**RAG MODEL (Retrieval-Augmented Generation)**

**Overview**

* **Purpose:** Retrieval-Augmented Generation (RAG) enhances student learning by providing accurate, context-rich answers by combining real-time information retrieval with natural language generation—helping students understand complex topics more clearly, complete research faster, and develop critical thinking through exposure to reliable sources.
* Accepts user queries and retrieves relevant documents from a knowledge base.
* Uses a language model to generate responses that are both context-aware and factually grounded in the retrieved content.
* Showcases seamless integration with user interfaces and real-time interaction capabilities.
* Supports multi-turn interactions, allowing the model to maintain context across multiple questions or dialogue rounds.

**The model architecture likely involves a dual-phase pipeline:**

* Retriever: Gathers top-k relevant documents from an external corpus using vector similarity or semantic search.
* Generator: Produces a fluent, contextually enriched response using the retrieved content, typically powered by a transformer-based model.

**Benefits**

* **Increased Accuracy:** Responses are grounded in factual sources, reducing hallucinations typical of standard language models.
* **Domain Adaptability:** Easily integrates with domain-specific corpora (legal, healthcare, academic, etc.).
* **Real-Time Response Generation:** Capable of retrieving and generating answers in near real-time for user queries.
* **Explainability:** Since retrieved documents can be traced, the model offers transparency in how answers are formed.
* **Security and Authentication**: Implements JWT (JSON Web Token) authentication to ensure secure, stateless, and scalable user verification. This protects API endpoints, enables role-based access, and ensures that only authorized users can interact with sensitive components of the system.

