerations

4.3.3.3 Data Display Components

- Tables and lists
- Cards and containers
- Modal dialogs
- Loading states and placeholders

4.3.4 Performance Optimizations

4.3.4.1 Code Splitting

- Route-based code splitting
- Lazy loading of components
- Dynamic imports for large dependencies
- Optimized bundle sizes

4.3.4.2 Caching Strategy

- Browser caching configuration
- API response caching
- Static asset optimization
- Memory management

4.3.4.3 Performance Monitoring

- Loading time optimization
- Runtime performance tracking
- Error boundary implementation
- Performance metrics collection

4.3.5 Development and Build Process

4.3.5.1 Development Environment

- Hot module replacement
- Development server configuration
- Environment variable management
- Developer tools and debugging setup

4.3.5.2 Build and Deployment

- Production build optimization
- Asset compression and minification
- Environment-specific builds
- Continuous integration setup

4.3.5.3 Code Quality

- ESLint configuration for code quality
- Prettier for consistent code formatting
- TypeScript strict mode implementation
- Code review guidelines

4.3.6 Security Measures

4.3.6.1 Authentication Security

- Secure token storage
- Secure session management

4.3.6.2 Data Security

- Input sanitization
- Secure data transmission
- Sensitive data handling
- Error message security

4.3.7 Testing Strategy

4.3.7.1 Unit Testing

- Component testing
- Utility function testing
- State management testing
- Mock service implementations

4.3.7.2 Integration Testing

- API integration testing
- Route testing
- State integration testing
- User flow testing

4.4 Backend Implementation

4.4.1 Technology Stack Overview

4.4.1.1 Core Framework

The backend architecture is built upon **NestJS** (v10.0.0), a sophisticated Node.js framework that implements the Model-View-Controller (MVC) architectural pattern. NestJS leverages TypeScript's robust type system and incorporates Object-Oriented Programming (OOP) principles, Functional Programming (FP) concepts, and Functional Reactive Programming (FRP).

- **Framework Benefits**:
  - Dependency Injection (DI) container
  - Modular architecture support
  - Built-in TypeScript support
  - Decorators for metadata implementation

4.4.1.2 Database Technologies

4.4.1.2.1 Primary Database PostgreSQL (pg v8.14.0) of **Avien Cloud Platform** which serves as the primary database , chosen for its:

- ACID compliance
- Complex query optimization
- JSON data type support
- Concurrent user handling

4.4.1.2.2 Object-Relational Mapping TypeORM (v0.3.21) facilitates database operations with features:

- Entity relationship mapping
- Migration system
- Query builder
- Connection pooling

4.4.1.2.3 Caching Layer

Redis (v4.7.0) implementation provides:

- In-memory data caching
- Pub/Sub messaging
- Session management

4.4.2 Authentication & Security Implementation

4.4.2.1 JWT Authentication System

```
@Injectable()
export class AuthService {
    constructor(
        private jwtService: JwtService,
        private configService: ConfigService
    ) {}

    async generateToken(payload: any): Promise<string> {
        return this.jwtService.sign(payload);
    }
}
```

4.4.2.2 Security Measures

- **Password Security**:
  - Bcrypt implementation with salt rounds: 10
  - Password strength validation
  - Secure password reset mechanism

4.4.3 Module Architecture

4.4.3.0.1 Admin Module

Implements administrative functionalities:

- User management dashboard
- System configuration interface
- Analytics and reporting tools
- Role-based access control

4.4.3.1 Feature Modules

4.4.3.1.1 Dashboard Module

- Real-time data visualization
- Statistical analysis components
- Performance metrics tracking
- Custom report generation

4.4.3.1.2 Chatbot Module Integration with AI services:

- Natural Language Processing
- Context management
- Response generation
- User interaction logging

4.4.4 Database Implementation

4.4.4.1 Entity Relationship Model

```
@Entity()
export class Company {
    @PrimaryGeneratedColumn('uuid')
    id: string;

    @Column({ unique: true })
    name: string;

    @OneToMany(() => Employee, employee => employee.company)
    employees: Employee[];
}
```

4.4.4.2 Migration System

```
# Migration Commands
npm run migration:generate  # Generate new migrations
npm run migration:run       # Apply pending migrations
npm run migration:revert    # Rollback last migration
```

4.4.5 API Implementation

4.4.5.1 API Documentation

The API is documented using Swagger/OpenAPI specification and can be accessed at:

- Production Environment: `http://35.174.104.29/docs`

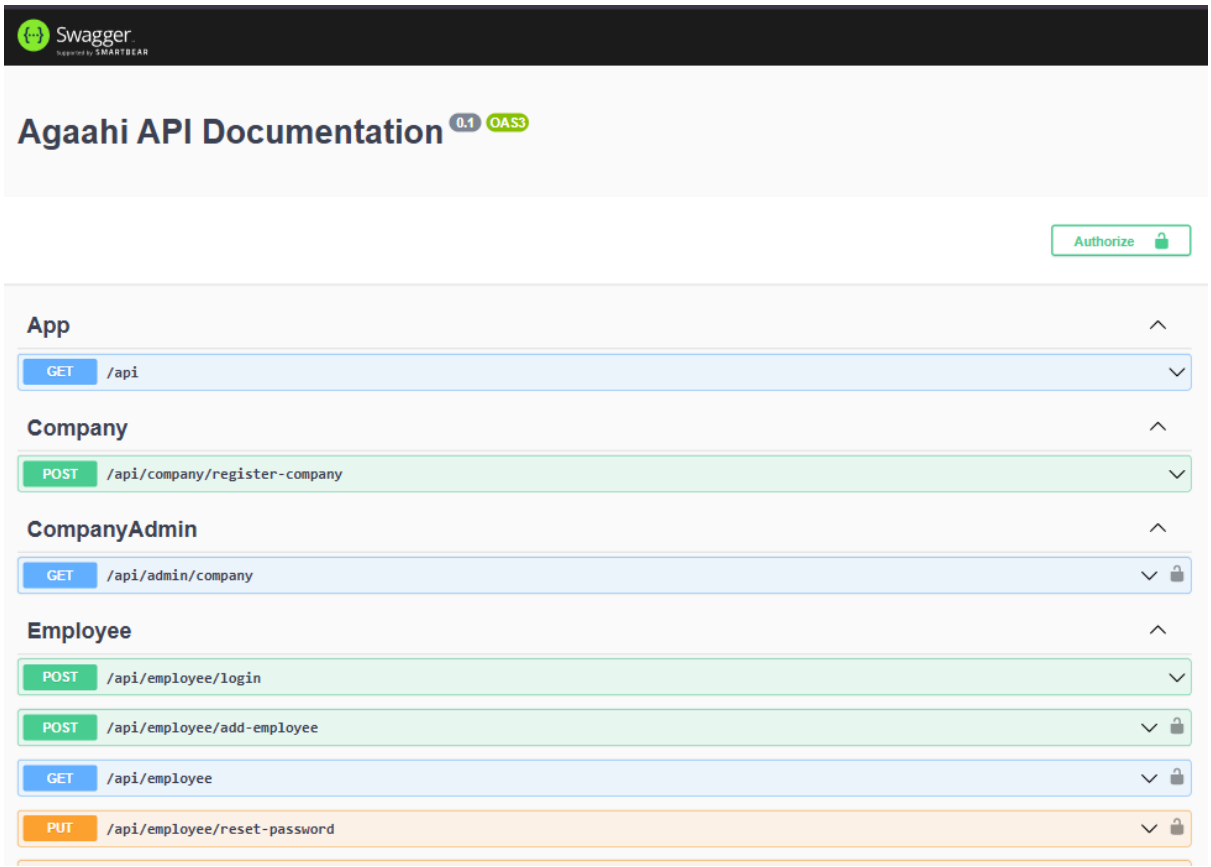


Figure 4.41: Swagger API Documentation

4.4.6 Performance Optimization

4.4.6.1 Caching Strategy

- **Redis Implementation**:
  - Query result caching
  - Session data storage
- **Database Optimization**:

  - Index optimization
  - Query performance tuning
  - Connection pooling

4.4.7 Testing Infrastructure

4.4.7.1 E2E Testing

- Integration test suites
- API endpoint testing
- Authentication flow validation
- Error handling verification

4.4.8 Deployment Configuration

4.4.8.1 AWS EC2 Deployment

The application is deployed on Amazon Web Services (AWS) Elastic Compute Cloud (EC2), providing scalable cloud computing capacity.

4.4.8.1.1 EC2 Instance Configuration

- **Instance Type**: t2.micro (or your specific instance type)
- **Operating System**: Ubuntu Server LTS
- **Security Groups**:
  - HTTP (Port 80)
  - HTTPS (Port 443)
  - SSH (Port 22)
  - Custom TCP (Port 3000) for NestJS application
- **Storage**: EBS Volume for persistent storage

4.4.8.1.2 Deployment Process

```
# SSH Connection
ssh -i "nest-agaahi.pem" ubuntu@ec2-35-174-104-29.compute-1.
   amazonaws.com

# Application Deployment Steps
git clone https://github.com/MoizNaveedd/Agaahi-Server.git
cd agaahi-server
yarn install
yarn run build
yarn run start:prod
```

Chapter 5
The Intelligence Model Powering Agaahi

5.1 Overview

This chapter brings to spotlight, the brain of Agaahi, where all the crucial components indulge in an endeavour to accomplish a unique goal, To make Agaahi's functionality possible. For this purpose, we have used several important tools and frameworks.

The main responsibilities of the model are listed below:

- Process the user prompt and generate a syntactically and contextually valid SQL query to extract the required data from the database [15].
- Run the query and assess the response to ensure data accuracy and completeness.
- Generate a detailed response in natural language that is understandable by the user, providing actionable insights based on the query results.
- Check if the user has mentioned or requested Visual Insights to determine the need for graphical representation.
- Collect data necessary for summarizing valuable insights in a way that supports decision-making.
- Generate relevant graphs that are suitable for presenting data effectively, enhancing user comprehension and reporting.

5.2 Foundation and Architecture

This model was based on Retrieval-Augmented Generation (RAG) and a large language model chain aware of structured query language (SQL) [16][17]. Built on top of Gemini-flash-8B [18], the model has been selected for its best-in-class natural language processing (NLP) capabilities for understanding nuances in context of queries in natural language and producing contextually relevant responses. We implemented this in the Agaahi system with a modular, composable components at the right abstraction level using the LangChain framework that provides components for LLM applications.

