

Runze Zhang

(774) 641-7223 | rz348@georgetown.edu | linkedin.com/in/runze-zhang82 | github.com/RunzeZ0710

EDUCATION

Georgetown University, McDonough School of Business

Expected May 2028

Bachelor of Science in Business Administration; Minor in Computer Science, Major in Finance

Washington, D.C.

- GPA: 3.90/4.00; AIME Qualifier; USA Computing Olympiad (USACO) Silver Division
- CS Coursework: Object-Oriented Programming, Computer Science I–II, Math Methods for CS, Linear Algebra, Python

TECHNICAL SKILLS

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

Frameworks: React, Next.js, Node.js, FastAPI, Django

Cloud & Data: PostgreSQL, Amazon Web Services (AWS), Google Cloud Platform (GCP), Docker, Vercel

APIs & Auth: RESTful APIs, OAuth 2.0, Gmail API, Stripe API

Developer Tools: Git, GitHub, GitHub Actions (CI/CD)

AI: NVIDIA NIM & Riva ASR, OpenAI API, TypeSafe (Jev), RAG, embeddings, vector search

EXPERIENCE

Kine

Apr. 2026 – Present

Founder & Full-Stack Software Engineer

Washington, D.C.

- Engineered fault-tolerant payment processing for 15,000+ annual transactions with idempotent webhooks, retry-safe state transitions, event logging, and reconciliation checks to prevent duplicate charges and recover from failures.
- Designed and executed 800+ automated tests running in CI on every push across payment, refund, webhook, transfer, and reconciliation workflows, reproducing asynchronous failure scenarios and resolving defects before production deployment.
- Built and deployed a multi-tenant payments platform (Next.js, TypeScript, PostgreSQL, Stripe Connect, AWS) serving five businesses and processing \$1.2M+ in annualized volume.
- Designed PostgreSQL schemas and Row Level Security policies to isolate customer, payment, invoice, and membership data across five organizations.

Kinstead Health

Jan. 2026 – May 2026

Software Engineering Intern

Washington, D.C.

- Built an automated search engine optimization pipeline with Django and the OpenAI API; measured an 18% lower cost per acquired client versus paid advertising in a controlled customer-acquisition experiment.
- Built a Django credentialing pipeline that parsed provider documents, organized application data, and generated submission emails for 19 nurse practitioners, replacing repetitive manual data-entry steps.

PROJECTS

Inboxd | Next.js, TypeScript, PostgreSQL, Google Cloud, Gmail API, OAuth 2.0, LLMs

2026 – Present

- Building an AI networking inbox for students recruiting in finance that syncs Gmail, classifies networking threads with TypeSafe's Jev model, and drafts personalized cold emails and replies that students approve before sending.
- Engineered an incremental Gmail sync into a PostgreSQL header mirror within per-user API quotas, reading bodies live, with Row Level Security and pgTAP tests isolating each student's mail.
- Cut thread-open latency from 245ms to ~18ms with a client-side thread cache, and graph drag frame time from 85ms to 12ms by moving physics and hover state off React.

Kourted | Next.js, TypeScript, React, PostgreSQL, Stripe API, Vercel

2025 – Present

- Built and deployed a full-stack facility-management platform for reservations, memberships, invoicing, check-ins, and guest payments, adopted by 7 sports facilities and 4,500+ users.
- Improved schedule-navigation performance through server-side filtering, request caching, prefetching, and client-side state management, cutting week-switch and update render times from 400–3,000ms to under 100ms.

Koherent | Python, FastAPI, NVIDIA NIM & Riva, Next.js, Docker

2026

- Built an AI learning-assessment application that compares student lecture notes with class recordings to evaluate concept coverage and identify explanations, examples, and topics omitted from the notes.
- Engineered the AI pipeline on NVIDIA's hosted inference stack: Riva ASR (Parakeet) for timestamped lecture transcription, NIM embeddings (nv-embedqa-e5-v5) for semantic retrieval over notes and transcripts, and Llama-3.3-70B via NIM for grounded feedback on missed concepts; piloting in a 20+ student course in Fall 2026.
- Built an evaluation harness scoring retrieval quality (hit@k, MRR) and answer groundedness via LLM-as-judge, with all AI adapters behind one dependency-injection seam so the full pipeline is regression-tested offline.