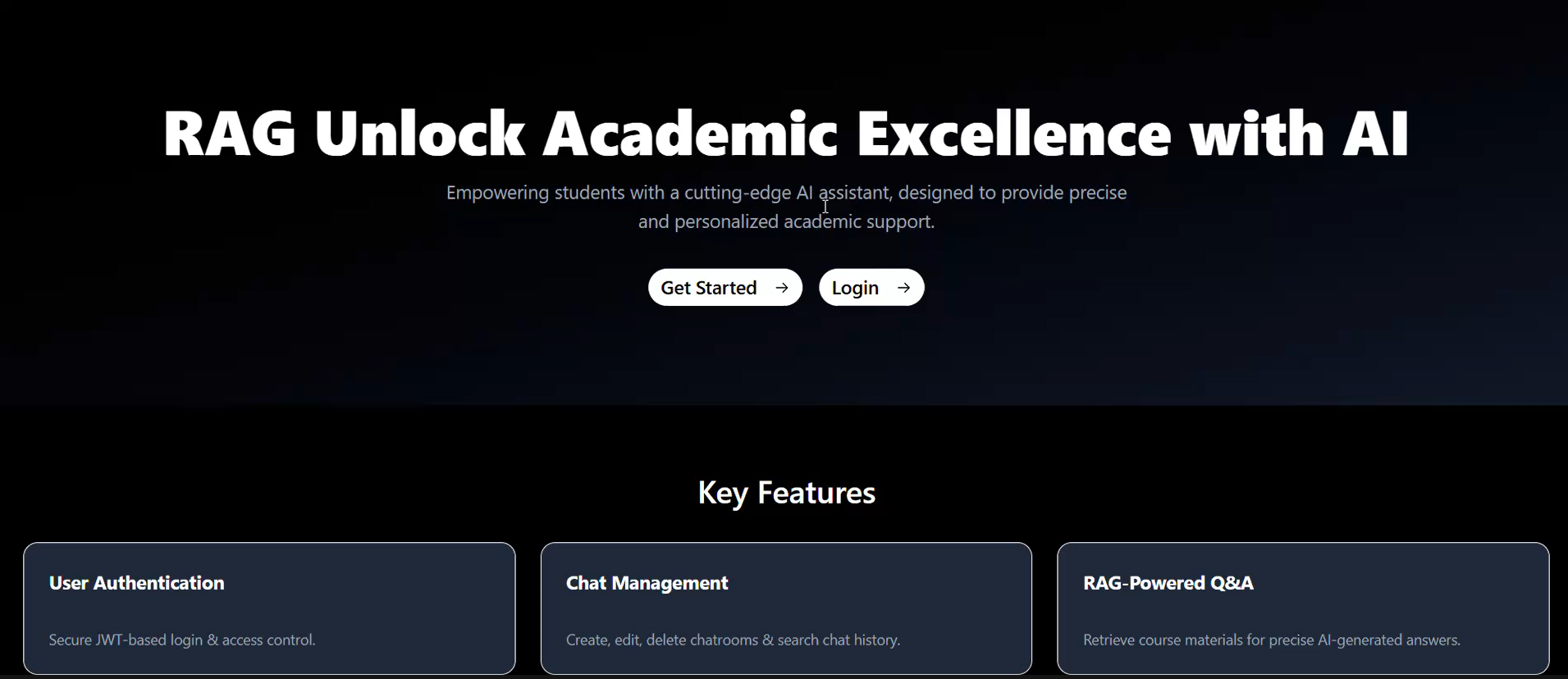
**Tech Stack**

**Frontend:** React, Tailwind CSS

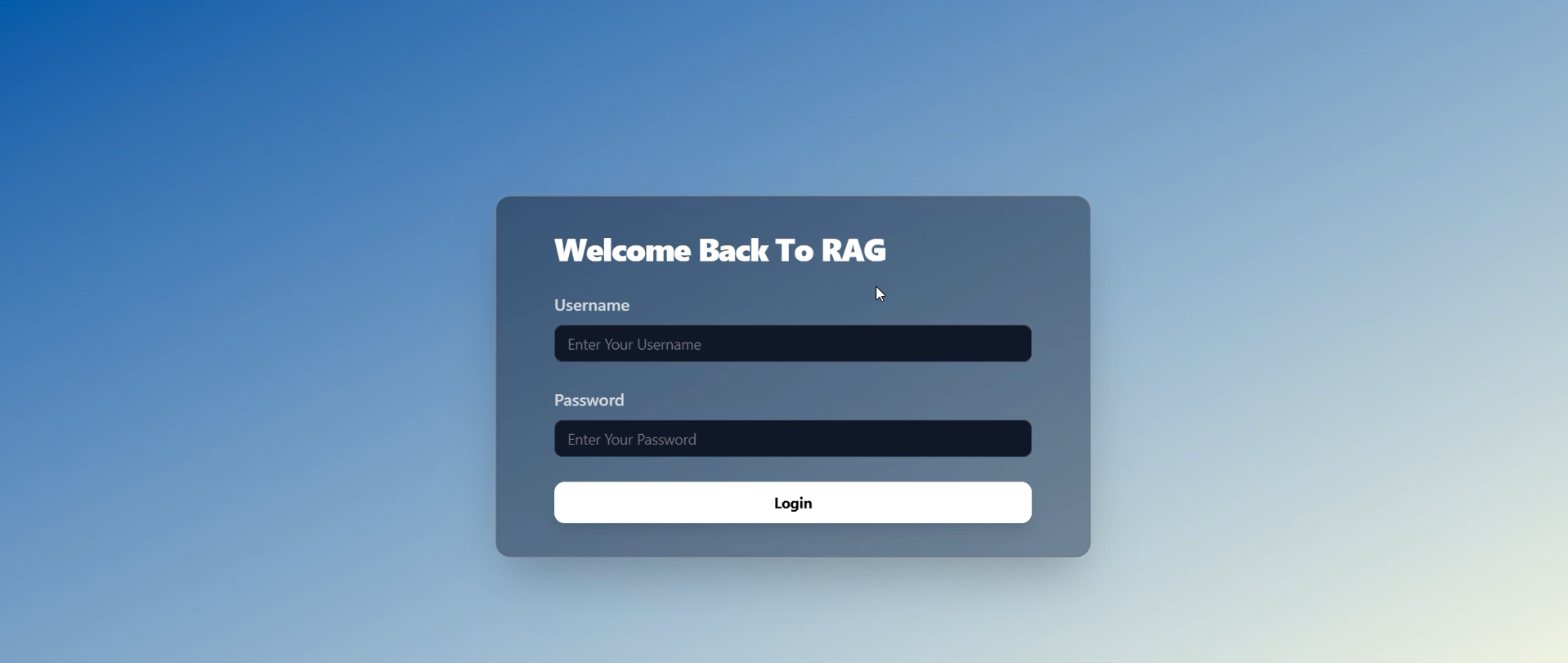
**Backend:** Spring, Spring boot, Hibernate, PostgreSQL

