Yeon Lee

📱 +1 (678) 644-3180 — @ yeonholee50@gmail.com — 🌐 yeonthelee.tech

linkedin.com/in/yeon-lee — github.com/yeonholee50

Objective

Software engineer who builds typed, deterministic services (data pipelines, messaging, observability). I'm looking to join a team where I can ship production systems that measurably improve reliability and speed—while learning from world-class engineers and raising the bar with disciplined ops.

Experience

Attachments King Feb 2025 – Jul 2025

Software Engineer — AI Schema *San Francisco, CA*

- Standardized AI rule authoring across teams by introducing a centralized cursor-rule router; adoption as the single rules pathway reduced fragmentation and simplified maintenance.
- Replaced ad-hoc rules with a tree-based compatibility engine on AWS Neptune that encodes hydraulic power, interface, and loader-arm constraints and emits valid combos dynamically, cutting test-matrix generation from days to under 1 hour.
- Enabled same-day price updates across about 20k SKUs/week and reduced bad-data incidents by >86%, by normalizing PDF/CSV/HTML sources with drift/orphan detection and auto-publishing clean SKUs.
- Removed manual CSR checks for 12+ vendors and improved availability freshness to p95 < 10 min, by deploying a reusable, real-time scraping pipeline for non-API suppliers.

Techrupt Innovations Dec 2024 – Feb 2025

Software Developer *Remote*

- Improved momentum estimator validation accuracy by 0.5 percentage points by refining feature generation and establishing reproducible training and evaluation runs.
- Shortened research cycles by introducing versioned configurations and disciplined run tracking for consistent experiment comparisons under NDA constraints.

LymphaTech Aug 2023 – May 2024

Backend Developer *Atlanta, GA*

- Enabled reliable multi-device access with fewer synchronization conflicts and lower resource use by adopting a two-stage UI–server communication pattern with flag-based sync.
- Held contour measurement error within four percent of ground truth by tuning PyTorch components and integrating Open3D-based contour line generation.
- Improved delivery predictability across biweekly sprints by coordinating Scrum ceremonies and aligning stakeholder feedback via transparent Jira workflows.

Projects

AmpyFin (Platform: OSS foundations + proprietary systems) AmpyFin Org • OSS AmpyFin

- **Purpose** — Plug-and-play trading platform with typed message contracts and deterministic replays; swappable ingestion, transport, configuration, and observability layers.
- **Open-source foundations** — ampy-proto (canonical Protobufs: `bars.v1`, `ticks.v1`, `fundamentals.v1`; explicit decimals; `event_time/ingest_time/as_of`), ampy-bus (standard envelope & headers: run/universe IDs, trace, QoS; NATS JetStream & Kafka), ampy-config (typed, layered config with validation + secret indirection), ampy-observability (uniform logs/metrics/tracing via OTLP), yfinance-go (multi-session free-data path).
- **Provider adapters (modular)** — *DataBento C++ client* (normalized bars/ticks for ensemble learning; bounded concurrency, backoff); *Benzinga Go client* (real-time earnings & news streams for event-driven signals); *Tiingo Go client* (validated fundamentals with currency/period semantics); *yfinance-go* (OSS concurrent pulls with rotating sessions to avoid rate limits).

- **Model orchestration** — Ranked ensemble across specialized systems with weights adapting to performance & market regimes; strict contracts allow side-by-side replay of market data and decisions.
- **Operations** — Reference Docker Compose, golden samples, and CI smoke tests for consistent bring-up, schema evolution checks, and deterministic bus replays.
- **Proprietary systems (selected)** — *Prag* (volatility-aware, risk–reward optimizer), *Hyper* (growth in low-vol), *Riemann* (LLM-ranked analyst signals), *Euler* (regime/volatility forecasting), *Tachyon* (cross-venue pricing), *Aether* (macro & sentiment), *Sigma* (13F portfolio ranking), *Baek* (dynamic fine-tuning & RL), *Val* (consensus fair value).

yfinance-go — *yfinance-go*

- **Purpose** — Free-data ingestion path that matches AmpyFin's proto/bus/config/obs contracts so users can later swap to paid providers (e.g., DataBento) without code changes.
- **Concurrency & rate limits** — Multiple HTTP sessions + a bounded worker pool enable true parallel pulls (e.g., ∼8 concurrent tickers) with rotating sessions to avoid rate limiting.
- **Coverage** — Daily/weekly/monthly/intraday bars and quotes; standardized output to `ampy-proto`; session rotation, backoff, and circuit breakers; library + CLI (`yfin pull`).
- **Fallbacks** — Modular HTML parsing for views not exposed via API endpoints; strict validation on decode to preserve deterministic replays.

NyxHub — *NyxHub*

- **Overview** — Centralized file-sharing app inspired by Greek mythology; secure, fast transfers over a global network with destination-by-username routing and a clean, cross-platform UI.
- **Stack** — FastAPI backend (JWT auth, CORS, structured logging), MongoDB + Motor with GridFS for binary storage, dotenv-based config; ReactJS frontend.
- **Security & speed** — End-to-end encryption for exchanges; in-progress C/C++ module for custom encryption/scrambling to improve throughput and protect at-rest blobs.
- **UX** — Users send files directly to a recipient's handle; recipients retrieve from a dashboard with transfer status.

LeetCode Twitter (🐦) — *LeetCode Twitter*

- **Overview** — "Meme project" turned working social site based on LeetCode's Design Twitter; built with FastAPI + ReactJS.
- **Features (base)** — Post tweets, follow/unfollow, and a personalized news feed with the 10 most recent tweets from self + follows.

Jin Slackbot — *Jin SlackBot*

- **Overview** — Workspace automation bot for message management, reminders, polls, and data interactions, built on Slack API and MongoDB with a maintainable command design.

Education

Georgia Institute of Technology — Aug 2020 – May 2024

B.S. in Computer Science (Threads: Intelligence; Systems & Architecture) — *Atlanta, GA*

- Selected coursework: Data Structures and Algorithms, Systems and Networks, Computer Networking, Design and Analysis of Algorithms, Artificial Intelligence, Computer Vision, Automata and Complexity

Skills

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

Infrastructure: Docker, Render, Linux, Grafana, Prometheus, OpenTelemetry, Kafka, NATS JetStream, Protobuf

Databases and Web: MongoDB, AWS Neptune, DuckDB, REST API, FastAPI, ReactJS, PostgreSQL

Tool: GitHub, Git, Linear, Jira, VS Code IDE, Cursor IDE

Practices: Agile Methodology, SCRUM, CI/CD, TDD, Code Review, Documentation