Vijit Dua

vijitdua.com/portfolio
(xxx) xxx-xxxx, inbox@vijitdua.com, @vijitdua, @vijitdua, Davis, CA

EDUCATION

University of California: Davis [UC Davis] — Davis, CA
B.S. Computer Science — *September 2023 – June 2027 (expected)*

- **GPA:** 3.616; Dean's Honor List (2x); Tech Lead - #include; Member - Davis Competitive Programming; Novice - Men's Rowing

University of Hong Kong [HKU] — Hong Kong SAR
Semester Exchange — *September 2024 – December 2024*

- **GPA:** 4.0

PROFESSIONAL EXPERIENCE

Happen Bank (formerly LendingClub) — San Francisco, CA
Software Engineering Intern — *June 2026 – August 2026*

- Built a dedicated pre-approval experience for 20k+ monthly loan & deposit applicants, to access in-progress applications (previously web-only) within the company's cross-platform **React Native** Mobile App
- Cut new deep-link rollout from ~2 weeks to instant (no app release) by designing a runtime **Node.js** link resolver with **heap-events** to replace Salesforce Marketing Cloud & per-link **AASA** + app binary updates. (Self-proposed project, after finishing internship scope early)
- Fixed a production **distributed tracing** bug that affected every user of a backend loan-application **microservice**. Requests mixed up Trace IDs under concurrent traffic. Found the cause (**AsyncLocalStorage**), deployed the fix, and verified the outcome in **Splunk**

LendingClub — San Francisco, CA
Software Engineering Intern — *June 2025 - August 2025*

- Deprecated 95% of a legacy consumer portal (200+ files) by merging pre-approval & post-approval customers, who shared several application steps but sat on separate platforms, into one **Django** and **React** (**TypeScript** + **Webpack**) app with two gated root experiences
- Built a **Fastify** orchestrator **microservice** from scratch, servicing ~17k monthly loan/deposit applicants, so pre-approval customers only hit a gated API designed for non-KYC'd users.
- Wrote 40+ **TAP** regression test suites in TypeScript, and verified behavior using demo deployments & **Splunk** monitoring
- Achieved 100% production adoption of the internship project for all new loan/deposit applicants following a phased A/B rollout

UC Davis IET — Davis, CA
Jr. Application Developer SPDP — *February 2025 - Present*

- Rebuilt the official campus mobile app used by 10k+ students by replacing fragmented legacy **iOS/Android** codebases with a unified **React Native** and **Expo** app (**SQLite**, **DrizzleORM**), and authored 25% of the code on a 12-person team
- Prevented a day-one production outage for real-time transit tracking by catching a third-party API rate-limiting edge case during pre-launch testing & fixed it with **AWS Lambda** API proxies
- Built an event monitoring system for AggieCloud, an internal campus cloud credential manager, by shipping full-stack features using **Vue.js 3**, **Vite**, and **Pinia** alongside **AWS Lambda** services deployed via **Serverless Framework** and monitored with **AWS CloudWatch**

KAIST — Daejeon, South Korea
Deep Learning Researcher — *July 2024 - July 2024*

- 1 of 18 students selected globally; developed **U-Net** based satellite image segmentation models using **PyTorch** & **NumPy**
- Trained on ~10k images (SpaceNet/Massachusetts), achieving an F1 score of 0.73 (exceeding ~0.70 benchmarks)

More Experiences (2+): vijitdua.com/experience ; Recommendations: vijitdua.com/recommendations

PROJECTS

VTA | *Next.js, Express.js, PostgreSQL, Redis, Cloudflare Tunnels and R2 (S3), Docker, GitHub Actions, WebSockets, Tanstack Query, Zod....*

- Engineered a full-stack production platform with auth, admin console, bug reports, searchable logs, email queue (direct sends, campaigns, RFC 8058 one-click unsubscribe), & **PWA** – and shipped 5+ apps on it, including services.vijitdua.com
- Built an upgrade tool that merges new template features into forked apps with `git apply --3way`, ensuring forks retain their own changes
- Set up **CI/CD** so `git push origin develop:production` deploys **Docker** containers to a **VPS** behind **Cloudflare Tunnels** in ~4 minutes
- Kept API **stateless**: sessions & cache in **Redis**, files in **R2**. More containers can be added as needed to **horizontally scale**, with no code changes
- Moved scheduled jobs, log processing, media cleanup, and other heavy tasks to a separate worker that talks to the API over Redis so slow jobs don't block the API requests

Aggie Schedule Sniper | *Manifest v3 Chrome Extension*

- Automated time-critical course registration for 500+ students *(500 MAU, 150 weekly)* via **Manifest V3 Chrome extension**. Additionally integrated RateMyProfessor ratings, session keepalives, & Google Calendar syncing; Maintainer of open-source repo contributions.

More Projects (22+): vijitdua.com/projects

LANGUAGES & TOOLS

- **Languages:** TypeScript, JavaScript, Python, C++, C, Java, RISC-V (assembly), x86 (assembly), HTML, CSS
- **Frameworks & Tools:** React.js, Next.js, Vue.js, React Native, Expo, Express.js, Fastify.js, FastAPI, Django, Material UI, Vite, PyTorch
- **Databases:** Redis, PostgreSQL, MySQL, SQLite, MongoDB, Firestore, Sequelize (ORM), Drizzle (ORM)
- **Dev. Ops:** Docker, Git, GitHub Actions, AWS (Lambda,CloudWatch, EC2), Digital Ocean, Firebase, NGINX, Cloudflare
- **AI Tools:** Cursor IDE, Claude Code, Grok Bot