

Rishav Jha

Bengaluru, India | rishav.nitdgp24@gmail.com | 7439876937 | rishavjha.com | linkedin.com/in/rishav-devraj
github.com/rishav-jha-mech

Summary

Full-stack engineer with 3 years of experience shipping **React** and **TypeScript** frontends backed by **Rust** and **Python** services, from design systems and streaming AI interfaces to concurrent execution engines and RAG pipelines.

Education

National Institute of Technology Durgapur, B.Tech. in Mechanical Engineering

Dec 2020 – May 2024

Experience

Tessell, Member of Technical Staff 2 (Software Engineer) – Bengaluru, IN

Aug 2025 – Present

- Won Tessell's company-wide hackathon within two weeks of joining by building **Fetch**, a **Python RAG** knowledge engine over company Slack history and documents that answers natural-language questions with cited source permalinks; it continued as an internal tool after the hackathon
- Architected Fetch's retrieval pipeline end to end: paginated **Slack API** ingestion into normalized **SQLite** storage, batch sync into **ChromaDB** with local **sentence-transformers** embeddings, **LLM query rewriting** and dual-collection retrieval that routes ownership questions to a people index
- Architected and own Tessell's **design system** (~80% of its commits): **50+ production React components** with **Vanilla Extract** Sprinkles and primitive/semantic **design tokens**, shipped as a versioned **npm** package with **Storybook** docs and **Chromatic** visual-regression CI
- Designed **streamUiQuery**, a centralized **Server-Sent Events (SSE)** streaming layer with a typed event contract, and migrated **six AI chat surfaces** onto it, replacing duplicated per-surface stream handling
- Built the **generative-UI rendering engine** that turns AI-streamed node contracts into **React UI**, then merged it into the design system, collapsing two lockstep-released packages into one and removing version-drift bugs
- Engineered a **two-lane concurrent execution engine in Rust** for chat and background AI workloads with bounded concurrency, **MPMC** channel dispatch and hardware-aware parallelism, plus **SQLite**-persisted execution state with automatic crash recovery of orphaned jobs
- Authored **v2-app-storage**, a **Rust Tauri plugin** that moved chats, user metadata and preferences from ChromaDB (5 to 8 second reads) to **SQLite** with a typed per-table query layer, eliminating data persistence failures
- Implemented end-to-end **observability** with request-ID correlation across **TypeScript** and **Rust** into **AWS CloudWatch**, and hardened the **Tauri** security surface with compile-time **IPC** gating and capability restrictions

Cuemath, Software Engineer (Intern to Full-Time) – Bengaluru, IN

Oct 2023 – Aug 2025

- Refactored the 1v1 multiplayer gameplay layer built on **Colyseus**, replacing **setTimeout** workarounds and 10 to 20 events per room state mutation with a single event-driven update, cutting new 1v1 game delivery from 1 to 2 months to **14 days** (~75% faster)
- Implemented player reconnection and AFK handling on **Colyseus** rooms: a 30-second reconnection window, round-level AFK forfeiture and automatic room teardown for bot matches, reducing mid-game drops and orphaned rooms
- Built the **Journey** coachmark and guided-onboarding system with Principal Architects: 8-directional collision detection, absolute and scrollable positioning, gesture-nudge and card formats, and full **Storybook** edge-case coverage, adopted across Circle and Leap
- Built **Colors Clash** in **Phaser.js** across 46 unique map variants, plus a custom **Figma plugin** that extracts vector coordinates from design files to generate the planar adjacency graphs the game logic needs
- Migrated the sunset Circle app's games and streaks into the **Leap** web platform and built a real-time teacher-student demo matching flow for sales trials
- Shipped multi-persona onboarding for Students and Parents across **Leap (React)** and the **React Native** apps, with conditional step-routing state machines

Google Summer of Code (GSoC), Contributor and Mentor – gso.c.rishavjha.com

May 2023 – 2025

- Redesigned the Talawa Admin UI for the Palisadoes Foundation in **React** and **TypeScript** from a new Figma design as a responsive, modular interface, deployed incrementally screen by screen as reusable, documented components
- Refactored the Talawa website codebase and added **test coverage**, resolving **cross-browser compatibility** issues and

improving long-term maintainability

- Returned as a **GSoC mentor** for the Palisadoes Foundation in the **2024 and 2025** cohorts, guiding contributors

Capricon Technology, React Native and Flutter Developer

Jun 2022 – May 2024

- Built **Treklers**, a **React Native** and **TypeScript** taxi booking app with **50+ screens**: real-time driver tracking, scheduled rides, emergency calling and cash, wallet and card payments; separating UI from business logic let four full redesigns ship without regressions
- Built **Humanist Earth** (formerly Illuminate Investing) screens (live on the App Store and Google Play): dashboard, portfolio allocation, risk questionnaire and shareholder voting; built **Training Tracker**, a workout logger with **React Native Skia** charts

Projects

Almond AI

almondai.tech

- Built **Almond AI** with a small team: a shared, multiplayer **memory layer for AI tools** where a team records decisions once and every connected tool reads the same searchable memory; in private beta
- Built native integrations for **Claude Code**, **Cursor**, **Figma** and **Figma Make**, plus any **Model Context Protocol (MCP)** compatible tool, so AI assistants read and write shared context instead of resetting it
- Worked across the stack: a **Python** backend with **Neo4j**, a **React** and **TypeScript** frontend, and **Docker** packaging for self-hosted, on-premises deployment

VertiView

[Chrome Web Store](#)

- Built a **Chrome extension** that turns any video on a page into a zoomable, pannable 9:16 portrait panel so several videos can run side by side; wrote the zoom, pan-bounds and crop layout engine with **unit tests**, plus a **Next.js** marketing site
- **Tools**: TypeScript, Chrome Extension APIs, Vite, Next.js

Placement Management System

NIT Durgapur

- Built the campus placement portal for NIT Durgapur in **Django** on **AWS EC2**, tracking 180+ companies from registration to offer letters with admin and coordinator accounts, task tracking, push notifications and CSV export
- **Tools**: Django, Python, JavaScript, Bootstrap, AWS EC2

Buzzpost

[Live Site](#)

- Built a blogging platform with a social feel in **Django**: rich-text posts, likes, comments and replies, search and a trending page, plus a **React Native** companion app
- **Tools**: Django, Python, JavaScript, React Native, TinyMCE

Squid Game

[GitHub](#)

- Built a Squid Game themed event site for 300+ participants: a **Three.js** Green Light Red Light round using **Pico.js** face detection to catch players who moved, served from **AWS EC2** with **Gunicorn**, **Nginx** and **Supervisor** over SSL
- **Tools**: Django, Three.js, Pico.js, AWS EC2, Nginx, Gunicorn

Skills

Languages: TypeScript, JavaScript, Python, Rust, SQL, HTML5, CSS3

Frontend: React, Next.js, Redux, Zustand, Tailwind CSS, Vanilla Extract, Bootstrap, Storybook, React Native, Phaser.js, Three.js

Backend: Django, FastAPI, Flask, Node.js, Express.js, Tauri, REST APIs, GraphQL, Server-Sent Events (SSE), WebSockets, Colyseus

Data & Cloud: PostgreSQL, SQLite, Neo4j, SurrealDB, Redis, DynamoDB, ChromaDB, Amazon Web Services (AWS): EC2, S3, CloudWatch, Cognito; Docker, Nginx, GitHub Actions, CI/CD, Git

Concepts: Full-Stack Architecture, Design Systems, Concurrency, Observability, RAG, LLM Integration, Real-Time Systems, System Design