

# Krish Gandhi

U.S. Citizen | 224-334-4525 | [kjg2352@gmail.com](mailto:kjg2352@gmail.com) | [linkedin.com/in/krish-gandhi12](https://www.linkedin.com/in/krish-gandhi12) | [www.krishgandhi.dev](http://www.krishgandhi.dev)

## EDUCATION

<b>University of Illinois Urbana-Champaign</b>	May 2027
<b>Master of Computer Science</b> in Computer Science	<i>Urbana, IL</i>
<b>University of Illinois Urbana-Champaign</b>	May 2026
<b>Bachelor of Science</b> in Computer Engineering, <b>Bachelor of Science</b> in Statistics, Minor in Business	<i>Urbana, IL</i>
<b>Activities:</b> Quant @ Illinois, Financial Engineering Club, Intramural Football, Intramural Basketball	

## EXPERIENCE

<b>Parasol Lab   University of Illinois Urbana-Champaign</b>	Aug. 2025 – Present
<i>Software Developer</i>	<i>Urbana, IL</i>
<ul style="list-style-type: none"><li>Will be using C++ to develop STAPL (Standard Template Adaptive Parallel Library), an open-source framework for developing parallel programs in both shared and distributed memory parallel systems.</li></ul>	
<b>Project Parlay</b>	Jun. 2025 – Sept. 2025
<i>Software Engineering Intern</i>	<i>Orange, CA</i>
<b>Optum</b>	Jun. 2025 – Aug. 2025
<i>Software Engineering Intern</i>	<i>Eden Prairie, MN</i>
<ul style="list-style-type: none"><li>Built and containerized code analysis tool using Python, Docker and AWS ECS to evaluate company codebase.</li><li>Hosted app on AWS EKS with Kubernetes LoadBalancer, distributing traffic and reducing request latency by 87%.</li><li>Designed an event-driven pipeline using GitHub Webhooks, AWS API Gateway, and AWS Lambda to process push events and update code evaluation scores in AWS DynamoDB.</li><li>Leveraged Databricks and scikit-learn to develop a CatBoost classification model to predict patient claim outcomes.</li></ul>	
<b>Quant at Illinois   University of Illinois Urbana-Champaign</b>	Feb. 2025 – Present
<i>Quantitative Developer</i>	<i>Urbana, IL</i>
<ul style="list-style-type: none"><li>Vectorized articles using FinBert embeddings and trained an XGBoost model to predict and trade on price deltas.</li><li>Developed a FX trading strategy in Rust using Floyd-Warshall's algorithm to execute triangle arbitrage trades.</li></ul>	
<b>Financial Engineering Club   University of Illinois Urbana-Champaign</b>	Feb. 2025 – Present
<i>Quantitative Researcher</i>	<i>Urbana, IL</i>
<ul style="list-style-type: none"><li>Trained Random Forest, neural networks, and regression models with PyTorch and XGBoost to predict implied volatility of options.</li><li>Accelerated machine learning models with CUDA, speeding up training time by 68% and inference time by 53%.</li><li>Deployed a full-stack React app to deliver estimations and ML inferences to traders and risk managers.</li></ul>	
<b>Optum</b>	Jun. 2024 – Aug. 2024
<i>Software Engineering Intern</i>	<i>Basking Ridge, NJ</i>
<ul style="list-style-type: none"><li>Created a CI/CD pipeline to automate unit testing of 250+ critical components with Apache Kafka, GitHub Actions, AWS S3 and Java to increase system security and save \$300,000+ in resources and 2,600 man hours annually.</li><li>Built a full-stack web app using Flask to automate network analysis, speeding up process by 81%.</li><li>Developed a script to determine 185 relevant GitHub repos out of 30,000 and parse from the configuration files.</li></ul>	
<b>Pharmacy Associates</b>	Jan. 2024 – Present
<i>Founding Software Engineer</i>	<i>Chicago, IL</i>
<ul style="list-style-type: none"><li>Designed and built cloud infrastructure for data warehousing and CI/CD pipelines with Snowflake and AWS services (S3, Glue, Lambda, Step Functions, IAM) to automatically load and store 1,500,000+ records monthly.</li><li>Created CI/CD pipelines to automatically deploy AWS Lambda and AWS Glue functions using GitHub Actions.</li><li>Deployed machine learning solutions using AWS (Textract) to perform ETL tasks on 50,000+ records monthly.</li></ul>	
<b>Yu Research Lab   University of Illinois Urbana-Champaign</b>	Jan. 2024 – May 2024
<i>Software Engineering Intern</i>	<i>Urbana, IL</i>
<ul style="list-style-type: none"><li>Produced initial states and simulated Markov models with NumPy, pandas and SciPy libraries to find models which yielded distributions similar to real statistics for patients with Major Depressive Disorder or Bipolar disorder.</li><li>Visualized changes in Markov model states over simulated time by integrating Matplotlib and Seaborn libraries.</li></ul>	

- Designed and optimized models of complex systems in MATLAB and Simulink to simulate and visualize generator behaviors based on generator properties and environment conditions, speeding up computing time by 97%.
- Automated data extraction and analysis processes by building internal web scraping and ETL tools with Selenium, NumPy and Pandas libraries, increasing data accuracy and reducing manual processing time by 92%.
- Led cross-functional team efforts on design, development and execution of electrical test equipment and procedures for cutting-edge 3-Phase and 6-Phase standstill frequency response testing, which is now used by a team full-time.

