

# Daniel Nguyen

## Senior Full-Stack & AI Engineer

Houston, TX | (251) 309-3500 | [hello@danielhnguyen.net](mailto:hello@danielhnguyen.net) | [LinkedIn](#)

### PROFESSIONAL SUMMARY

---

Senior Full-Stack & AI Engineer with over 10 years of experience architecting high-scale SaaS platforms and autonomous agent systems. Expert in bridging traditional distributed systems with modern generative AI, building robust infrastructure for real-time Voice AI, conversational agent platforms, and financial automation. Proven track record of optimizing low-latency data pipelines, designing resilient event-driven architectures, and shipping production-grade applications that integrate with legacy enterprise systems. Specialized in system reliability, cloud infrastructure, developer tooling, and delivering autonomous business logic that drives operational efficiency at scale.

### TECHNICAL SKILLS

---

**Languages:** Python, JavaScript, TypeScript, Java, SQL, Bash, HTML, CSS

**Frontend:** React, Next.js, Angular, Redux Toolkit, React Query, Tailwind CSS, SPA/SSR, Virtualized Rendering, Performance Optimization

**Backend & APIs:** FastAPI, Django, Node.js, Express.js, Nest.js, Spring Boot, REST, OpenAPI/Swagger, GraphQL, gRPC, WebSockets, Server-Sent Events (SSE), Celery, OAuth2/JWT, Middleware Pipelines

**AI, LLM & Voice:** OpenAI API, Anthropic API, Function Calling/Tooling, ReAct Agent Loops, LangChain, RAG, Embeddings, Vector Search, Pinecone, pgvector, Semantic Caching, Prompt Engineering, Evals Pipelines, Sentiment Analysis, Whisper (STT), Twilio Voice/Media Streams, NLP, Multi-Model Orchestration

**Data & Messaging:** PostgreSQL, Redis, Apache Kafka, MongoDB, SQL Tuning, Indexing/Partitioning, Background Jobs, Idempotency Patterns, Distributed Locking, Event-Driven Architecture, Pub/Sub

**Cloud & DevOps:** AWS (ECS, Lambda, RDS, ElastiCache, S3, CloudWatch, IAM, VPC), Docker, Kubernetes, Terraform (IaC), CI/CD (Jenkins, GitHub Actions), Linux, Git, Ephemeral Environments, Canary Rollouts

**Payments & Integrations:** Stripe Connect, Split-Payment Orchestration, PCI Compliance, PII Redaction, FMS Integrations (StorEdge, SiteLink, Storage Commander)

**Testing & Reliability:** Pytest, JUnit, Mockito, Jasmine, Karma, Integration/Regression/E2E Testing, Load/Latency Testing, Feature Flags, Incident Response, Monitoring/Alerting, Distributed Tracing, SOC 2 Compliance

### PROFESSIONAL EXPERIENCE

---

**Swivl** | *Senior Full-Stack Engineer*

Jan 2025 – Present

*Led development of a real-time **Voice AI Agent** and full supporting infrastructure for a self-storage conversational AI platform. Owned the voice pipeline end-to-end—from **Twilio** telephony ingestion to **LLM** response—built bi-directional **FMS integrations**, and stood up the **AWS** cloud infrastructure and **CI/CD** workflows serving 4,500+ facilities.*