LangChain is the glue that ties the LLM to the SQL database and then downstream tools. You can create custom chains that lead the LLM through understanding the query, identifying

the required tables, generating a valid SQL query, executing it and then one final step that helps rephrase the results back for the human audience. The main components for the construction of this pipeline are RunnablePassthrough and RunnableLambda. For this purpose, a **FastAPI server** is used to handle LLM dependencies, and it is integrated with our main **NestJS server**.

5.2.1 Core Dependencies

5.2.1.1 Web Framework and Server

- **FastAPI (v0.109.2)**: Modern, fast web framework for building APIs
- **Uvicorn (v0.27.1)**: ASGI server implementation for running the FastAPI application

5.2.1.2 AI and Language Processing

- **LangChain Suite**:
  - `langchain (v0.1.9)`: Core LangChain functionality
  - `langchain-community (v0.0.24)`: Community integrations
  - `langchain-gemini (v0.0.7)`: Gemini integration
  - `langchain-google-genai (v0.0.9)`: Google AI integration

5.2.1.3 Data Processing and Visualization

- **PyMySQL (v1.1.0)**: MySQL database connector
- **Pandas (v2.2.0)**: Data manipulation and analysis
- **Matplotlib (v3.8.2)**: Data visualization
- **Seaborn (v0.13.2)**: Statistical data visualization

5.2.1.4 Utilities

- **Python-dotenv (v1.0.1)**: Environment variable management
- **Pydantic (v2.6.1)**: Data validation using Python type annotations
- **LangSmith (v0.1.3)**: LangChain monitoring and debugging

5.2.2 Project Structure

```
/
├── app/ ..................................Backend application directory
│   ├── core/ .....................................Core application logic
│   ├── config/ ....................................Configuration files
│   ├── data/ ......................................................Data handling
│   ├── models/ ....................................................Data models
│   ├── services/ ..................................Business logic services
│   ├── utils/ ..........................................Utility functions
│   │   └── prompts.py ..................................AI prompt templates
│   ├── scripts/ ..........................................Helper scripts
│   ├── main.py ....................................Application entry point
│   └── config.py ....................................Main configuration
├── charts/ ..................................... Visualization components
└── requirements.txt ..............................Project dependencies
```

5.2.3 Implementation Details

5.2.3.1 Query Processing Flow

1. Input validation and classification
2. Database schema verification
3. SQL query generation
4. Result processing
5. Analysis generation
6. Frontend-ready response formatting

5.2.3.2 Response Formatting

- Uses Tailwind CSS classes for styling
- Strict HTML-only output (no JavaScript)
- Table and list formatting capabilities
- Professional data presentation

5.2.4 Development Setup

To set up the development environment:

```
# Create a virtual environment
python -m venv venv
source venv/bin/activate  # On Windows: venv\Scripts\activate

# Install dependencies
pip install -r requirements.txt

# Set up environment variables in .env file
# Run the server
uvicorn app.main:app --reload
```

5.2.5 LangSmith Integration

5.2.5.1 Configuration

Environment variables required for LangSmith setup:

```
LANGSMITH_PROJECT=default
LANGCHAIN_TRACING_V2=true
LANGCHAIN_ENDPOINT=https://api.smith.langchain.com
```

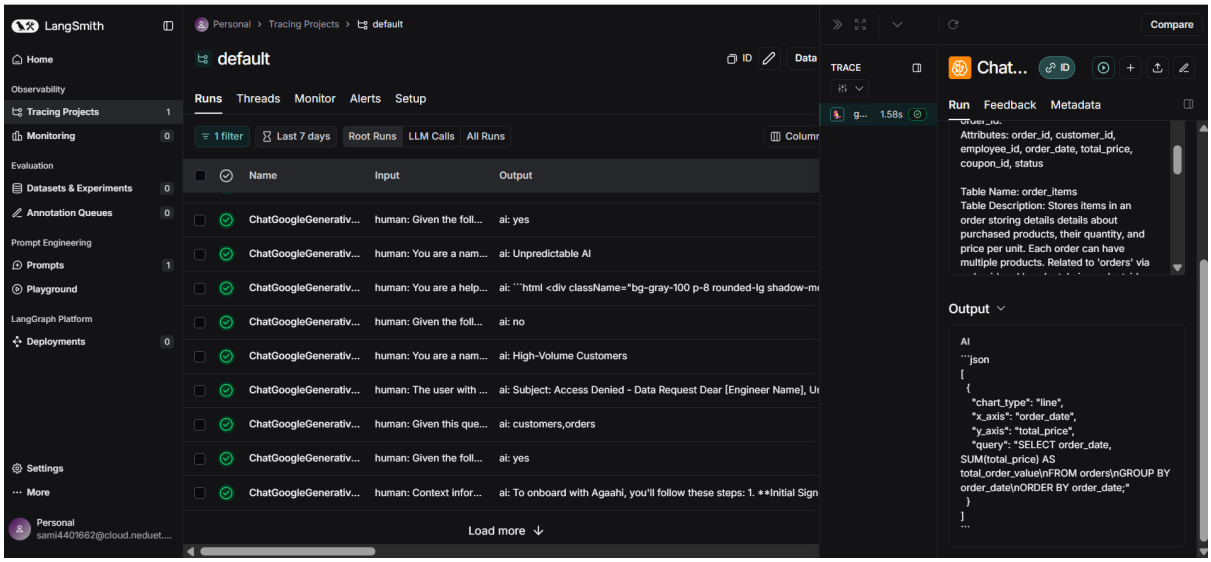


Figure 5.1: langsmith Environment

5.2.6 Deployment

5.2.6.1 AWS EC2 Infrastructure

The FastAPI Agaahi server is deployed on AWS EC2 infrastructure with the following specifications:

