

Sandeep V

dev.sandeep@gmail.com | +91-9380730286 | github.com/sand1027 | linkedin.com/in/sandeepv
leetcode.com/u/sandy_1027 | sandeepdev.space | Bangalore, Karnataka

SUMMARY

Full-Stack Engineer with 1.5 years of experience, sole owner of 5+ production products end-to-end. Deep experience across React Native, Node.js/GraphQL, MongoDB aggregation pipelines, and Next.js, architected and implemented payment, notification, and automation systems from schema through resolvers, and shipped mobile releases through App Store and Play Store.

PROFESSIONAL EXPERIENCE

Software Development Engineer - 1
Stance Health

Aug 2025 – Present
Bangalore, Karnataka

Backend API Node.js, GraphQL, MongoDB, TypeScript

- Designed GraphQL schema, MongoDB models, resolvers, and service layer with tsyringe DI from first commit to production.
- Rebuilt slot availability engine via two strategies: predefined org-level slot pre-generation and aggregation pipeline rewrites, cutting response from 1 min to 2 sec and payload by 95%.
- Built Razorpay integration end-to-end: order creation, webhook verification, refund processing, and dynamic PDF invoice/receipt generation.
- Designed RBAC with global/center/org-scoped permissions and GraphQL middleware enforcement; built Policy Consent System with OTP-gated verification and versioned policy compliance recording.
- Built clinical Rule Engine (VALD asymmetry thresholds, patient alerts) and Credit System (REFERRAL/REFUND types, multi-credit invoice application).
- Built an event-driven Automation Rule engine (condition/action pairs on events like PAYMENT_COMPLETED) that auto-issues referral credits and drives patient status transitions without manual staff intervention.
- Built the WhatsApp Business API architecture (Interakt) powering automated invoice receipts, advance receipts, and appointment reminders across every center.
- Deployed a FastAPI attendance-tracking sidecar reverse-proxied behind the main Node API, isolating a CPU-heavy scheduling service without disrupting the existing GraphQL surface.

Dashboard Next.js, Apollo Client, WebSocket, TypeScript

- Built Billing UI (advance/invoice/credits), RBAC-protected rendering, VALD Metrics Dashboard with normative benchmarks, and real-time sync via WebSocket hook.
- Built Alerts/Rule Engine UI, Booking Link Generator, WhatsApp share with dynamic templating, and patient profile management with referral type, cohort, and Goals UI.

Mobile App: stance.app React Native, Expo/EAS, FCM

- Shipped iOS/Android app with full EAS pipeline across 5+ production versions: signing, build profiles, and device provisioning.
- Built FCM end-to-end: token registration, foreground/background/quit handling, and session-level deduplication; OTA delivery via expo-updates with JS hotfixes reaching users in minutes.
- Built appointment timeline screen with search, multi-filter (cancelled/upcoming/past), reschedule/cancel flows, and Firebase Analytics event tracking on all interactions.
- Fixed critical iOS crashes: Firebase init race in AppDelegate.swift, Analytics v24 migration, and UIKit modal stacking crash.

Booking App: stance-booking Next.js, Razorpay, Sole Architect

- Built 7 booking flows (in-person/online/prepaid x new/repeat + offline-new) with org-slug multi-tenant routing, Razorpay prepaid, tab-isolated sessions, UTM attribution, and deep-link system.
- Built Meta Pixel and GA4/GTM marketing funnel tracking (Lead, CompleteRegistration, InitiateCheckout, AddPaymentInfo, Purchase) across every booking flow for ad-spend attribution.

Production Operations and DevOps Vercel, EAS, MongoDB Atlas, GitHub Actions

- Sole engineer deploying and maintaining all 5 services; shipped 801+ fix commits across payment, slot, invoice, app crash, and push notification systems.

- Managed EAS build channels, OTA rollouts with runtime version pinning, and MongoDB Atlas Triggers for appointment auto-cancellation (3-min/30-min retry) and notification dispatch.

