**Development Tools and Technologies**

* **Frontend:** React.js with Material UI (MUI) for styling
* **Backend:** Node.js with Express.js
* **Database:** MongoDB
* **Chatbot:** Python (Flask) using OpenAI & Pinecone
* **Vector Database:** Pinecone for storing and retrieving embeddings
* **Embedding Model:** Hugging Face embeddings
* **Deployment:** Vercel

**Development Methodology**

* **Agile Development:** Iterative approach with continuous feedback and improvements
* **Microservices Architecture:** Separate backend services for chat processing and user interactions
* **RESTful API Design:** The chatbot interacts with the frontend using HTTP requests
* **Version Control:** GitHub for code management

**Programming Languages and Tools**

* **Languages:** JavaScript (React, Node.js), Python (Flask, LangChain)
* **Backend Frameworks:** Express.js (Node.js), Flask (Python)
* **Frontend Library:** React.js with Material UI
* **Database:** MongoDB (NoSQL)
* **Vector Search:** Pinecone
* **API & AI Integration:** OpenAI (for chatbot responses), LangChain (for RAG model)
* **Environment Management:** dotenv for managing API keys

**Third-Party Components and Libraries**

* **Frontend:**
  + Material UI (MUI) for styling
  + Axios for API requests
* **Backend:**
  + Express.js for handling HTTP requests
  + MongoDB with Mongoose ORM
  + Flask for chatbot API
  + Pinecone for vector database storage
  + LangChain for retrieval-augmented generation (RAG)
  + OpenAI API for chatbot responses
* **Others:**
  + dotenv for environment variables
  + Hugging Face models for embeddings

**Algorithms**

* **Text Embedding:** Uses a Hugging Face embedding model to convert PDF text into vectors
* **Similarity Search:** Pinecone’s cosine similarity algorithm to find the most relevant document
* **Retrieval-Augmented Generation (RAG):** Combines retrieved document chunks with OpenAI’s LLM to generate responses
* **Tokenization:** Splitting documents into smaller chunks for efficient retrieval
* **Natural Language Processing (NLP):** Enhancing chatbot responses with AI-based understanding

**Software/Hardware Requirements**

**Software Requirements:**

* **Frontend:** Node.js, React.js, Material UI
* **Backend:** Node.js, Express.js, Python, Flask, MongoDB
* **Dev Tools:** VS Code, Postman, Git
* **API Keys:** OpenAI API, Pinecone API

**Hardware Requirements:**

* **Development Machine:**
  + Minimum: 8GB RAM, Core i5, SSD
  + Recommended: 16GB RAM, Core i7/i9, NVMe SSD