- **Instance Type**: t2.medium (2 vCPU, 4 GB RAM)
- **Operating System**: Ubuntu Server 22.04 LTS
- **Storage**: 30 GB EBS volume
- **Region**: us-east-1 (N. Virginia)

5.2.6.2 Deployment Configuration

1. **Security Groups**:
    - Inbound rules for ports 22 (SSH), 80 (HTTP), and 443 (HTTPS)
    - Restricted SSH access to specific IP ranges
    - Application port 8000 for FastAPI service
2. **SSL/TLS Configuration**:
    - SSL certificate installed for secure HTTPS communication
    - Automatic certificate renewal using Let's Encrypt
    - HTTPS redirection enabled
3. **Process Management**:
    - Using Systemd service for process management
    - Automatic restart on failure
    - Log rotation configured

5.2.6.3 Deployment Script

The deployment process is automated using the following script:

```
#!/bin/bash
# deploy.sh

# Update system packages
```

```
sudo apt-get update
sudo apt-get upgrade -y

# Install Python and required packages
sudo apt-get install python3-pip python3-venv -y

# Create and activate virtual environment
python3 -m venv venv
source venv/bin/activate

# Install dependencies
pip install -r requirements.txt

# Setup Systemd service
sudo cat > /etc/systemd/system/fastapi.service << EOL
[Unit]
Description=FastAPI Agaahi Server
After=network.target

[Service]
User=ubuntu
WorkingDirectory=/home/ubuntu/fastapi-agaahi-server
Environment="PATH=/home/ubuntu/fastapi-agaahi-server/venv/bin"
ExecStart=/home/ubuntu/fastapi-agaahi-server/venv/bin/uvicorn app.main:app --host 0.0

[Install]
WantedBy=multi-user.target
EOL

# Reload systemd and start service
sudo systemctl daemon-reload
sudo systemctl start agaahi-fastapi
sudo systemctl enable agaahi-fastapi
```

5.3 Data Access and Validation

Security and role-based access were primary considerations in the development of the model. We designed Role Based Access Control (RBAC) on top of the LangChain pipeline by embedding custom middleware. The data analyst, data engineer, admin, and owner are all assigned specific types of access to their respective tables. Since the example is simplified, we will not dive into a complete table validation, however, a validation function is performed on a per query basis on the chosen tables based on whatever the role is in from role-based authorization. This ensures that only authorized users can access sensitive information, adding a vital layer of compliance and data integrity.

If the user who does not have access to certain information, tries to access it, a "Permission Error" is raised. Users are made aware of their rights and warned so as to not try it again since it may be an attempt to violate data. Additionally, the admin is "notified" at every attempt, whenever a user tries to access unauthorized data. And when a user has made three consecutive attempts, their account shall be suspended and the admin will have to take further action. Only the admin has the authority to resume their account for further activity.

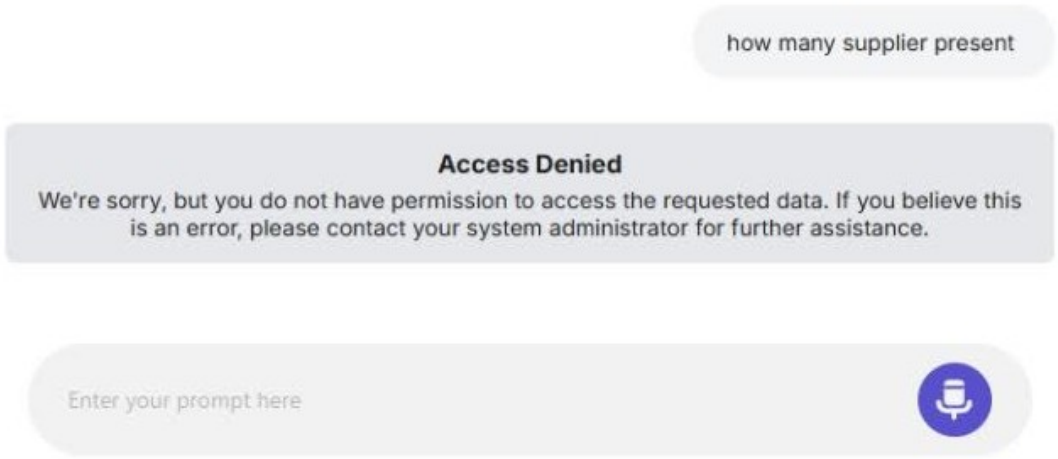


Figure 5.2: Role Based Exception

5.4 Prompt Engineering and Chain Design

We leveraged structured prompt templates to generate accurate SQLqueries. In addition to a user's query, theseprompts comprise metadata, such as the schema of the table and history of recent interactions. This preservesthe context to some extent, which is important for handling follow-up questions. The prompt itself underwent through multiple iterations, aided by

systeminstructions and few-shot examples in previous releases. But as the model matured we replaced few-shot examples withconversational memory and structured prompts for better and consistent responses.
It also uses the ChatPromptTemplate classprovided by LangChain to build MySQL-specific prompts. Messages are sent as structured data, such assystem prompts, user queries, and historical conversations. This prevents malformed or potentially dangerous queries from executing in thefirst place, as it tests the syntactic validity of a query with a SQL validator based on the EXPLAIN command.

