

Tanay Desai

(667) 289-3387 tanaydesai2004@gmail.com linkedin.com/in/tanaydesai1510/ github.com/TanayDesai-1510 [Portfolio](#)

Education

Rutgers University–New Brunswick

Sep 2023 – May 2027

Bachelor of Science in Computer Science & Minor in Data Science

Shri Bhagubhai Mafatlal Polytechnic

Dec 2020 – Jun 2023

Diploma in Information Technology

Experience

Rutgers Office of Information Technology - Software Engineer

Feb 2024 – present

React.js, Redux, TypeScript, JavaScript

New Brunswick, NJ

- Modernized Rutgers' central web portal serving over **70,000** students, faculty, and staff daily, building and migrating features onto a React.js, Redux, and TypeScript architecture alongside a cross-functional team.
- Designed **8+** reusable, responsive React.js components adopted across multiple internal applications, standardizing shared UI patterns to cut duplicated markup and accelerate feature delivery.
- Improved accessibility of shared portal components for students using assistive technology, aligning markup and interaction patterns with accessibility standards.
- Resolved recurring UI and state-management defects by refactoring Redux data flows, improving reliability of shared components across student-facing services.

Vertex Inc. - Software Engineering Intern (Backend)

Jun 2026 – Aug 2026

Java, Spring Boot, Gradle, Docker

King Of Prussia, PA

- Remediated **18** security vulnerabilities across distributed Java/Spring Boot microservices, including a critical improper-authentication flaw (Snyk priority 655/1000) exposing remote-code-execution paths, by upgrading dependencies and validating fixes across a multi-repository Gradle build.
- Patched a critical soft-delete bypass on a certificate platform by enforcing null-safe deleted-record filtering across **MyBatis** queries and tests, stopping deleted records from leaking into downstream view, approval, edit, and sync flows.
- Diagnosed a production data-integrity bug where a write was missing across related records, enforcing a **synchronized** two-write update path that eliminated downstream wizard-edit failures.
- **Migrated AWS S3** buckets from deprecated inline Terraform attributes to standalone lifecycle, versioning, and CORS resources, preventing unintended object expiration under managed lifecycle policies.

Projects

QuotaForge | *Java 21, Spring Boot, Redis, PostgreSQL, Docker, OpenTelemetry, Grafana, k6*

- Built a multi-tenant distributed rate-limiting service in Java 21 / Spring Boot, enforcing per-tenant request quotas with atomic Redis Lua scripts to guarantee correct limiting under concurrent load without race conditions.
- Handled requests on virtual threads with an async, idempotent metering pipeline, sustaining correct usage accounting under high concurrency without double-counting.
- Instrumented the service with OpenTelemetry distributed tracing and Grafana dashboards, surfacing per-tenant latency and quota-rejection metrics end-to-end.

SignalWatch | *TypeScript, Next.js, React, PostgreSQL, Prisma, NextAuth, Tailwind*

- Built an **AI-powered** stock monitoring pipeline that ingests real-time BSE/NSE corporate announcements and uses Claude (Anthropic SDK) to auto-summarize, categorize into 10 types, score importance (1–10), and tag sentiment; then fans out preference-based email alerts and live updates via Server-Sent Events.
- Shipped a full-stack Next.js 16 / React 19 / TypeScript app on Vercel with NextAuth authentication, a normalized PostgreSQL/Prisma schema (11 models spanning watchlists, notification preferences, and monitoring logs), CSV portfolio import, and a background job system with per-source delivery logging.

RouteMesh | *Java 21, Spring Boot, Spring Cloud Gateway, Kafka, PostgreSQL, Docker, OpenTelemetry, Prometheus, Grafana*

- Built a **distributed** API management platform with a versioned PostgreSQL-backed control plane, dynamic service discovery, lease-based registration/heartbeats, and health-aware round-robin routing across service instances without gateway restarts.
- Engineered **fault-tolerant** request routing with circuit breakers, bounded retries, timeouts, last-known-good configuration caches, and automatic instance eviction/re-registration, preserving availability during downstream and control-plane failures.
- Designed an **event-driven** observability pipeline using Kafka, idempotent telemetry persistence, retry/DLT handling, OpenTelemetry distributed tracing, and Prometheus/Grafana metrics to correlate HTTP requests, asynchronous usage events, latency, failures, and per-instance traffic end-to-end.

Technical Skills

Programming Languages: Java, Python, C/C++, JavaScript, TypeScript, SQL, HTML/CSS

Libraries and Frameworks: Spring Boot, React.js, Node.js, Next.js, Vue.js, Express.js, Flask

Databases: PostgreSQL, MySQL, Oracle, MongoDB, Redis, SQLite

Tools & Platforms: Git, GitHub, Docker, Gradle, Linux/Unix, OpenTelemetry, Grafana, k6, Postman