

Kenny Nguyen

US Citizen

(480)-939-0964 | kennytrung0603@icloud.com | linkedin.com/in/kenny-nguyenn | github.com/Intro0

EDUCATION

Arizona State University	Tempe, AZ
<i>Master of Science in Computer Science, Accelerated 4+1 Program</i>	<i>Aug 2026 – Dec 2027</i>
<i>Bachelor of Science in Computer Science, GPA: 3.88/4.0</i>	<i>Aug 2022 – May 2026</i>

Relevant Coursework: Operating Systems, Computer Networks, Data Structures and Algorithms, Software QA and Testing, Software Engineering, Foundations of Machine Learning, Computer Organization and Assembly Language

EXPERIENCE

Opportunity Hack	Remote
<i>Software Engineer Intern</i>	<i>May 2025 – Aug 2025</i>
<ul style="list-style-type: none">Collaborated in a 3-person team to architect and develop a full-stack alumni platform for a global community of over 1,000 users, taking lead on the implementation of the Next.js frontend and Supabase (PostgreSQL) backend.Translated complex UI/UX designs into a dynamic world map using Mapbox GL JS, implementing robust state management to resolve data consistency issues and ensure the stable rendering of 1,000+ alumni data points during real-time filtering.Engineered an end-to-end security model by integrating Clerk for user authentication and implementing Row Level Security (RLS) policies in Supabase to enforce strict data access rules between 'admin' and 'user' roles.Built a comprehensive admin dashboard that enabled the migration of 1,000+ legacy records through a CSV import tool with custom data validation, and integrated Mailgun to facilitate targeted bulk-email campaigns to the user base.	

PROJECTS

Redis Clone | Go, TCP Sockets, Concurrency, RESP Protocol

- Built a **Redis server from scratch** implementing the **RESP protocol**, key-value storage with TTL expiration (PX/EX), and Redis Streams with entry ID validation.
- Engineered **concurrent client handling** using goroutines with **mutex-protected shared storage**, enabling multiple simultaneous client connections on port 6379.

Distributed Storage System | Python, UDP Sockets, Multi-threading, XOR Parity

- Architected a **fault-tolerant distributed storage system** implementing **Block-Interleaved Distributed Parity (BIDP)** for redundancy, enabling automatic recovery from single-disk failures using **XOR-based reconstruction** across networked storage nodes.
- Engineered **parallel I/O operations** using multi-threaded socket communication to concurrently read/write data blocks to multiple disks, with thread-safe synchronization and configurable error injection for real-time parity verification.
- Designed a modular **UDP-based protocol** with 23 command handlers across manager, user, and disk processes, managing user registration, DSS configuration, file ownership enforcement, and coordinated recovery operations.

Shell Clone | Go, Systems Programming, Process Management

- Developed a **POSIX-like shell** with a REPL, builtin commands (exit, echo, type), and **PATH resolution** for locating executables across system directories.
- Implemented **external program execution** using Go's os/exec with proper stdin/stdout/stderr passthrough and executable permission validation.

TECHNICAL SKILLS

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

Frameworks & Libraries: Next.js, React, Node.js, Tailwind CSS, Mapbox GL JS, shadcn/ui

Databases & Services: PostgreSQL, Supabase, Clerk, Mailgun

Developer Tools: Git, Linux/Unix, RLS, Vercel, Claude Code

CERTIFICATIONS

CodePath Intermediate Technical Interview Prep (Advanced), Summer 2025