**Tools:** Git, Github, Postman API

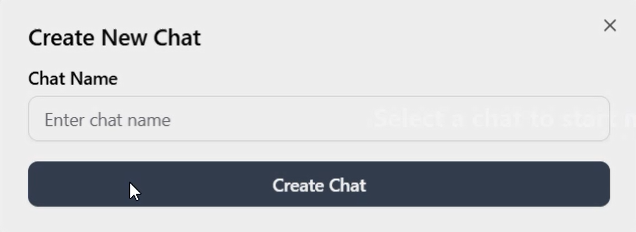
**Landing Page**

****

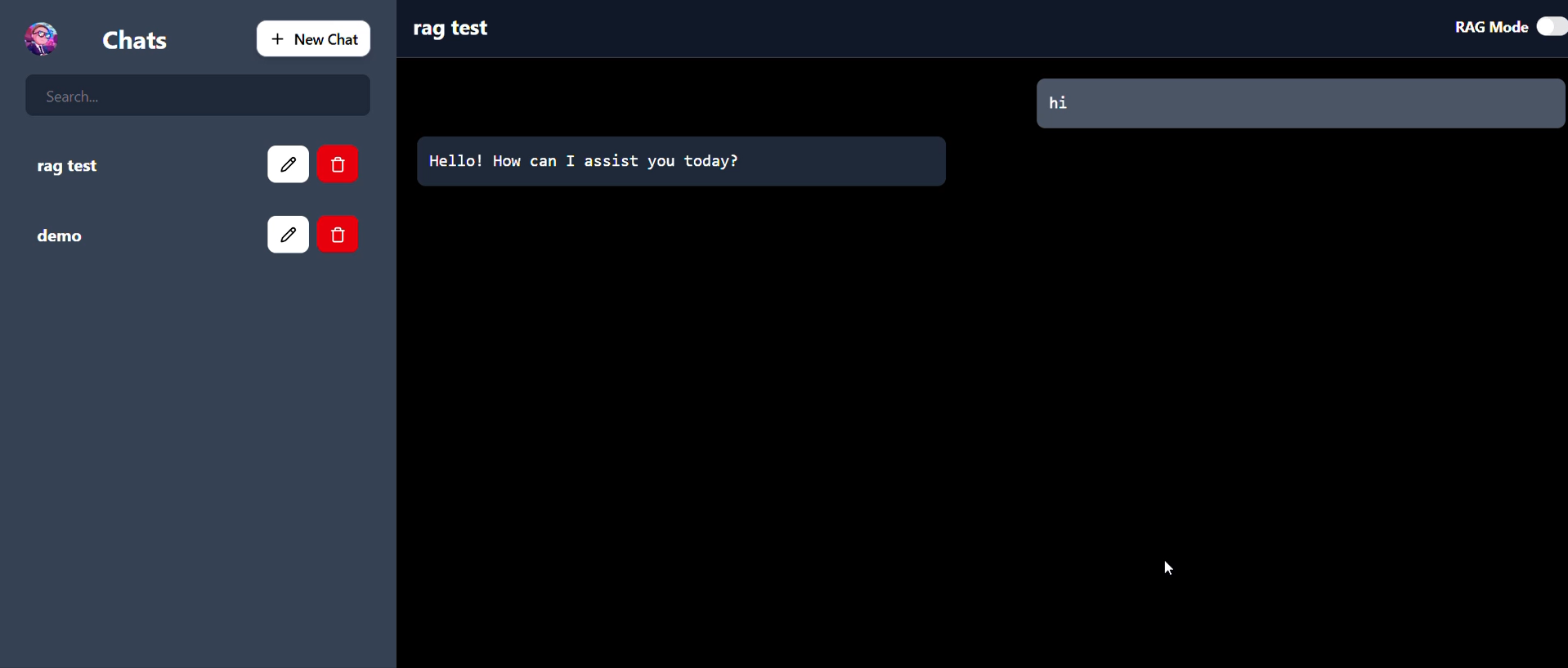
