Agile Requirements – Pinecone RAG Chatbot

# Application Purpose

* 1. As a user, I want to ask questions about uploaded documents, so that I can retrieve relevant information quickly using AI.

# Backend / API Functionality

* 1. As a user, I want to enter a natural-language question, so that the backend can send it to OpenAI and Pinecone for an intelligent response.
  2. As a user, I want to upload PDFs, Word docs, or text files, so that their content can be embedded and queried later.
  3. As a user, I want to receive responses from the chatbot in a streaming format, so that I can read the answer as it is being generated.
  4. As a user, I want answers to include document citations or source snippets, so that I can verify the AI’s response.

# Frontend (React UI)

* 1. As a user, I want to interact with a clean, responsive chat UI, so that I can ask and receive answers in a familiar format.
  2. As a user, I want chatbot responses to support markdown formatting, so that code, bold text, and links appear properly.
  3. As a user, I want to upload files from the browser, so that I can enrich the chatbot with custom documents.
  4. As a user, I want to see my past questions and answers, so that I can reference previous chats easily.

# Configuration and Environment

* 1. As a developer, I want to manage API keys and configuration through a .env file, so that I can securely connect to Pinecone and OpenAI.
  2. As a developer, I want to host the frontend on Firebase, so that users can access the chatbot via a public web app.
  3. As a developer, I want to deploy the Flask backend to GCP Cloud Run, so that it scales securely and integrates with the frontend.

# CI/CD and DevOps

* 1. As a developer, I want a Jenkins pipeline or Cloud Build config, so that my app is tested and deployed automatically on commit.
  2. As a developer, I want errors to be logged in the backend, so that I can troubleshoot issues in production.

# Stretch Goals (Optional)

As a user, I want the chatbot to remember previous context, so that I can have more natural, multi-turn conversations.

As a user, I want to choose between different LLM models, so that I can balance speed, cost, and accuracy.

As a user, I want to speak my questions using a microphone, so that I can use the chatbot hands-free.