- Architected a real-time **Voice AI Agent** for inbound tenant support using **Python**, **FastAPI**, and **WebSockets**, ingesting **Twilio Media Streams (base64 mulaw)** with **Whisper (STT)** for speech recognition, enabling seamless **barge-in** with **sub-500ms** end-to-end latency and reducing live-agent routing by **40% QoQ**.
- Implemented **Redis Semantic Caching** for high-frequency tenant intents, bypassing **OpenAI API** inference for **35%** of inbound calls and cutting **p95 latency by 600ms**.
- Designed **NLP** intent classification and routing logic using **LLM function calling/tooling** that maps caller utterances to autonomous actions, escalation paths, or knowledge-base lookups based on confidence thresholds and tenant context.
- Built an **Embeddings**-powered knowledge retrieval system using **pgvector** and **RAG** (Retrieval-Augmented Generation) to ground agent responses in facility-specific FAQs, policies, and operational data.
- Implemented a **PII Redaction middleware** pipeline and scheduled asynchronous post-call jobs via **Celery/Redis** broker for transcript generation, call scoring, and CRM updates—ensuring **PCI compliance** without blocking the real-time voice pipeline.
- Engineered a bi-directional **FMS sync engine** integrating **StorEdge** and **SiteLink REST APIs** with **distributed locking** and conflict resolution, ensuring **idempotent** writes across concurrent updates for **50,000+ storage units**.
- Integrated real-time **inventory lookup** and **access code retrieval** from FMS systems and developed a secure **Action Layer** with **Stripe Connect** for autonomous **split-payment** processing via **OAuth2/JWT**-secured API calls, documented with **OpenAPI/Swagger** specs.

- Built omni-channel workflows dispatching **SMS/email** follow-ups via **Pub/Sub** event triggers and designed seamless escalation from **Voice AI** to live agents with full conversation context summaries, reducing agent ramp-up time on escalated calls.
- Built a **Next.js** Live Monitor and property-manager dashboard (**React**, **Tailwind CSS**, **React Query**) using **Server-Sent Events (SSE)** for real-time call transcripts, with **LLM-powered sentiment scoring** to flag negative tonal shifts and configurable toggles for AI behavior controls.
- Configured **CloudWatch** dashboards, custom metrics, and alerting for voice pipeline health (**WebSocket** latency, **LLM** response time, FMS sync failures), enabling proactive **incident response** with **distributed tracing**.
- Engineered a **telephony emulation suite** to mock **Twilio Media Streams** and **FMS state**, with **Pytest**-based integration and **load/latency tests** validating sub-500ms p95 under simulated peak traffic for full **end-to-end regression testing**.
- Provisioned **AWS** infrastructure (**ECS**, **RDS**, **ElastiCache**, **S3**, **CloudWatch**, **IAM**, **VPC**) using **Terraform** IaC, enabling reproducible deployments across staging and production environments.
- Built **CI/CD pipelines** with **GitHub Actions** for automated linting, testing, and **Docker** image builds with zero-downtime deployments to **AWS ECS**; orchestrated local development with **Docker Compose** to reduce environment drift.
- Established **feature flags** and **canary rollout** strategies for safe incremental releases; managed **Linux** deployment targets and wrote **Bash** automation scripts for log rotation, health checks, and database migrations.

**Tech Stack:** Python, FastAPI, WebSockets, Next.js, React, TypeScript, Tailwind CSS, React Query, Redis, PostgreSQL, pgvector, Celery, OpenAI API, Twilio Voice/Media Streams, Whisper (STT), Stripe Connect, SSE, OAuth2/JWT, OpenAPI/Swagger, Docker, Docker Compose, AWS (ECS, RDS, ElastiCache, S3, CloudWatch, IAM, VPC), Terraform, GitHub Actions, Pytest, Feature Flags, Bash, Linux

**Sierra** | *Senior Software Engineer*

Jan 2023 – Dec 2024

*Built core agent infrastructure for a conversational AI platform serving enterprise brands. Owned the **ReAct agent runtime**, **vector search** retrieval layer, **multi-model orchestration**, and **developer tooling** powering multi-step customer workflows across chat, voice, and messaging channels.*