**Authentication Page**

****

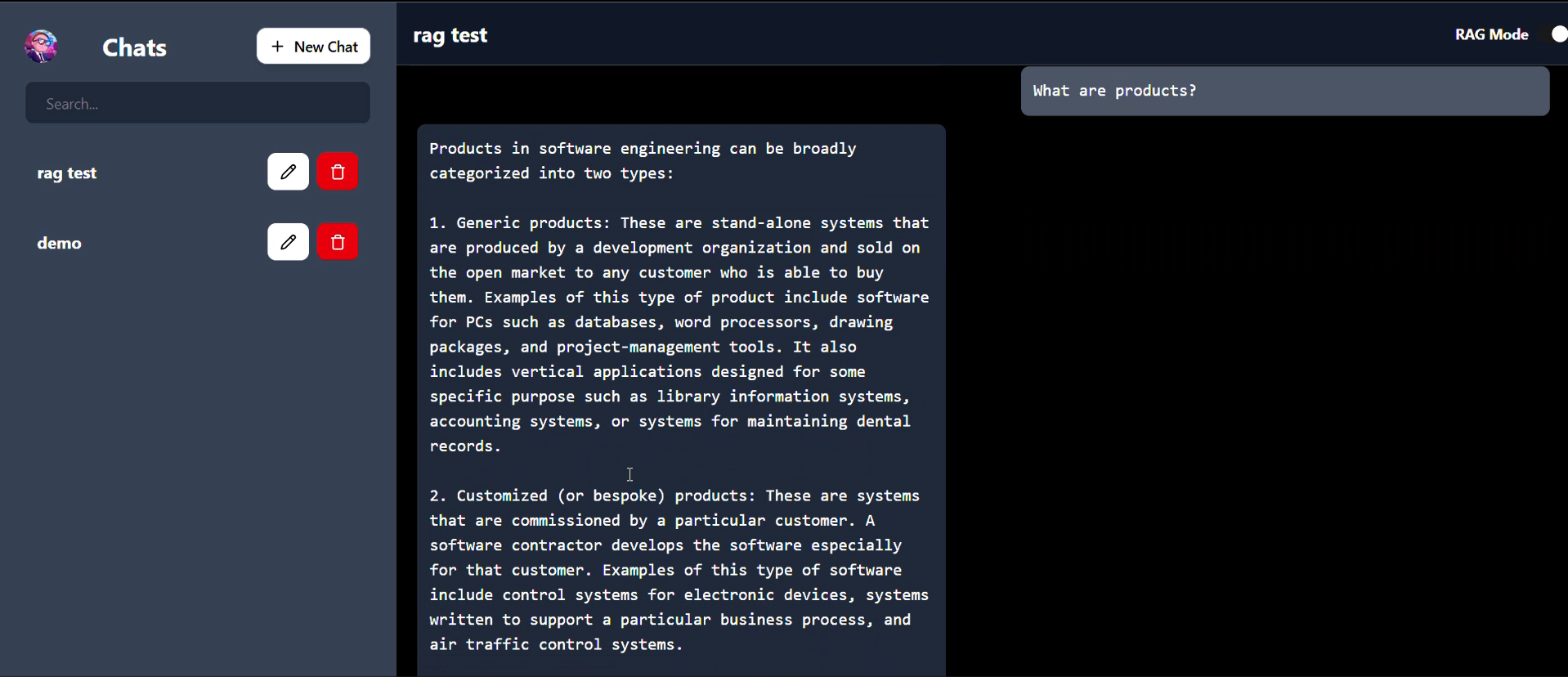
**Creating Chatroom**

****

**Chatroom**

****

**RAG Mode**

****