

Yoni Dayagi

San Francisco Bay Area | yoni.dayagi4@gmail.com | [linkedin.com/in/yonid4](https://www.linkedin.com/in/yonid4) | github.com/yonid4 | yonidayagi.me

EDUCATION

University of California, Santa Cruz

Santa Cruz, CA

B.S. in Technology and Information Management

Sep. 2020 - Aug. 2024

Relevant Courses: Data Structures and Algorithms, Database Management Systems, Information Systems

EXPERIENCE

Software Engineer (Contract) – Idea (AI B2B Startup), Remote

Jan. 2025 - May 2025

- Delivered a working prototype of an AI-powered RFP analysis tool for a construction-firm client as the company's sole engineer for 3 months, cutting RFP review from 2–3 days of manual work to under 8 minutes.
- Architected the backend in Python and Flask with Supabase (PostgreSQL), owning infrastructure and tooling decisions from a loosely defined spec (PDF RFP in, structured analysis out).
- Built a Retrieval-Augmented Generation (RAG) pipeline using ChromaDB vector embeddings and cosine-similarity search to work around OpenAI's lack of PDF input, reaching 70–80% accuracy on manually verified outputs.
- Prototyped a Gemini-based pipeline, benchmarked it against the OpenAI version, and convinced management to migrate, raising accuracy to 95%+ and cutting processing time from 15–20 minutes to under 8.

PROJECTS

When. - Group Scheduling App when-now.com *Python, FastAPI, Supabase, Next.js*

Sep. 2025 - Jul. 2026

- Developed a FastAPI backend with a provider abstraction over Google Calendar API and Microsoft Graph (MSAL), syncing busy slots across multiple OAuth-connected accounts per user and writing finalized events to each participant's own calendar.
- Implemented a deterministic scheduling algorithm that generates candidate windows within each day's configured hours (including overnight windows) and ranks them by conflict count and preferred-slot overlap to surface the top 5 meeting times.
- Designed the PostgreSQL schema with Row Level Security isolating each user's events, availability, and calendar tokens, plus a database trigger enforcing a single write calendar per user across providers.
- Pushed live RSVP, availability, and proposal updates to all viewers via Supabase Realtime (WebSockets), and ran hourly background calendar syncs with APScheduler, detecting expired tokens to prompt reconnection.

Job Autopilot — Job Search Automation [GitHub](https://github.com) *Python, Gemini API, GitHub Actions*

Mar. 2026 - Present

- Engineered a pipeline that scrapes LinkedIn and HiringCafe listings, scores each against a parsed resume with Gemini, and writes only jobs scoring 80+/100 to Google Sheets with deduplication.
- Automated a daily GitHub Actions job that reads Gmail, classifies application replies (rejection, assessment, interview, offer), matches each to its application row, updates status, and sends Discord alerts.
- Built a companion tracker that polls Google, Amazon, Netflix, Apple, Microsoft, Meta career portals through their JSON APIs on a GitHub Actions cron and posts new matches to Discord.

MediMind - AI Health Assistant [GitHub](https://github.com) *Python, FastAPI, LangGraph, PostgreSQL, React, TypeScript* Jan. 2026 - Mar. 2026

- Designed a LangGraph agent pipeline that triages each message by severity (emergency, clinical, general, off-topic), injects the user's health profile, vitals, and medications as context, and routes drafts through a critic node that sends unsafe responses back for up to 2 revisions.
- Created an eval harness of 52 labeled cases that gates prompt and model changes on accuracy and emergency recall, validating a move of triage to a lighter Gemini model at 97% accuracy, 7/7 emergency recall, and 0 over-escalations.
- Co-developed the FastAPI backend with a partner, streaming responses over Server-Sent Events while a parallel summarizer node titles new conversations.

SKILLS

- **Languages:** Python, TypeScript, SQL, C/C++
- **Backend:** FastAPI, Flask, REST APIs, OAuth 2.0, JWT, WebSockets, Server-Sent Events, APScheduler
- **Databases:** PostgreSQL, Supabase, MySQL, ChromaDB
- **AI/LLM:** Gemini API, OpenAI API, LangGraph, RAG
- **Frontend:** Next.js, React, TanStack Query, Tailwind CSS
- **Tools:** Git, Docker, GitHub Actions, Google Cloud Platform, pytest