

AI Commander

Summary

- Hassan is the owner of the AI microservice (voclio-ai).
- Responsible for everything related to AI: prompts, OpenAI integration, JSON outputs.
- Has the right to request support tasks from any member of the team (e.g., testing, mock data, documentation, or API stubs).
- No one modifies AI code without his approval

Responsibilities

- **AI Microservice Setup**
 - Repo: voclio-ai (standalone).
 - Stack: Python + FastAPI.
 - Clear repo structure (routes, services, schemas, tests).
- **OpenAI Integration**
 - Wrapper function for OpenAI API calls.
 - Environment variables for API keys.
 - Consistent settings for temperature / tokens.
- **JSON Contracts**
 - Define and lock fixed output formats for every feature.
 - Examples:
 - **Task Breakdown** → {task, steps[], deadline}
 - **Prioritization** → ordered_tasks[]
 - **Daily Summary** → {completed[], pending[], motivation}
- **Core Features**
 - /parse-task: break tasks into steps.
 - /prioritize-tasks: rank tasks by deadlines & importance.
 - /adaptive-followup: reschedule if user falls behind.
 - /daily-summary: end-of-day recap.
 - /calendar-merge: merge Google Calendar events with tasks.
- **Endpoints**
 - Each feature = one POST endpoint.
 - Input = plain text or JSON.
 - Output = standardized JSON.
- **Testing & Documentation**
 - Provide input/output examples.
 - Write README: setup, run, and endpoint usage

Goal

- AI service should be stable, return consistent JSON, and integrate smoothly with backend.
- Backend and frontend should only care about **calling endpoints** — all logic stays inside voclio-ai.