**Ai Agent**

**Design Document**

**Version 1.0**

****

**Group Id: F240233211**

**Supervisor Name : Usman Waheed Bhatti**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date (dd/mm/yyyy)** | **Version** | **Description** | **Author** |
| 20/11/2024 | 1.0 | AI agent in Node Js and React with a user and admin role, admin can upload Knowledge base and user can query from knowledge base if the answer is not found then a custom response will be sent. | BC220401569 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Table of Contents**

1. [Introduction of document](#gjdgxs)
2. [Scope (of the project)](#1fob9te)
3. [Functional Requirements Non Functional requirements](#3znysh7)
4. [Use Case Diagram](#2et92p0)
5. [Work Plan](#tyjcwt)
6. [Entity Relationship Diagram (ERD)](#3dy6vkm)
7. [Database Design](#1t3h5sf)
8. [Interface Design](#4d34og8)

1. **Introduction of Document**

This document provides a detailed description of the functional and non-functional requirements, Use case of the application, Workflow, ERD(Entity Relationship Diagram), Database Design Diagram and Screenshots of the UI which are developed in the React js for developing an AI Agent based on a Large Language Model (LLM). The AI Agent application is designed to assist users in retrieving information from a knowledge base uploaded by an admin. The document is structured as follows:

#### I. Purpose

#### The purpose of this document is to outline the requirements and specifications for developing an AI Agent application based on a Large Language Model (LLM). This AI Agent aims to assist users in retrieving accurate and relevant information by processing queries against a predefined knowledge base uploaded by the Admin.

#### The system is designed to:

#### Streamline the process of accessing information by providing direct responses to user queries.

#### Offer Admins control over the AI Agent’s configuration and behavior, including exception handling when the system cannot answer a query like when the system cannot find the answer or the restricted word is detected in the query (eg: nudity, weapon, booming etc ).

#### Improve user experience by delivering fast, reliable, and intuitive interactions through a beautiful interface.

#### 

#### II. Benefits of the AI Agent

#### This part outlines the advantages of the AI Agent, such as:

* Allowing users to retrieve specific information quickly and efficiently.
* Providing admins with straightforward control over the system’s behavior, knowledge base and restricted keywords (eg: weapon, nudity, sexual content etc).
* Handling exceptions in user queries seamlessly, enhancing user experience.

1. **Scope (of the project)**

This project focuses on the development of an AI agent powered by a pre-existing, free, and open-source Large Language Model (LLM). The AI agent will be designed to provide an efficient and interactive experience for users by leveraging a document-based knowledge base uploaded by an admin. The scope of this project includes the following:

**User Roles and Interfaces**

**Admin:** The admin will have the authority to configure and manage the AI agent's functionalities, including user interactions and responses to exceptional conditions.

The admin can upload text-based documents (e.g., .txt or .docx) as the knowledge base for the AI agent.

**User:** The user will interact with the AI agent through a web-based chat interface

**Core Functionality**

**Knowledge Base Querying:** The AI agent will extract information from the uploaded document to answer user queries.

**Fallback Mechanism:** When a query cannot be answered from the knowledge base, the agent will respond with a default message to inform the user about its limitations.

**Technology Stack**

The AI agent will be implemented using a programming language like Node js.

**Deliverables**

A functional AI agent capable of querying a document-based knowledge base.

An interface (console or web-based) for user interaction with the AI agent.

A simple mechanism for admin-controlled features and document management.

1. **Functional Requirements Non Functional requirements**

**Functional Requirements:**

Knowledge Base Management (Admin)

* The Admin shall be able to upload Word or text document files as the knowledge base.
* The system shall allow the Admin to update or replace the uploaded knowledge base file.
* The Admin shall be able to configure how the AI Agent responds when a query cannot be answered (i.e., default fallback message).
* The Admin shall be able to review and modify the settings for exception handling and query processing.

