

Justin Lee

justlee888@gmail.com | (415)-965-9148 | www.linkedin.com/in/lee-justin-j | <https://justinlee.dev>

EDUCATION

San Jose State University

Bachelor of Science in Computer Science - 3.73/4.0

San Jose, CA

Dec. 2025

SKILLS

Languages: Javascript, Python, SQL, Java

Front-end: React.js, Next.js, Tailwind CSS, HTML/CSS

Tools: AWS, Docker, GitHub Actions (CI/CD), GCP

Back-end: Express.js, Django, PostgreSQL, MongoDB

PROFESSIONAL EXPERIENCE

Lead Software Engineer

Arctex

Corvallis, OR

Sep. 2024 - Current

- Dockerized and deployed CollegeAppAssist, a MERN app using AWS Fargate and ALB for scalable hosting, resulting in stable performance and reliably serving 25,000+ daily requests with Cloudflare-enabled protection.
- Built a CI/CD pipeline using GitHub Actions to automate deploying to AWS Fargate, eliminating the need for manual intervention and saving 5+ minutes per deployment, accelerating development iteration and reducing friction.
- Implemented session-based authentication with OAuth, enabling secure user login for 300+ monthly users.
- Integrated infrastructure for payment workflows, including Stripe Webhooks to handle real-time transactions.
- Designed a flexible MongoDB schema to efficiently store large amounts of variable HTML ID fields across multiple websites, supporting over 100 top-level attributes and deeply nested structures for consistent and scalable data querying.

Software Engineer Intern

Cadence Design Systems

San Jose, CA

Jul. 2025 - Aug. 2025

- Rebuilt a 5-year-old HTML-to-PDF generator for the Technical Publications team, designing a modern Python solution with BeautifulSoup to combine and standardize 70+ HTML files into a single, professional PDF document.
- Proposed and implemented a scalable architecture by converting individual HTML files into PDFs before merging, enabling processing of 1,000,000+ lines of HTML without crashing headless browser tools like Playwright.
- Engineered a dynamic table of contents system using PyMuPDF to locate and link bounding boxes of headings in the output PDF, ensuring clickable TOC links remained accurate across merged documents.
- Optimized script runtime by over 60% through multiprocessing and efficient memory management, reducing generation time from ~20 minutes to under 8 minutes on large file sets.

Software Engineer Intern

Delta Labs AI

Nashville, TN

May. 2025 - Jul. 2025

- Built the entire frontend using React.js, TypeScript, and TailwindCSS, delivering 7+ responsive pages with modular architecture ensuring clean code organization, component reusability, and long-term scalability.
- Refactored a legacy Flask backend by introducing a DAO layer and implementing connection pooling using psycopg2, significantly improving code modularity and reducing average query latency by over 70%.
- Restructured and designed normalized PostgreSQL schema with 20+ tables and integrated role-based access control.
- Deployed a full-stack app on Google Cloud Compute Engine with Dockerized NGINX, React, and Flask services, using managed infrastructure like Cloud SQL and GCP Secrets Manager for secure, scalable deployment.

PROJECTS

ACM-CS @ SJSU Lenses

Jan. 2024 - May. 2025

Tech Stack: Django, PostgreSQL, Next.js, Docker, OCI

- Scraped and cleaned 25,000+ raw data entries consisting of reviews, professor data, course information, and semester specific schedules to build a reliable dataset for an SJSU Course Scheduler Application.
- Oversaw the development of a relational database using PostgreSQL that holds 12 tables and 30,000+ data entries.
- Developed a secure login system with Google OAuth, implementing token management and automatic refreshing.
- Designed and refactored 20+ RESTful API endpoints for CRUD operations using Django to replace inefficient data handling logic, reducing backend response times and improving maintainability across services.

TheTypingSchool

Jun. 2024 - May. 2025

Tech Stack: Express.js, EJS, OpenAI API, MongoDB, AWS

- Developed a leveling and rewards system for 50+ monthly users, boosting engagement and progress.
- Refactored the Express.js codebase for modularity and performance, reducing endpoint response times by 20%.
- Integrated OpenAI's Batch API with node-cron to auto-generate 100+ monthly typing prompts, replacing manual content creation and enhancing challenge variety for increased user retention.
- Built an eye-tracking feature using WebGazer.js with a calibration screen to block typing when users looked down, reducing gaze-down events by 30%, improving learning outcomes and focus during typing challenges.