Neel Gundlapally

936-648-0946 | neelgundlapally@utexas.edu | www.neelgundlapally.me | linkedin.com/in/neel-gundlapally

Education

The University of Texas at Austin May 2026
Bachelor of Science in Computer Science, Minor in Business *Austin, TX*
Relevant Coursework: Data Structures, Discrete Math, Computer Organization and Architecture, Operating Systems, Linear Algebra, Statistics and Probability, Machine Learning, Software Engineering, Cloud Computing, Algorithms and Complexity

Technical Skills

Languages: Python, TypeScript, Javascript, C++, C, Go, Java, SQL, HTML, CSS
Frameworks & Libraries: LangChain, MCP, Next.js, React, Node.js, FastAPI, PyTorch, Apache Flink
Developer Tools & Platforms: Linux, Docker, Kafka, Git, AWS, Firebase, MySQL

Experience

Paycom May 2025 – August 2025
Software Engineer Intern | PHP, Javascript, Python, PyTorch, MySQL *Dallas, TX*
- Led a team of 6 interns to deliver a production-level ML recommendation system for the company's client-facing learning module platform, **boosting course discovery by 10%**
- Engineered an end-to-end, real-time machine learning recommendation pipeline featuring a two-tower model for candidate generation and an MLP-based reranker to optimize top-K suggestions, **achieving a 75% Hit Rate and 0.4 MRR**
- Deployed the full-stack system using FastAPI and React, serving low-latency recommendations to over **10,000 daily users**

Grubtok May 2024 – August 2024
Software Engineer Intern | Flutter *Austin, TX*
- Optimized asynchronous video upload process in Flutter by simplifying user workflow and **reducing user clicks by 40%**
- Shipped cross-platform profile sharing via deep linking and state management, **boosting user engagement by 10%**

The University of Texas at Austin May 2024 – Present
Undergrad Research Assistant at Oden Institute | Python, Jax, Equinox *Austin, TX*
- Led development of a parallelized pipeline automating soft-tissue deformation simulations, **accelerating throughput 40%**
- Innovated a physics-informed neural network (PINN) approach that accelerated the research cycle by 25% by reducing solver convergence time while maintaining 1% relative error by incorporating FEA residuals into the loss function
- Optimized the simulation model for performance by applying model reduction techniques, such as POD and NN-based approaches, **cutting computation time by 30%**

Autonomous Mobile Robotics Laboratory May 2024 – August 2024
Autonomous Robotics Researcher | Python, C++ *Austin, TX*
- Reconfigured a Python-based API into C++, **enhancing performance by 10%** through efficient memory management
- Refactored the Boston Dynamics Spot Robot codebase, increasing maintainability and **reducing code volume by 20%**

Projects

AI Knowledge Management System | *Python, SQLite (FTS5), Claude API, MCP, LLM Hooks*
- Architected a local-first AI memory system **indexing 2k+ artifacts** using SQLite FTS5 and 384-dim embeddings
- Developed a Model Context Protocol (MCP) server to auto-inject semantic context into Claude Code agentic workflows
- Engineered an LLM-powered enrichment pipeline to distill raw artifacts into structured memory, cutting prompt noise

GenAI Football Analytics for the NCAA Transfer Portal | *Python, Qwen*
- Pioneered a novel player-program fit system, discovering 4 QB and 5 offensive scheme archetypes via Hierarchical Clustering
- Designed a Random Forest classifier to predict these QB archetypes, **achieving 79.4% test accuracy (0.79 macro F1)**
- Leveraged a Qwen LLM to generate actionable scouting insights for the NCAA Transfer Portal, identifying undervalued recruits by matching player archetypes to specific offensive schemes

Real-Time Market Data & Analytics Pipeline | *Go, Java, Kafka, Apache Flink*
- Architected an end-to-end pipeline to ingest, process, and serve **thousands of market data ticks with high availability**
- Engineered a stateful Apache Flink job to process **10,000 market data events** per second to calculate real-time trends
- Developed a web service layer providing analytics via a REST API for on-demand queries and a WebSocket feed for updates

Pintos Operating System | *C*
- Engineered kernel multithreading using priority scheduling and synchronization, boosting performance for prioritized tasks
- Integrated demand paging and LRU disk-swapping, **expanding virtual memory by 10% and reducing faults by 20%**
- Built a file system for efficient disk management, allowing for multiple system calls such as exec(), create(), write()