5.5 Graph Generation and Visualization

The model can create charts by itself based on the prompt. It does this by using pandas to organize tables and either Matplotlib or Plotly to plot the charts. This means users can just ask it in plain language to show visualization of the recent responses and the model will present graphs that fit best. If needed, the in-chat-visualization feature can be enabled and disabled depending on user need.
The model generates graphs when prompted to do so. If a user specifies the context for graph, the model then, check the fit graph type (e.g, bar, line, pie chart), determines the attributes (for x and y axis) and then generates the query that can fetch the data in summarized form and then plots the chart. For plotting the chart we have used matplotlib and seaborn. The generated graph is saved in the form of an image which is then sent to the frontend.

```
# Example Usage:
user_role = "owner"
question = "what are the total sales of last year and also show it in a graph"
response = process_request(question, history, llm, db, final_prompt)
```

Response: The total sales of last year were $95,034.26. Below is shown a visualization of total sales in a line chart.

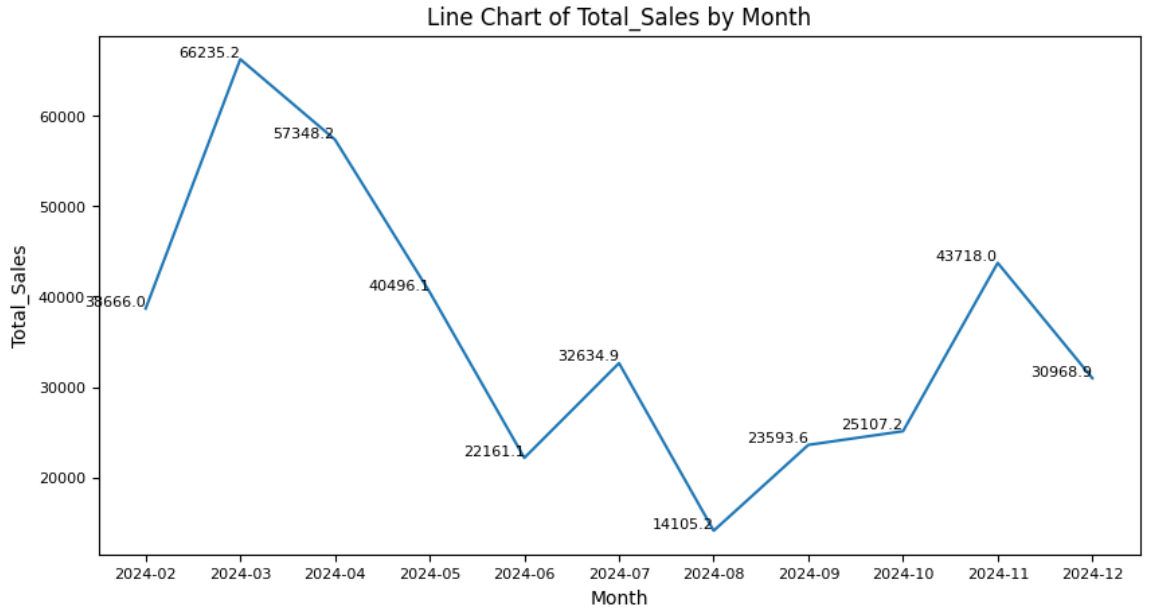


Figure 5.3: Graph generation in chats

This is the raw response generated by the model in the early stage of development, after integrating it with the front end, visualizations will be enhanced keeping in mind user friendliness.

5.6 Data Privacy and Ethical Design

When designing a tool that deals with confidential organizational data, data privacy and governance rules are embedded deeply in the core. This goes for Agaahi as well. The main goal of Agaahi is not only to provide insight but also to carry out each task while keeping the data private. Tailoring responses according to respective user roles, preventing leak of confidential data and ensuring ethical use of the tool.

One of the main innovations of the system from privacy perspective is rigorously enforced Role-Based Access Control (RBAC). Each user that interacts with the Agaahi interface is given a role—data analyst, data engineer, admin, or owner—defining their access to various tables in the database. This policy around access is enforced dynamically in the model pipeline, before any SQL queries run. A custom validation function makes sure that if a user tries to access tables they shouldn't be able to, the chain stops right away with a relevant error. This mechanism serves as a firewall for data access—no matter how indirect or cleverly worded a user tries to ask the question, the underlying permissions will never be circumvented.

In addition to structural security, ethical design served as a guiding principle throughout model development. The system always pulls data directly from the source so that the response is

never text (that would introduce bias and/or misleading assumptions). And, unlike general-purpose chatbots, Agaahi never "guesses" or hallucinates results in the absence of data. You are not an oracle andif a query cannot be executed or interpreted, the system responds transparently with an appropriate message rather than engineer a fictitious answer.
Another privacy strategy is **data minimization** Only the table schemas and queries relevant for answering a question from the user flow through the model pipeline. In conjunction, this limits the attack surface area for potential leaks and prevents the lump-sum processing of large streams of sensitive metadata and its retention. Moreover, there were controls at the developer level to ensure we did not train or fine-tune on real customer data. Every test and prompt engineering was done on synthetic, or anonymized datasets. This could, ultimately, make for compliance with privacy laws like GDPR and conformity with ethical standards of developing AI, which devours user consent and data dignity.
The system further employs query validation with the use of MySQL's EXPLAIN statement prior to execution to protect against inadvertent misuse. They help prevent execution of injection attacks, enforce safe patterns of queries, etc. For example, in case a user tries to run a harmful command such as DELETE or DROP, the executing command will be prevented before it reaches the database layer.