User Query Interaction

* The system shall provide a web-based chat interface for the User to interact with the AI Agent.
* The User shall be able to submit text-based queries.
* The AI Agent shall process and respond to the User’s queries based on the uploaded knowledge base.
* The AI Agent shall respond with a relevant answer derived from the knowledge base if available.
* If the query cannot be answered based on the knowledge base, the system shall provide a default message (e.g., "I couldn't find an answer to your question, please try again with a different query.").
* Users can create a new conversation and this conversation will be save for later review.

System Query Processing

* The AI Agent shall be able to parse and search the knowledge base (Word or text document) to find relevant information.
* The system shall return answers from the knowledge base that are most closely related to the User’s query.

**Non-Functional Requirements:**

Usability

* The Web based User interface shall be simple, intuitive, and easy to use.
* The Admin interface for managing the knowledge base and configuring settings shall be user-friendly and provide sufficient feedback for each action (e.g., upload success, error messages).

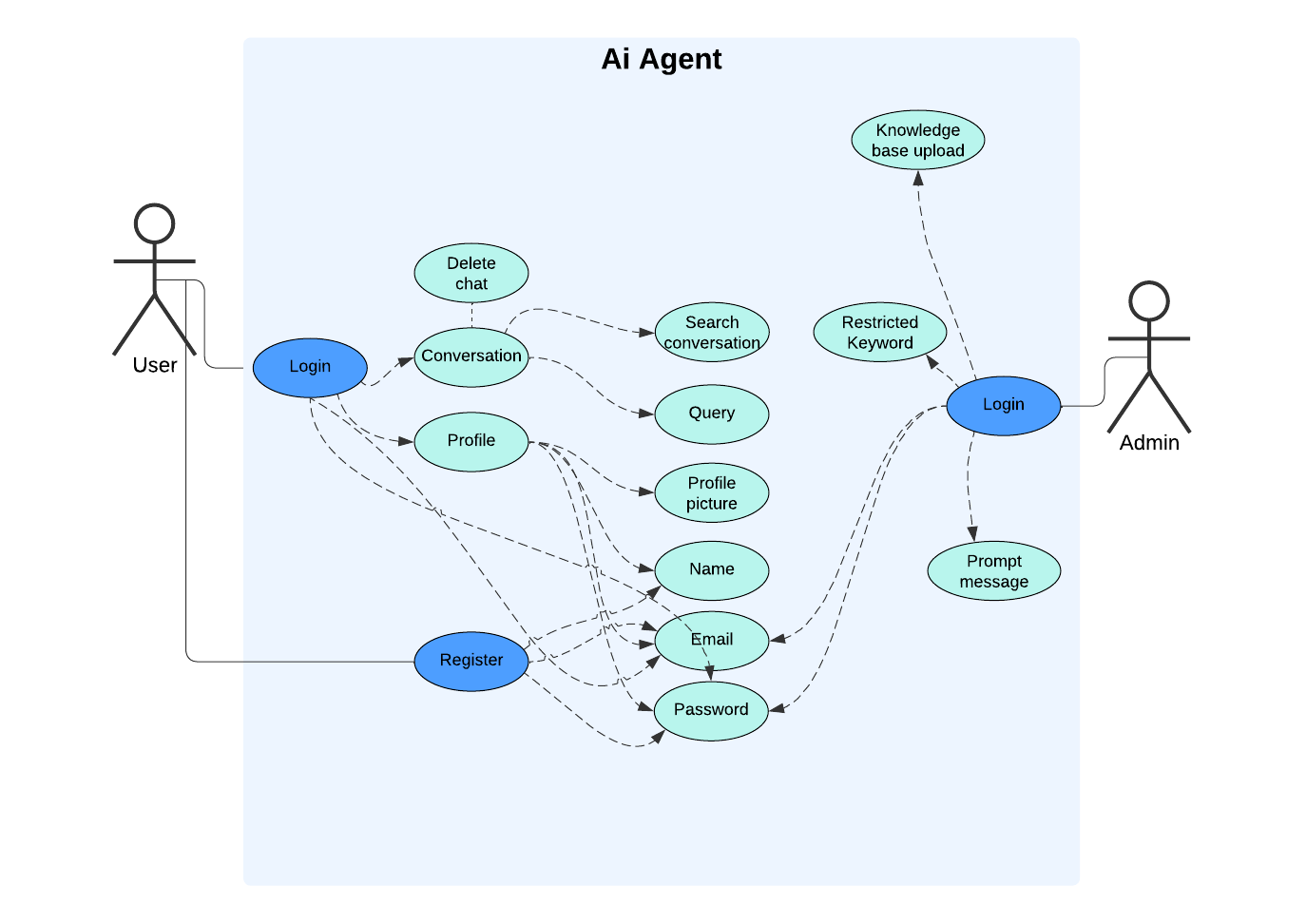
Security

* The system shall ensure that only authorized Admin users can upload or modify the knowledge base.
* User queries shall be handled securely, and no personal information shall be collected.
* The system shall have basic protection against malicious input or attacks (e.g., SQL injection).

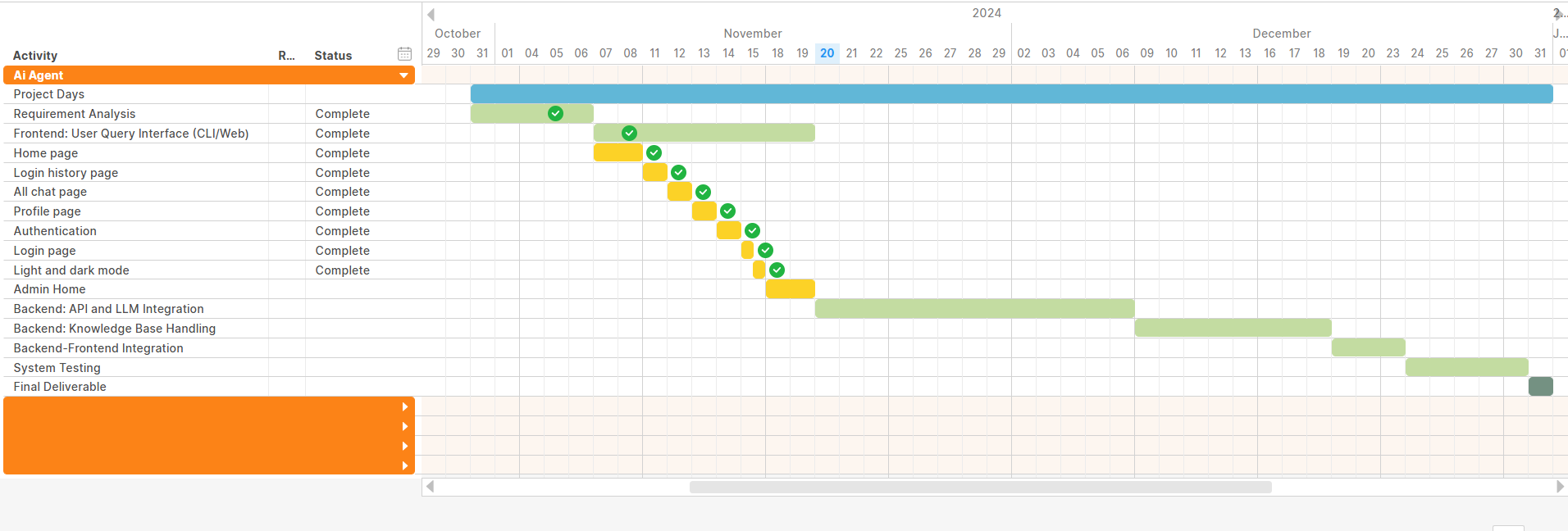
Compatibility

* The web-based interface shall be compatible with modern browsers (Chrome, Firefox, Edge).

