**🧠 Project Summary: Full Local Chatbot with LLaMA, Auth, and ChatGPT-Style UI**

You're building a **fully local AI chatbot** powered by **Meta's LLaMA**, with a UI experience similar to **ChatGPT**, plus full **authentication** and extra smart features.

**🏗️ Key Features**

**💬 Chatbot Functionality**

* Runs **Meta’s free LLaMA model locally** (via Ollama).
* Uses **RAG (Retrieval-Augmented Generation)** with **ChromaDB** or **FAISS** for knowledge retrieval.
* Local embedding generation using all-MiniLM-L6-v2.

**🎨 Frontend**

* **ChatGPT-style UI** using **React + Tailwind CSS** or **Next.js** for better routing and SSR.
* Clean chat bubbles, input field, and responsive layout.

**🔐 Authentication (Auth.js)**

* **Login & Register** using:
  + OAuth (Google, etc.)
  + Traditional credentials
* **Validation Rules**:
  + **Username**: ≤10 characters, must include a **capital letter**
  + **Password**: ≥8 characters, must include:
    - A **capital letter**
    - A **special character**
    - **Numbers**

**📚 Vector DB for RAG**

* **ChromaDB or FAISS** stores knowledge chunks and retrieves relevant info on user queries.

**✅ Smart Additions (Now Included)**

* **Session Management**: Store sessions securely using local storage or cookies.
* **Rate Limiting**: Prevent API/LLM overload from rapid-fire queries.
* **Markdown & Code Rendering**: Pretty output with syntax highlighting for code.
* **Theme Toggle**: Light/Dark mode support (just like ChatGPT).

**📦 Final Stack Overview**

| **Layer** | **Tech** |
| --- | --- |
| Frontend | React + Tailwind / Next.js |
| Backend | FastAPI or Flask |
| LLM | Meta LLaMA (via Ollama) |
| Auth | Auth.js (OAuth + credentials) |
| Vector DB | ChromaDB or FAISS |
| Embeddings | SentenceTransformers (MiniLM) |
| Extra Features | Session, rate limit, markdown, theme |