PROJECTS

Archly – Browser-Based System Design Tool

github.com/sand1027/Archly

Go, Next.js 14, PostgreSQL, Redis, WebSocket

- Excalidraw-based canvas with an 80+ component library (compute/storage/messaging/network/AI), AI text-to-diagram generation (Gemini primary, OpenRouter fallback), and real-time multi-user collaboration over a Go WebSocket hub.
- Built a traffic simulation engine (RPS/latency/error-rate metrics, packet animation) with 7 chaos-injection scenarios to stress-test drawn architectures before they're built.
- Shipped diagram export to Terraform, Docker Compose, and Kubernetes manifests, plus a standalone Go CLI (Homebrew tap + `go install`) for headless generate/export/session workflows against the same API.
- Designed the full-stack architecture end-to-end: Next.js 14 frontend, Go+Chi REST/WebSocket API, PostgreSQL with goose migrations, Redis caching, and Kafka for async event processing across 9 Dockerized services.
- Built JWT auth, rate-limiting middleware, and a community gallery with share links for saved architecture diagrams.

ApiLens – API Discovery, Replay & Testing CLI

github.com/sand1027/apilens

Go, Cobra, GraphQL, OpenAPI

- Automatic Api discovery tool: parses OpenAPI/Swagger and scans Express/Fastify/NestJS/Gin/Fiber/Echo source to populate an API registry, then a loopback proxy (`apilens watch`) captures real traffic and turns any call into a runnable YAML test.
- Shipped a full test lifecycle: chained DSL v2 response references, JSON Schema/OpenAPI contract testing, bounded load testing with p50/p95/p99, opt-in MongoDB/SQL database assertions, and JUnit output for CI.
- Added first-class GraphQL support (SDL discovery, query/mutation assertions distinguishing HTTP 200 from GraphQL `errors[]`) validated end-to-end against a production GraphQL API.
- Shipped a local web dashboard (`apilens ui`) reusing the same Go engine as the CLI, API dependency graphs and mock serving from captured traffic, and a `coverage` command cross-referencing the discovered registry against written tests.

TECHNICAL SKILLS

- **Languages:** TypeScript, JavaScript, Go, Python, C, C++
- **Architecture & Engineering:** Payment Systems, Notification Systems, Multi-Tenant Architecture, Webhook Integrations, Background Jobs, Production Debugging, Performance Optimization
- **Frontend:** React, Next.js 14, Tailwind CSS, Framer Motion, Apollo Client, Radix UI, shadcn/ui, React Flow, Recharts
- **Mobile:** React Native, Expo, EAS Build/Submit, OTA Updates, FCM, Firebase
- **Backend:** Node.js, Express, GraphQL (Apollo Server), REST API, MongoDB Aggregation, Razorpay, PDF Generation, tsyringe DI, System Design, Event-Driven Architecture, Redis, Database Design, API Design, Dependency Injection
- **Databases:** MongoDB Atlas, Atlas Triggers, Mongoose, PostgreSQL, MySQL, PL/SQL, Firestore, Supabase, SQLite
- **DevOps & Tools:** AWS (EC2, S3, Lambda, Event Brodge), GitHub Actions, CI/CD, Docker, Vercel, EAS Channels, Git, Postman, n8n, Figma
- **Auth & APIs:** RBAC, OTP, JWT, Clerk, NextAuth, Google OAuth, WhatsApp Business API, WebRTC, WebSocket

EDUCATION

Master of Computer Applications (MCA)

2023–2025

PES University, Bangalore, Karnataka

Bachelor of Computer Applications (BCA)

2020–2023

BGS First Grade College, Mysore, Karnataka

ACHIEVEMENTS

Bias for Action Award

2025

Stance Health – Recognized for rapid incident ownership and shipping critical fixes under pressure across 5 live services. [View Certificate]