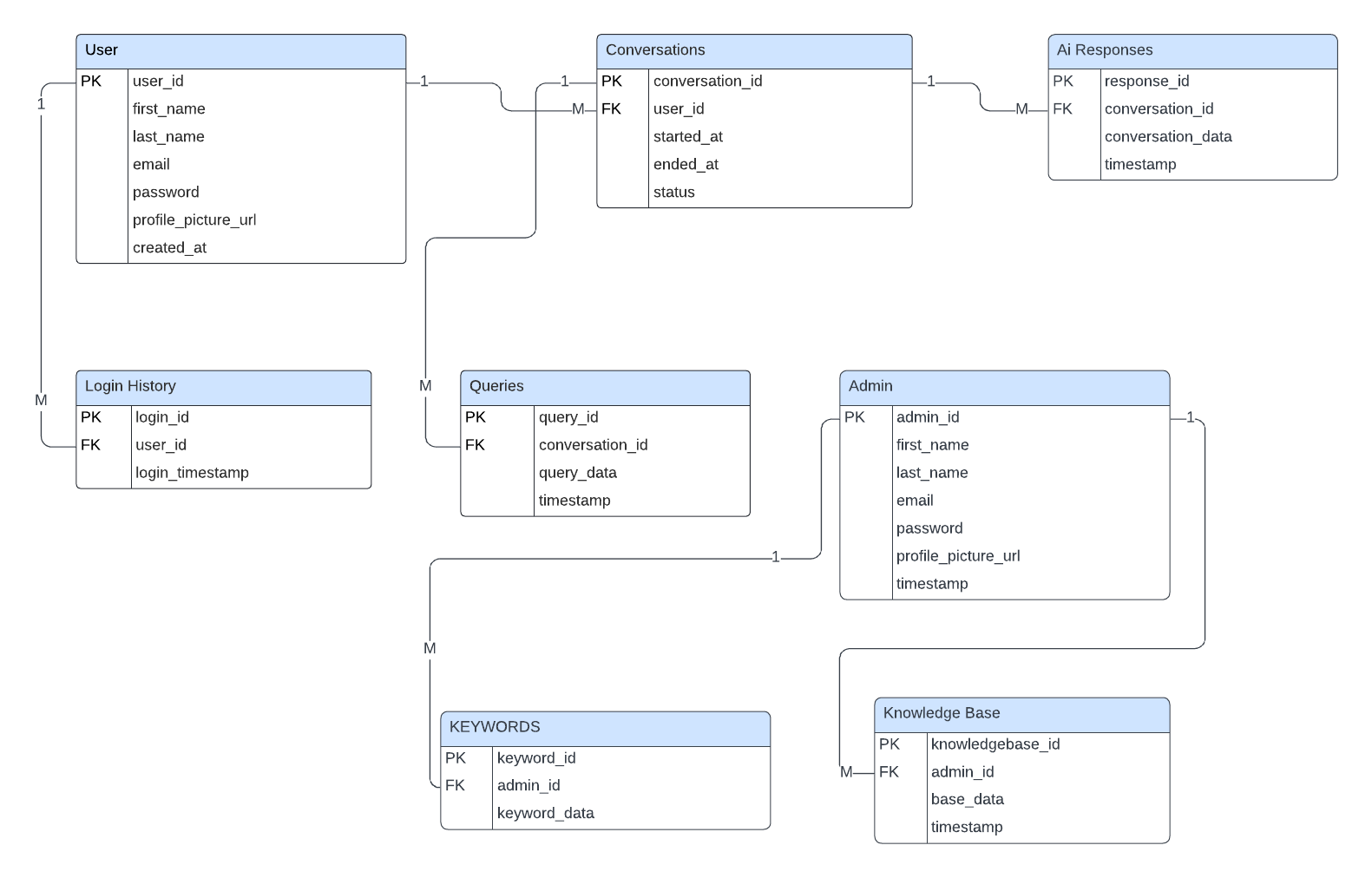
1. **Use Case Diagarm**



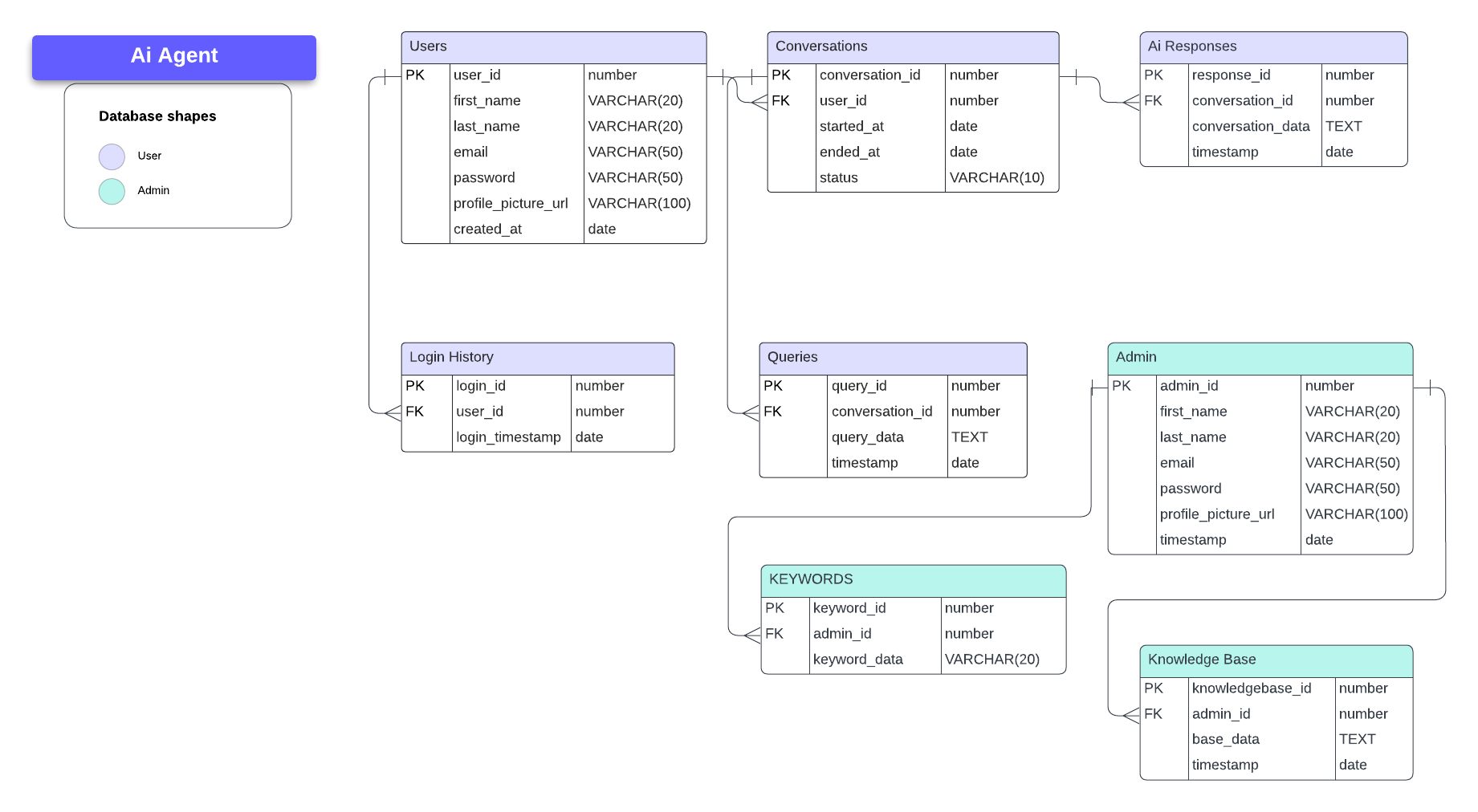
1. **Work Plan**



1. **Entity Relationship Diagram (ERD) (To be developed using Microsoft Visio or any other drawing software of your choice)**



1. **Database Design**



1. **Interface Design**

