DocuChat AI - Talk to Your Documents

A smart document assistant where users can upload PDFs or images, and then chat or talk with their documents to extract, understand, and summarize their contents using AI.

Tech Stack

Frontend

- React (Vite)
- Tailwind CSS for styling
- Zustand for global state management
- React Router for routing
- Web Speech API (SpeechRecognition + SpeechSynthesis)

Backend

- Node.js + Express
- MongoDB + Mongoose
- JWT + bcrypt for authentication
- pdf-parse for PDF extraction
- Tesseract.js for image OCR
- Gemini API for AI chat and document processing (updated from OpenAI)
- Nodemailer to send chat summaries to user's email

Structure & Conventions

- Uses type: "module"
- Follows naming convention like: auth.route.js, user.model.js, doc.controller.js, etc.

Core Features

1. User Authentication

- Signup/login using JWT
- Authenticated users can only access their own documents & chats
- Secure password hashing with bcrypt

2. Upload Document

- Accept PDFs and images (JPG/PNG)
- Extract: PDF pdf-parse, Image Tesseract.js
- Save to DB: { userId, title, filename, extractedText, createdAt }
- Redirect user to document chat view

3. Two Chat Modes

Chat with Document

- Type questions to chat with the uploaded document
- Backend sends question + extracted text to Gemini API
- Response from AI is stored and displayed

Talk to Document

- Use microphone (SpeechRecognition API) to speak questions
- AI replies with speech using SpeechSynthesis
- Optional: display transcript in chat

4. Profile / History Page

- View list of uploaded documents (Title, type, date)
- Click to reopen chat session or continue conversation

- Optionally: Rename or delete the document

5. Summarize This Chat

- Button: "Summarize this chat"

- Uses Gemini to summarize chat history

- Actions: Copy summary, Download .txt, Send to user's email via Nodemailer

6. Doc Toolbox with AI Actions

- Floating toolbox on document chat view:

  Extract bullet points

  Rephrase in simple terms

  Convert to checklist

  Copy all key points

  Export all data (text or JSON)

Milestone Roadmap

Milestone 1: Auth System

Backend

- Setup Express server with ESM

- JWT & bcrypt setup

- Routes: /api/auth/register, /api/auth/login

- Auth middleware

Frontend

- Create auth pages: Login & Register

- Zustand store for user auth
- Token management (login protected routes)

Milestone 2: Document Upload & Text Extraction

Backend

- Route: /api/docs/upload
- File handling with Multer
- Extract PDF pdf-parse, Image Tesseract.js
- Save doc with metadata & text

Frontend

- Upload page
- Form with title + file input
- Show loading redirect to chat on success

Milestone 3: Chat with Document

Backend

- Route: /api/docs/:id/chat
- Take: user message + extracted text Gemini
- Store Q&A in DB under the document

Frontend

- Chat UI
- Zustand useChatStore
- Scrollable, styled Q&A
- Typing input + loading indicators

Milestone 4: Talk to Document (Voice Mode)

Frontend Only

- Voice input via SpeechRecognition API

- AI replies with SpeechSynthesis

- Optional transcript displayed

Milestone 5: Profile / History Page

Backend

- Get all documents by user

- Get chat history for a document

- Delete or rename document

Frontend

- Profile page UI

- Card/list view of uploaded docs

- Buttons: Open, Rename, Delete

Milestone 6: Summarize This Chat

Backend

- Route: /api/docs/:id/summarize

- Use Gemini to process chat history

- Use Nodemailer to email summary

Frontend

- Button: Summarize this chat

- Modal with: Summary preview, Copy, download, or email

Milestone 7: Doc Toolbox with AI Actions

Backend

- Routes: /api/docs/:id/extract-bullets, /rephrase, /checklist, /export

- Use Gemini to transform content

Frontend

- Floating toolbox UI on chat view

- Modal/drawer to show output

- Copy/export buttons