- Built the core **ReAct** (Reason + Act) **event loop** in **Python** with transactional state management, enabling agents to reliably execute multi-step workflows with automatic **retries**, **rollback**, and **idempotent** action execution.
- Designed multi-tenant **PostgreSQL** + **pgvector** schemas for enterprise knowledge retrieval, tuning **partial indexes** and **HNSW vector search** to query millions of rows at **single-digit ms p95 latency**.
- Built a **RAG** pipeline with custom chunking, **embeddings** (stored in **pgvector**), and **hybrid search** (keyword + **vector search**) to ground agent responses in brand-specific knowledge bases and policy documents.
- Developed a **Tool Registry** that converts **Python** type hints into **OpenAI API**-compatible **function calling** schemas at runtime, enforcing strict input/output validation for all backend tool calls.
- Implemented **multi-model orchestration** logic enabling agents to dynamically route between **OpenAI**, **Anthropic API**, and open-source LLMs based on task complexity, latency requirements, and cost constraints.
- Built a **React** Trace Visualizer with **TypeScript** to render step-level agent execution graphs (tool calls, retrieval context, reasoning chains, latency) for faster debugging and **prompt engineering** iteration.
- Engineered a deterministic action execution layer with **gRPC** service communication, ensuring all system-of-record mutations (CRM updates, order modifications) are **auditable** and compliant with enterprise security policies.
- Created an automated **Evals pipeline** using **Pytest** to run regression tests against a golden dataset of conversation scenarios, catching **prompt** regressions and **hallucinations** before every deployment.
- Integrated **PII redaction middleware** to scrub sensitive customer data (SSNs, credit cards) from prompts before sending to external **LLM providers**, ensuring data governance compliance.
- Provisioned **AWS** infrastructure (**ECS**, **RDS**, **Lambda**, **S3**, **VPC**) using **Terraform** IaC and **Kubernetes** orchestration, implementing **ephemeral preview environments** reducing staging spin-up from days to minutes.
- Implemented **feature flags** and **canary rollout** strategies with **monitoring/alerting** to safely deploy experimental agent behaviors to specific enterprise cohorts without production downtime.
- Built conversation **analytics and observability** tooling with **distributed tracing** to track agent metrics (resolution rate, CSAT, hallucination rate, latency) with real-time dashboards.
- Developed **prompt engineering** workflows with few-shot examples, in-context learning, and self-consistency techniques across brand-specific agent configurations, systematically improving response accuracy and reducing hallucination rates in production deployments.

**Tech Stack:** Python, React, TypeScript, PostgreSQL, pgvector, OpenAI API, Anthropic API, RAG, Embeddings, gRPC, AWS (ECS, RDS, Lambda, S3, CloudWatch, VPC), Terraform, Docker, Kubernetes, Redis, Pytest, GitHub Actions, Feature Flags

## HighRadius | *Software Engineer II*

Jan 2018 – Dec 2022

*Worked across the full stack of a fintech **SaaS** platform automating Accounts Receivable and Treasury processes. Decomposed monolithic systems into **event-driven microservices**, built high-throughput **payment data pipelines**, and delivered front-end tools for AR analysts.*

- Decomposed a monolithic payment processor into **event-driven microservices** using **Apache Kafka** with **Pub/Sub** topic partitioning, enabling asynchronous high-volume transaction processing with **guaranteed delivery** and fault isolation.
- Developed a high-concurrency **Java Spring Boot** service to parse and normalize unstructured payment data from **PDFs**, **EDI** files, and bank remittances at a throughput of **10,000+ documents/hour**, using **background jobs** for async processing.
- Engineered a custom **React virtualized** grid with **Redux Toolkit** to render **100,000+ rows** of financial data with zero **performance** degradation, replacing a legacy **JSP** interface and improving analyst productivity.
- Optimized multi-terabyte **SQL** queries by implementing **table partitioning**, **covering indexes**, and query plan analysis across the **PostgreSQL** data warehouse, reducing report generation from minutes to **sub-seconds**.
- Implemented **distributed locking** via **Redis** to ensure data integrity and prevent race conditions when multiple analysts modify the same invoice record simultaneously, following **idempotency patterns**.
- Collaborated with Data Science to deploy a **Propensity-to-Pay** predictive model, building the low-latency **REST** inference API with **OpenAPI/Swagger** docs that served real-time credit risk scores to the collections workflow.
- Built automated cash application matching algorithms that reconcile incoming payments against open invoices using **fuzzy matching** on remittance data, reducing manual intervention.
- Engineered **ETL pipelines** to ingest and normalize payment data from 50+ banking formats (**BAI2**, **MT940**, **CAMT.053**), standardizing heterogeneous financial data into a unified **SQL** schema.
- Built **SPA** dashboard components in **React** for AR analysts to monitor Days Sales Outstanding (DSO), aging buckets, and collection effectiveness using **GraphQL** queries for flexible data fetching.
- Led the migration to a global **Redux** store with normalized entity state, improving data consistency and reducing redundant API calls across the financial application suite.
- Wrote comprehensive integration tests using **JUnit** and **Mockito**, increasing code coverage to **90%+** across critical payment processing paths and ensuring **SOC 2** compliance.
- Implemented **role-based access control (OAuth2/JWT)** and **audit logging** across the platform to meet enterprise security requirements and support **SOC 2 Type II** certification.
- Designed **event-sourced** audit trails for all payment mutations, enabling full transaction replay and reconciliation reporting required for **SOC 2** Type II compliance and annual financial audits across 600+ enterprise clients.