---

## PROJECTS

### Dynamic Compute Cluster | *K3s, SLURM, Ansible, Rust, Go*

- Architecting an 8-node hybrid compute cluster with 64 cores and 128 threads and dynamic switching between Kubernetes (k3s) and Linux SLURM for containerized and HPC-style workloads.
- Developing a custom job scheduler with Rust to estimate resource requirements and route workloads to nodes.
- Designing a job submission and node monitoring dashboard using Go to provide real-time system metrics and logs.

### RISC-V Operating System | *C, RISC-V Assembly, QEMU Emulator*

- Built a Unix-like RISC-V operating system from scratch using C and Assembly, featuring concurrency, memory virtualization, a filesystem, interrupts, device I/O, forks, pipes, paging, syscalls, and user/kernel space separation.
- Virtualized UART, RTC, RNG and memory block VirtIO devices to implement preemptive interrupts, serial communication, random number generation, and filesystem persistence.
- Developed an ELF loader and shell capable of running interactive programs (Trek, Rogue, Zork, and Doom).

### Self-Hosted Private Network | *Tailscale, Ansible, Rust, GCP, MySQL*

- Configured an encrypted mesh network using Tailscale to enable low-latency SSH access across personal devices.
- Designed a diagnostic node to run and log network device status checks, and SSH sessions in a MySQL database.
- Developed and deployed custom diagnostic tools and commands across nodes using Rust and Ansible.

### Order Matching Engine | *C++, Python* | [Project Link](#) | [GitHub Link](#)

- Implemented an order matching engine in C++ with support for market, limit, stop, stop-limit and FOK orders.
- Configured cloud infrastructure to handle real-time event processing, allowing users to interact on the same engine.

### AI Translator and Summarizer | *Python, Hugging Face, GCP* | [Project Link](#)

- Developed an AI text translator web app using Facebook's mBART LLM, enabling multilingual text processing.
- Deployed Google's Gemma 2 LLM on GCP's Vertex AI and set up an API endpoint to deliver real time summaries.

### Doodle Jump on an FPGA | *SystemVerilog, C, FPGA* | [Project Link](#)

- Configured a SoC Spartan-7 FPGA and MicroBlaze processor to handle memory-mapped I/O using Xilinx Vivado.
- Implemented SPI and UART protocol to handle USB and VGA-to-HDMI input/output using SystemVerilog and C.
- Developed a gaming environment with real-time user control, random stage generation, graphics, collision detection, and scoring using SystemVerilog.

### Portfolio Website | *GCP, Docker, Node.JS, Astro, PostgreSQL, Cloudflare* | [Project Link](#) | [GitHub Link](#)

- Built and containerized a full-stack personal site with a FastAPI backend and deployed with Cloud Run using Cloud Build Triggers for continuous deployment.
- Engineered analytics infrastructure using PostgreSQL on Supabase to monitor user engagement and page views.
- Implemented uptime optimization with Cloud Scheduler to prevent cold starts, decreasing latency by 99.7%.

### Virtual Whiteboard | *Python, OpenCV, MediaPipe, NumPy* | [Project Link](#)

- Developed a program to mimic the functionality of a whiteboard by utilizing the user's device's camera.
- Integrated draw, clear, and exit functionality controlled by the user making certain hand gestures at the camera.
- Accessed camera to detect and track 21 landmarks on user's hands by utilizing OpenCV and MediaPipe libraries.

---

## AWARDS/SCHOLARSHIPS

- 2025-26 **Compall Family Scholarship** – Awarded by the ECE Department at UIUC

---

## CERTIFICATIONS

- Bloomberg Market Concepts

## COMPETITIONS

---

### Trading Competitions

- 2025 Quant at Illinois Jane Street Estimathon - 4th Place
- 2025 IMC Prosperity3
- 2025 Quant at Illinois UIUC Trading Competition
- 2024 Bloomberg Trading Challenge
- 2024 CME Group University Trading Challenge

## TECHNICAL SKILLS

---

**Languages:** Python, C++, C, Rust, Java, JavaScript, SystemVerilog, R, RISC-V Assembly, Bash, MATLAB

**Frameworks:** React, Astro

**ML/AI:** PyTorch, scikit-learn, OpenCV, HuggingFace

**Databases/DBMS:** Snowflake, Databricks, AWS RDS, AWS DynamoDB, MongoDB, PostgreSQL, MySQL, SQL

**Cloud Computing:** AWS, GCP, Supabase, Render, Vercel

**DevOps:** Docker, Kubernetes, Terraform, Ansible, Jenkins, Apache Kafka, GitHub, GitHub Actions, SSH

**Coursework:** Distributed Systems, Database Systems, Algorithms, Communication Networks, Applied Machine Learning, Computer Systems Engineering, Data Structures, Signal Processing, Applied Bayesian Analysis, Statistical Learning, Statistics and Probability, Statistical Modeling, High Frequency Trading, Financial Engineering