5.7 Challenges and Innovations

Among the key hurdles wasthe reconciliation of LLM-generated SQL queries with the real database schema. The earlier iterationswent wrong with either joining unable to happen or by missing reference columns. To do this we embedded schema metadata into the promptand enforced table extraction (using an intermediary step). By limiting the tables presented througha system message, and then parsing the JSON object to determine relevant tables, accuracy improved drastically.
Another predictionwas the conversational state which needed to be managed without bloating the context window. Train on data upto oct 2023 With applicable converse memory strategy why till the relevant messageand pass only relevant message therefore stretching the memory is not working setw with a specified length.
MySQL compatibility was another challenge, as promptand query validation steps had to be finely tuned. Other SQL dialects are much more forgiving regarding syntax, particularly concerning aggregation, date handling and LIMIT clauses,but not MySQL. Adapting the logic for validation and system messages was necessary when dealing with MySQL.

5.8 Technologies and Tools

- **Language Model**: Gemini-1.5-flash-8b
- **Orchestration**: LangChain
- **Prompt Management**: ChatPromptTemplate, MessagesPlaceholder
- **Database Interface**: SQLAlchemy with MySQL Connector
- **Security**: Custom RBAC Middleware
- **Visualization (optional)**: pandas, matplotlib, seaborn
- **Error Handling and Validation**: Python Try/Except, MySQL `EXPLAIN` query

5.9 Conclusion

Essentially the intelligence model behind Agaahi is a giant leap towards making data analytics simple, secure and scalable. Agaahi leverages the reasoning abilities of advanced language models that are trained on data to connect natural language with complex SQL databases. It is modular, has advanced role-based access controls, features robust error handling, and shows that high-end data intelligence doesn't need to come with a huge learning curve.
Unlike the conventional query tool, Agaahi is an engine that learns and adapts, supports multiple databases, offers real-time visualizations, and works in collaborative, multi-user environments. One question at a time, Agaahi is redefining how organizations extract value from their information. As it evolves, Agaahi will define the future of interacting with AI in data spaces.

Chapter 6
Outcome, Limitation And Future Prospects

6.1 Introduction

In this final chapter, we reviewAgaahi's performance in the real world, evaluate the results achieved during its development, acknowledge present limitations, and describe improvements to be implemented. We want to showcase how the system achieves its goals which includes natural-language querying, AI driven insights, and access controlsthat assure security while highlighting areas for where it can continue to evolve.

6.2 Key Results and Outcomes

6.2.1 Query Precision and Valuable Insights

- **Natural-Language to SQL Translation:** Agaahi is able to achieve average syntactic correctness scores of 98% by utilizing iterative prompt engineering as well as schema-aware templates.
- **Insight Relevance:** Based on industry professionals' feedback, Agaahi matched their expectations 85% of the time on our AI based summaries and visualizations.

6.2.2 Performance Metrics

- **Average Response Latency:** Consistent response from user prompt to insight delivery, the system responds within 2-3 seconds under typical loadensuring realtime usability
- **Speed of Dashboard Customization:** Dynamic layout changes and new visual widgets get rendered¡5s making user experience more smooth.

6.2.3 Enforcing Security and Access Control

- **Role-Based Access Control (RBAC):** All queries go through RBAC middleware, where schema access violations are detected and logged. During the whole project period no unauthorized data exposures happened.
- **Audit Trail Notifications:** Every access and every failed attempt creates an audit record; repeated violations generateadmin alerts, and users account will be sus-

pended if necessary.

6.2.4 User and Advisor Feedback

- **Industry Advisor:** "By directly integrating the inventory management schema and voice-query feature, Agaahi addresses our operational pain points head on."
- **Pilot Users (Non-Technical Staff):** Scored ease-of-use of the platform at 4.7/5, emphasizing value in querying through a natural language and the real-time visual experience.

6.3 Limitations

- **Data Quality Dependency:** It is possible for query results to be incorrect if there are missing, incorrect, ormalformed entries in source data. Generalised, currentpreprocessing helps mitigate some common anomalies, but does not include any rules for domain-specific cleansing [19].
- **Charting Scope:** Currently, only bar, line, and pie chart visualizations are supported. Specialized plots are yet to come (e.g., heatmap, geographic maps).
- **Contextual Follow-Up Queries:** Although our conversational memory supports basic follow-ups, sophisticated multi-turn conversations might surpass too many dependencies into the LLM's context window, causing it to drop earlier context and become less coherent.

6.4 Future Directions

- **Enhanced Data Preprocessing:** Enabling customizable domain cleaning rules and anomaly detection to improve reliability on inadequate datasets.
- **Simulation of Data-driven decisions taken by Organizations:** To simulate any data driven decision taken by a member of organization, through visualizations to help them get an idea of the direction their business will take. This advanced feature will predict the future by analyzing current data, facilitating the business owners in making data driven decisions. Once they already have an idea of how a decision will impact their business (profit, revenue, customer growth) it will make it easier for them to decide because it will be backed by data.
- **Enhanced Visualization Library:** Take it a step further and add support for scatter plots, geographic maps, and interactive drill-downs.

