

University of California, Berkeley

Berkeley, CA

B.S. Electrical Engineering and Computer Sciences, **EECS Honors Program**

Aug 2022 - May 2026

Relevant Coursework: Compilers, Intro to Machine Learning, Computer Security, Efficient Algorithms and Intractable Problems

QUALIFICATIONS

Skills: Fullstack Web Development, API Integration, Data Pipelines, Generative AI, RAG, NLP, Machine Learning, Cloud Deployment

Languages: Python, JavaScript, TypeScript, Go, Java, C, C++, SQL, HTML/CSS, Bash/Shell, GraphQL

Tools: Flask, React, Node.js, Snowflake, LangChain, SpaCy, Django, Docker, Git, Unix/Linux, Jupyter, AWS RDS, PyTorch, TensorFlow

PROFESSIONAL EXPERIENCE

- Software Engineer Intern | Delivr AI — Satellite Beach, FL

May - Aug 2025
- Engineered a Python-based backend service leveraging LLMs (RAG) to classify raw website traffic data into an 87k-topic taxonomy, exposing fit scores and natural-language explanations via REST API for downstream systems.
  - Built scalable storage and analytics pipelines in Snowflake (SQL, Python) to enable low-latency querying, real-time analytics, and seamless integration with audience targeting platforms.
  - Automated end-to-end ingestion and processing workflows using Docker and AWS cloud-native orchestration, eliminating manual intervention, increasing throughput by 20%, and improving production readiness.
- Software Engineer Intern | RecVue — Palo Alto, CA

June - Aug 2024
- Developed an NLP-driven backend application that converts natural language queries into parameterized SQL using SpaCy Named Entity Recognition (NER) and template-based generation, automating customer support workflows.
  - Integrated with AWS RDS (PostgreSQL) to execute dynamic queries and generate human-readable responses, leveraging Flask APIs, LLM prompting, and containerization with Docker for scalable production deployment.
  - Designed a modular, extensible NLP-to-SQL pipeline enabling automated, low-latency data retrieval across multiple customer-facing teams.

- RESEARCH & EXTRACURRICULARS
- Research & Development | ACE Lab, UC Berkeley (Prof. Dan Garcia)

January - May 2025
- Built and deployed AutoRemind, a Python-based backend microservice integrating with LMS platforms via REST APIs to deliver personalized, real-time reminders, improving student engagement and reducing instructor workload.
  - Developed Retrieval-Augmented Generation (RAG) pipelines using LangChain and OpenAI APIs to process course resources and generate targeted, context-aware messages at scale.
  - Designed and optimized retrieval workflows to aggregate and extract data from multiple sources, enabling accurate contextual message generation and seamless integration into production systems.
- Website Developer | Codify Berkeley

January - May 2025
- Developed and maintained Codify Berkeley’s educator-facing website using React and JavaScript, providing structured resources, lesson plans, assignments, and syllabi to support web development courses.
  - Enhanced the club’s online presence by designing and implementing dynamic content sections for general meetings, community events, and member information, improving communication and engagement for over 100 students and educators.
  - Optimized site usability and content management workflows, reducing update turnaround time for new educational materials and logistical announcements significantly.

- FullStack Developer | Web Development at Berkeley

August - Dec 2024
- Contributed to full-stack development of Audi-Friends, a social audiobook discovery platform, building backend architecture with Django and implementing dynamic, responsive frontend rendering.
  - Integrated the Spotify API for real-time audiobook search and personalized recommendations, enabling users to browse titles, view metadata, and discover related content instantly.
  - Implemented user-generated reviews with Django templating and client-side JavaScript, supporting live updates, community interaction, and improved engagement metrics.

- PROJECTS
- Secure File Sharing
- Implemented secure file storage and sharing service in Go using hybrid encryption (RSA + AES), HMAC-based integrity verification, and digital signatures to ensure data confidentiality, integrity, and authenticity.
  - Designed and built robust access control mechanisms, including secure user initialization, invitation-based sharing, and access revocation, engineered to remain resilient against malicious adversaries.