**Tech Stack:** Java, Spring Boot, Apache Kafka, React, Redux Toolkit, JavaScript, TypeScript, PostgreSQL, Redis, MongoDB, GraphQL, REST APIs, OpenAPI/Swagger, Python, JUnit, Mockito, Jenkins, AWS, Docker

## BMC Software | *Software Engineer*

Jun 2015 – Dec 2017

*Developed front-end frameworks and backend microservices for BMC's enterprise **ITSM** platform (Remedy/Helix). Migrated legacy services to **RESTful** architectures and established the team's first automated testing and **CI/CD** practices.*

- Developed a custom **Angular** directive framework to dynamically render complex **ITSM** forms based on **JSON** schemas, enabling no-code workflow building for IT administrators and reducing form development time by **70%**.
- Migrated legacy **SOAP/XML** endpoints to **RESTful Node.js (Express.js)** microservices, implementing **circuit breakers** (Hystrix patterns) to prevent cascading failures and improve system resilience under load.
- Built a real-time **log aggregation pipeline** using **MongoDB** for server log ingestion, enabling **monitoring** of automation success rates, incident resolution times, and service health metrics.
- Automated build and deployment pipelines using **Jenkins CI/CD**, reducing manual release times by **30%** and enabling one-click staging deployments with rollback capabilities on **Linux** servers.
- Built a reusable **Angular** component library (with **HTML/CSS** and **JavaScript**) for ITSM form elements (approval chains, **SLA** indicators, escalation triggers), standardizing UI patterns across the Remedy suite.
- Developed **RESTful API** endpoints in **Node.js (Express.js)** for ITSM workflow automation, enabling third-party integrations with monitoring tools, ticketing systems, and **CMDB** data sources.

- Developed server-side **middleware** for request validation, authentication, and rate limiting across the **Node.js** microservices layer, improving API security posture and **performance optimization**.
- Implemented **ITIL-aligned change management** workflows with automated approval routing, conflict detection, and release scheduling, reducing change-related incidents across managed infrastructure.
- Mentored junior developers on asynchronous **JavaScript** patterns (Promises, async/await, event loops), conducting code reviews that improved overall team velocity by **15%**.
- Implemented unit testing frameworks using **Jasmine** and **Karma**, establishing the team's first **continuous testing** protocol for the frontend codebase with **Git**-hook-based pre-commit validation.
- Developed **RESTful** webhook integrations between Remedy and third-party **monitoring** tools (Nagios, SNMP traps), enabling automated incident creation and escalation from infrastructure alerts with deduplication logic.
- Created interactive **Angular** dashboards with real-time **SLA** compliance tracking, surfacing MTTR, first-call resolution, and backlog metrics from **MongoDB** aggregation queries for operations leadership.

**Tech Stack:** JavaScript, TypeScript, Angular, Node.js, Express.js, MongoDB, REST APIs, SOAP/XML, HTML/CSS, Jenkins, Jasmine, Karma, Hystrix, JSON Schema, Linux, Git, Bash

## EDUCATION

---

**University of Houston** | Houston, TX

Bachelor of Science (B.S.), Computer Science | 2012 – 2015