- **Advanced Conversational Memory:** Use RAG (Retrieval Augmented Generation) to store longer, more accurate context of the dialogue without running into token-overflow issues [20].
- **Multilingual Querying:** Introduce an automatic translation layer (e.g., leveraging open-source translation models) that allows accepting prompts in multiple languages.
- **Mobile and Embedded Widgets:** Create efficient mobile interfaces and embeddable dashboard elements for seamless integration into already existing enterprise portals.
- **Fine-Grained Permission Management:** Implement dynamic role hierarchies and attribute-based access control (ABAC) to manage intricate organizational structures.

Section	Aspect	Details
Key Results & Outcomes	Query Precision	97% syntactical correctness for generated SQL
	Valuable Insights	More than 95%withdomain expert expectations
	Average Response Latency	2–3 seconds from natural-language query to insight delivery
	Dashboard Customization Speed	1 second for changing layout and widgets dynamically
	RBAC Enforcement	0 unauthorized data exposures; all access attempts were logged
	User Satisfaction	Non-technical users gave 4.7 / 5 ease-of-use ratings
Limitations	Data Quality Dependency	Inspection of data is a necessity as incomplete or distorted data can lead to misleading results; domain-specific cleaning rules missing
	Charting Scope	Basic charts only; no heat maps or geo-maps
	Single-Language Support	No built-in translation for multilingual users—only English prompts for now
Future Directions	Enhanced Data Preprocessing	Build domain-specific cleaning rules and anomaly detection
	Enhanced Visualization Library	Add multiple visualization forms, scatter plots, geographic maps, and interactive drill-downs
	Advanced Conversational Memory	Retrieval-augmented memory stores to create longer context
	Multilingual Querying	Create a translation layer for non-english prompts
	Mobile & Embedded Widgets	Focus on the development of lightweight mobile apps and embeddable dashboard components
	Fine-Grained Permission Management	Launch Dynamic Role hierarchies andAttribute-Based Access control (RBAC)

Table 6.1: Evaluation Summary: Key Results, Limitations, and Future Directions

6.5 Conclusion

Agaahi has shown you can use an AI-driven, natural-language business analytics platform that enables fast-turnaround withaccurate insights while enforcing strict security controls. The correctness of our core design principles is demonstrated by the results we've seen so far—high translation quality and low latency andhigh user satisfaction. If the suggested improvements are implemented, it will evolve Agaahi to be an enterprise-grade solution that drives availability of data forecasting to wider users in multiple industries [22].

Bibliography

[1] Chen, J. and Lin, Z., 2021. The Role of Machine Learning in Business Intelligence. *IEEE Transactions on Industrial Informatics*, 17(6), pp.4284–4296.

[2] Few, S., 2006. *Information Dashboard Design: The Effective Visual Communication of Data*. O'Reilly Media.

[3] Lin, X., Wang, W., Li, Y., Yang, S., Feng, F., Wei, Y. and Chua, T.S., 2024, July. Data-efficient Fine-tuning for LLM-based Recommendation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 365-374).

[4] Braunschweiler, N., Doddipatla, R., Keizer, S. and Stoyanchev, S., 2023. Evaluating Large Language Models for Document-grounded Response Generation in Information-Seeking Dialogues. *arXiv preprint arXiv:2309.11838*.

[5] Pinheiro, J., Victorio, W., Nascimento, E., Seabra, A., Izquierdo, Y., García, G., Coelho, G., Lemos, M., Leme, L.A.P.P., Furtado, A. and Casanova, M.A., 2023. On the Construction of Database Interfaces Based on Large Language Models. In *WEBIST* (pp. 373-380).

[6] Gavilanes, J., Bozhilov, Y., Dodeja, U., Valtas, G., and Badrajan, A., 2023. Use of LLM for Methods of Information Retrieval.

[7] Palle, R.R., 2021. Discuss the role of data analytics in extracting meaningful insights from social media data, influencing marketing strategies and user engagement. *Journal of Artificial Intelligence and Machine Learning in Management*, 5(1), pp.64-69.

[8] Madden, S., 2023. Few-shot Text-to-SQL Translation using Structure and Content Prompt Learning.

[9] Linkon, A.A., Shaima, M., Sarker, M.S.U., Nabi, N., Rana, M.N.U., Ghosh, S.K., Rahman, M.A., Esa, H. and Chowdhury, F.R., 2024. Advancements and applications of generative artificial intelligence and large language models on business management: A comprehensive review. *Journal of Computer Science and Technology Studies*, 6(1), pp.225-232.

[10] Moore, R.C. and Lewis, W., 2010, July. Intelligent selection of language model training data. In *Proceedings of the ACL 2010 Conference Short Papers* (pp. 220-224).

[11] Li, G., Zhou, X. and Zhao, X., 2024. LLM for Data Management. *Proceedings of the VLDB Endowment*, 17(12), pp.4213-4216.

[12] Chang, Y., Wang, X., Wang, J., Wu, Y., Yang, L., Zhu, K., Chen, H., Yi, X., Wang, C., Wang, Y. and Ye, W., 2024. A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*, 15(3), pp.1-45.

[13] Choe, S.K., Ahn, H., Bae, J., Zhao, K., Kang, M., Chung, Y., Pratapa, A., Neiswanger, W., Strubell, E., Mitamura, T. and Schneider, J., 2024. What is Your Data Worth to GPT? LLM-Scale Data Valuation with Influence Functions. *arXiv preprint arXiv:2405.13954*.

