

Chatbot AI Platform

Modern full-stack chatbot platform with user accounts, multi-project (agent) management, stored prompts, and a chat UI
powered by Groq LLM.

Monorepo Layout

- client/ — React + Vite + Tailwind UI
- server/ — Node.js + Express + MongoDB API

Prerequisites

- Node.js 18+
- npm 9+
- MongoDB (local or Atlas)

Quick Start

1. Clone and install

```
# Clone your repo (replace with your Git URL)
git clone https://github.com/Avishkar2004/chatBot-AI
cd chatbot-ai

# Install deps
cd server && npm install && cd ..
cd client && npm install && cd ..
```

2. Configure environment variables

- Copy server/.env.example to server/.env and set values:

```
MONGODB_URI=mongodb://localhost:27017/chatbot_ai
PORT=8000
JWT_SECRET=replace_with_a_long_random_string
JWT_EXPIRES_IN=7d

# Groq
GROQ_API_KEY=your_groq_key_here
GROQ_MODEL=llama-3.1-8b-instant
```

- Optionally create client/.env:

```
REACT_APP_API_BASE=http://localhost:8000
```

3. Run the servers (in two terminals)

```
# Terminal 1: API
cd server
npm run dev

# Terminal 2: Client
cd client
npm start / npm run dev
```

- API: http://localhost:8000
- Web: http://localhost:3000

Login & Test

- Register a user on the web app (Sign Up)
- Login to obtain a session (JWT stored in localStorage)
- Create a project in Projects
- Add prompts in the project detail page
- Open Chat for that project and send a message

API Overview

- Auth
 - POST /api/auth/register
 - POST /api/auth/login
 - GET /api/auth/me
- Projects (Bearer token)
 - GET /api/projects
 - POST /api/projects
 - GET /api/projects/:projectId
 - PUT /api/projects/:projectId
 - DELETE /api/projects/:projectId
- Prompts
 - GET /api/projects/:projectId/prompts
 - POST /api/projects/:projectId/prompts
 - PUT /api/projects/:projectId/prompts/:promptId
 - DELETE /api/projects/:projectId/prompts/:promptId

- Chat (Groq)
 - POST /api/projects/:projectId/chat →

Scripts

- Server
 - npm run dev — start Express in dev (nodemon)
 - npm start — start Express in prod
- Client
 - npm start — run start

Documentation

- See ARCHITECTURE.md for a brief system and design overview.