

Sangam Pratap Singh

sangampratapsingh576@gmail.com · +91-9205219678 · random-pikachu.dev · linkedin.com/in/sangam21
github.com/Random-Pikachu · invent.kde.org/randompikachu

EDUCATION

Indian Institute of Information Technology and Management, Gwalior

Aug 2023

B.Tech in Information Technology (with integrated MBA)

- **Coursework:** Data Structures & Algorithms, Object-Oriented Programming, DBMS, Operating Systems, Computer Networks, Machine Learning.

TECHNICAL SKILLS

- **Languages:** C++, C, Go, Python, JavaScript/TypeScript, SQL, Qt/QML
- **Core CS:** Data Structures & Algorithms, Object-Oriented Design, DBMS, Machine Learning, REST APIs
- **Backend & Frameworks:** Node.js, Express.js, Next.js, React.js, WebSockets, PyTorch
- **Databases (SQL & NoSQL):** PostgreSQL, MySQL, Prisma ORM, MongoDB, SpacetimeDB
- **Cloud & DevOps:** AWS (EC2, SES), Azure, Docker, Nginx, Linux, Git, CI/CD, Agile/Scrum

EXPERIENCE

Software Engineer Intern (Full Stack)

Jun 2026 – Present

Retsguide | TypeScript, Node.js, Next.js, React, PostgreSQL, Prisma, Azure

Remote, Copenhagen

- **Owned three customer-facing application flows end-to-end** — design through deployment and ongoing operation; designed the **Prisma/PostgreSQL** schema and ran live-data-preserving enum **migrations** on production data with full status-history tracking.
- Built an **AI-assisted document-classification and eligibility pipeline**, defining the extraction and eligibility-decision logic and validating outputs against real case data to catch and correct misclassifications before they reached the case flow.
- Engineered modular **REST services** on a layered **controller–service–repository (object-oriented) design** with **state-machine** status transitions, validated DTOs, idempotent case creation, and PII-sanitized responses.
- **Root-caused** a server-side-rendering bottleneck causing **2s+** page loads to redundant per-request *getServerSideProps* fetches; restructured the data path and cut server response time to **150ms**, a **13× reduction**.

Software Development Intern

Jul 2025 – Sep 2025

OpenSphere | Node.js, React.js, WebSockets, AWS SES, MongoDB

Remote, San Francisco

- Designed and deployed a **concurrent WebSocket service** powering real-time threaded commenting in the platform's text editor; an **event-driven, non-blocking** model served **simultaneous connections** at low latency, driving **2×** editor adoption post-launch.
- Removed a redundant data-filing stage from the **end-to-end drafting workflow**, delivering **2× throughput** and eliminating manual intervention at that step.

OPEN SOURCE

KDE Plasma Contributor

Dec 2025 – May 2026

C++, Qt/QML | Work featured in KDE's This Week in Plasma (Feb 2026)

Remote

- **Wallpaper Gallery Caching (MR 6280):** Added local caching to the Plasma Wallpaper Gallery with **SHA-256** content hashing for deduplication, keeping entries intact when source files move or are deleted.
- **Wallpaper Path Persistence (MR 6208):** Eliminated repeated directory-browsing friction by persisting picker state via **KConfig**; validated through **functional testing** across distributions and merged after **maintainer code review**; now shipping to **millions of Plasma users**.
- Additional merged MRs: **WCAG keyboard accessibility** in Plasma Digital Clock, **todo state-inheritance bug fix** in Marknote, and **spacebar canvas pan** in Drawy.

PROJECTS

Astra | *Electron, Node.js, Express, MongoDB, Tree-sitter, Gemini AI*

Aug – Sep 2025

<https://github.com/Random-Pikachu/Astra>

- Architected a **GenAI developer-productivity tool**: a **static-analysis pipeline** pairing **Tree-sitter AST parsing** with **Gemini** to surface CVEs via **OSV.dev** and auto-generate remediation patches across **500+ files**, delivered as programmatic **pull requests** — zero manual steps from detection to code review.

CodeRivals | *Go, React.js, SpacetimeDB, AWS EC2, Docker, Nginx, Monaco Editor*

Apr 2026

<https://github.com/Random-Pikachu/code-rivals>

- Developed a **real-time 1v1 competitive coding platform** where two players solve the same problem live; implemented the matchmaking queue, live match state sync, and an **Elo rating** update after each match.
- Self-hosted the open-source **Piston** execution engine on **AWS EC2** behind an **Nginx** reverse proxy, so submitted code runs inside a **sandboxed Docker container** instead of on the app server; wrote a **C++** parser that converts raw test-case input into the argument types each problem expects.

DevTracker | *Go, PostgreSQL, SQL, React.js, Docker, Resend*

Apr 2026

<https://github.com/Random-Pikachu/DevTracker-Backend>

- Created a **scalable Go aggregation service** over **PostgreSQL** unifying the **GitHub, LeetCode, and Codeforces** APIs into one normalized schema with **optimized SQL queries**; added a scheduled digest engine dispatching **HTML email** summaries. Released to real users.