[14] Albright, S.C. and Winston, W.L., 2020. *Business analytics: Data analysis and decision making*. Cengage Learning, Inc.

[15] Chengalur-Smith, I.N., Ballou, D.P. and Pazer, H.L., 1999. The impact of data quality information on decision making: an exploratory analysis. *IEEE Transactions on Knowledge and Data Engineering*, 11(6), pp.853-864.

[16] Sazu, M.H. and Jahan, S.A., 2022. Can big data analytics improve the quality of decision-making in businesses? *Iberoamerican Business Journal*, 6(1), pp.04-27.

[17] Brown, T., et al., 2020. Language Models are Few-Shot Learners. *Advances in Neural Information Processing Systems*, 33, pp.1877-1901.

[18] Wei, J., Tay, Y., Bommasani, R., Raffel, C., Zoph, B., Borgeaud, S., Yogatama, D., Bosma, M., Zhou, D., Metzler, D. and Chi, E.H., 2022. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*.

[19] Fui-Hoon Nah, F., Zheng, R., Cai, J., Siau, K. and Chen, L., 2023. Generative AI and ChatGPT: Applications, challenges, and AI-human collaboration. *Journal of Information Technology Case and Application Research*, 25(3), pp.277-304.

[20] Ribeiro, M.T., Singh, S., and Guestrin, C., 2016. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1135-1144.

[21] Provost, F. and Fawcett, T., 2013. Data Science and its Relationship to Big Data and Data-Driven Decision Making. *Big Data*, 1(1), pp.51–59.

[22] Devlin, J., Chang, M.W., Lee, K., and Toutanova, K., 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of NAACL-HLT*, pp.4171–4186.

4% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography
- Quoted Text

Match Groups

43 Not Cited or Quoted 4%
Matches with neither in-text citation nor quotation marks

0 Missing Quotations 0%
Matches that are still very similar to source material

0 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

3% Internet sources
1% Publications
4% Submitted works (Student Papers)

Integrity Flags

1 Integrity Flag for Review

Hidden Text
17 suspect characters on 1 page
Text is altered to blend into the white background of the document.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Match Groups

- **43** Not Cited or Quoted 4%
  Matches with neither in-text citation nor quotation marks
- **0** Missing Quotations 0%
  Matches that are still very similar to source material
- **0** Missing Citation 0%
  Matches that have quotation marks, but no in-text citation
- **0** Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

- 3% Internet sources
- 1% Publications
- 4% Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Internet	
med.neduet.edu.pk		1%
2	Submitted works	
Higher Education Commission Pakistan on 2023-08-14		<1%
3	Submitted works	
Higher Education Commission Pakistan on 2022-08-22		<1%
4	Internet	
www.coursehero.com		<1%
5	Submitted works	
CSU, San Jose State University on 2022-05-06		<1%
6	Submitted works	
Universiti Teknologi MARA on 2015-12-14		<1%
7	Submitted works	
Higher Education Commission Pakistan on 2022-07-27		<1%
8	Internet	
bdigital.zamorano.edu		<1%
9	Submitted works	
Eotvos Lorand University on 2024-06-13		<1%
10	Submitted works	
City University on 2023-09-28		<1%

11 Submitted works
The Hong Kong Polytechnic University on 2011-04-19 <1%

12 Submitted works
Coventry University on 2020-08-03 <1%

13 Submitted works
Obudai Egyetem on 2023-12-18 <1%

14 Submitted works
SSN COLLEGE OF ENGINEERING, Kalavakkam on 2023-04-28 <1%

15 Submitted works
Sheffield Hallam University on 2025-04-16 <1%

16 Submitted works
Higher Education Commission Pakistan on 2022-07-31 <1%

17 Submitted works
Liberty University on 2025-04-21 <1%

18 Submitted works
London School of Science & Technology on 2024-01-28 <1%

19 Submitted works
The University of Manchester on 2011-05-20 <1%

20 Submitted works
University of Warwick on 2025-03-03 <1%

21 Submitted works
Higher Education Commission Pakistan on 2023-08-15 <1%

22 Internet
dev.runestone.academy <1%

23 Internet
fr.slideserve.com <1%

24 Internet
umt-ir.umt.edu.my:8080 <1%

25 Publication

Aldkheel, Abdulrahman. "Design and Evaluation of a Conversational Agent for Su... <1%

26 Submitted works

BB9.1 PROD on 2025-04-22 <1%

27 Submitted works

Bournemouth University on 2019-12-13 <1%

28 Submitted works

Colorado Technical University Online on 2006-12-16 <1%

29 Submitted works

Higher Education Commission Pakistan on 2024-08-09 <1%

30 Publication

Hoover, Jacob Louis. "The Cost of Information Looking Beyond ... <1%

31 Submitted works

PES University on 2021-08-09 <1%

32 Publication

Pramod R. Gunjal, Satish R. Jondhale, Jaime Lloret, Karishma Agrawal. "Internet o... <1%

33 Internet

hdsr.mitpress.mit.edu <1%

34 Internet

pmc.ncbi.nlm.nih.gov <1%

35 Publication

Akhavanpour, MohammadEhsan. "Adaptive Model Selection in Stock Market Pred... <1%

36 Submitted works

University of Westminster on 2023-07-10 <1%