

Jin Ho Yon

(669) 758-1598 | jyon1122@gmail.com | linkedin.com/in/jinhoyon | San Jose, CA

EDUCATION

University of California, Davis

B.S. in Computer Science and Engineering

Graduation: September 2025

Davis, CA

- **Coursework:** Data Structures & Algorithms, Software Engineering, Web Programming, Human-Computer Interaction

EXPERIENCE

Software Developer, Research Assistant

March 2025 – September 2025

Interactive Organisms Lab - Dr. Katia Vega, UC Davis

Davis, CA

- Shipped the front-end of **Nail pHolish**, an Android app surfacing oral pH readings from a Nix Color Sensor; built screens and flows in **Kotlin/XML** via **Android Studio**, iterating with hardware/backend teammates on debugging and usability feedback.
- Prototyped UX flows in **Figma** for **DiabetiCat**, a feline-health monitoring app, partnering cross-functionally on screen architecture, API contracts, and front-end requirements.
- Co-authored **2 peer-reviewed publications** (ACM & INTERACT 2025), writing design and test documentation that translated research findings into shipped product behavior across hardware, backend, and UI teams.

PROJECTS

Darfin, AI-Powered Company Analysis Platform

June 2026 – July 2026

React.js, Spring Boot, Java, Python

- Built the **DART** filing pipeline in *darfin-company-analysis* (**Python**): **DART Open API** collection → **XML** parsing → diff vs. prior filing → **Gemini** summarization/risk analysis, run via **APScheduler** (daily 06:00/18:00 KST) plus an on-demand LLM worker queue.
- Fixed unreliable parsing caused by per-company DART XML variance by **redesigning the pipeline** to pair DART's structured fields with LLM semantic analysis instead of patching the parser; result stably processes all companies, and investor feedback shifted the UI toward filing-**diff** changes over a black-box AI score.
- Built the screens (financial trends, risk, business segments, shareholder status) in *darfin-front* (**React**) against the **Spring Boot** query API (*darfin-main*, isolated from the Python worker via **MySQL**); also built the landing page (SEO meta/structured data) and unified UI styling with **shadcn/radix** on a shared design system (DESIGN.md/DESIGN.SYSTEM.md).

Seenior, AI-Powered Developer Onboarding (IBM Bob Hackathon)

May 2026

React.js, Node.js, Google Gemini API, GitHub REST API, Mermaid.js, D3.js

- Built an end-to-end **LLM-based** full-stack app that reduces codebase ramp-up time from hours to **under 60 seconds** by integrating **Google Gemini 2.5** with the **GitHub REST API** to auto-generate documentation, UML diagrams, and quizzes from any public repository.
- Improved API throughput **80×** (60 → 5,000 req/hr) by implementing token-based authentication through **Octokit**, enabling reliable analysis of large multi-file repos without rate-limit failures.
- Designed a structured prompt + parser pipeline producing consistent docs and architecture breakdowns across **4,096-token contexts**, rendered as interactive, clickable UML diagrams via a custom **Mermaid.js** validation pipeline with **D3.js** and **XYFlow**; took the product from idea to working demo in days through rapid prototyping and iteration in a small cross-functional team.

Fitsbug, Fitness Marketplace Platform

April 2026 – May 2026

Java, Servlets, MyBatis, MySQL, REST APIs, AJAX, Toss Payments

- Designed **relational database schemas** and built an end-to-end PT reservation flow (calendars, booking, session history) backed by **20+ Java servlets**, **10+ MyBatis mappers**, and **RESTful controllers**.
- Integrated **Toss Payment Gateway** for the full payment lifecycle (ready → success → cancel → refund); validated the flow through end-to-end **manual testing and debugging**, verifying database state at each transition and handling cancellation/refund edge cases.
- Versioned the codebase with **Git/GitHub**, using feature branches and pull-request reviews to coordinate changes across a 4-engineer team.

TECHNICAL SKILLS

Languages: Python, Java, JavaScript, SQL, Kotlin, HTML/CSS, XML, C/C++

Frontend: React.js, React Native, TailwindCSS, Three.js, D3.js, Mermaid.js, Android (Kotlin/XML)

Backend & Data: Node.js, Express.js, Java Servlets, Spring Boot, MyBatis, REST APIs, MySQL, Relational Schema Design, APScheduler, STOMP WebSocket

AI / LLM: Google Gemini, OpenAI, Prompt Engineering, Retrieval-Augmented Generation patterns, Model Context Protocol (MCP)

DevOps & Tools: Git & GitHub, GitHub API (Octokit), Render (cloud PaaS), Figma, Claude Code, VS Code, Android Studio; familiar with Agile workflows

Certifications: Introduction to Model Context Protocol (Anthropic)