

# Colin Curtis - AI Systems and Software Engineering

 [colinkeurtis](#) |  [Colin Curtis](#)

## PROFESSIONAL SUMMARY

---

Software engineering in the AI era is a complex and fast-moving profession where team-work and clear communication is as important as technical skill. Delivering great customer experience and value is the goal. In order to consistently reach that goal, cleanly-architected systems and a focus on cost-effective practices are the vehicle.

## EDUCATION

---

2012 - 2014 M.S. (Physics) at **North Carolina State University**

2007 - 2012 B.S. (Physics w/ Math Minor) at **Appalachian State University**

## SKILLS AND INTERESTS

---

- Professional
  - C4 software system design
  - SOLID, YAGNI, DRY coding principles
  - ICE product/feature design
  - Linear Algebra, Calculus, Statistics, Error Analysis
- Personal
  - Gardening, Cooking, Sci-fi
  - Mountain Biking, Skiing, Running, Soccer

# WORK EXPERIENCE

---

## Machine Learning Ops and Software Engineering - Fidelity Investments April 2023 - Present

- Enterprise Data and AI (EDAI), AI/ML Solutions Engineering Squad
- Cloud Systems and Tech Stack:
  - AWS: Sagemaker, Cloudwatch, ECR, S3, Bedrock, RDS, EKS, IAM, Cost Explorer
  - Azure: OpenAI (OAI) portal + Tofu (Terraform) for model deployment/management
  - python 3: botocore, flask, fastapi, pydantic, openai, streamlit, pandas, numpy, sqlalchemy, etc.
  - Docker, kubernetes, Splunk, Argo, Jenkins, Snowflake, Datadog, jupyter NB
- Projects and Responsibilities:
  - Implemented enterprise-grade JWT authentication system to enhance platform security, enabling identity management and request attribution for data science workflows; enables fine-grained access control and audit capabilities while maintaining compatibility with existing OAuth2-token authentication flows
  - Modularized MLOps model server build pipeline (Triton, DJL, custom) using Docker multi-stage builds and containerization best practices, enabling CI/CD efficiency and faster developer velocity by isolating build layers based on user requirements
  - Implemented automated E2E testing and validation for model building pipeline to ensure consistent delivery, accelerate bug detection, and improve deployment reliability across ML infrastructure
  - Wrote signing and composition logic for pure HTTP requests to be used throughout our codebases in order to factor out CSP SDKs (boto3, etc.) that had begun to bog down our systems with dependency-complexity
  - Re-worked automation code (Jenkins) for integration and performance tests on our primary customer-facing application, GenAI Gateway; upgraded the test-runners to run on a weekly schedule in addition to manual usage; modified the test suites to upgrade lower-env deployed applications to their latest version(s) before launching tests against them
  - ML-training/inference cost-savings initiative, five-figure monthly savings outcome
  - SSO-authentication flask wrapper to secure data science web apps
  - Tested deployed models for LLM token-usage and response-time metrics
  - Extensive code review and deployment support for engineers and data scientists across teams
  - Handled database migrations, surgery and any other troubleshooting/fixing that needs doing
  - Defined and added new pieces of work to the JIRA ticket queue
  - Onboarded GenAI-Gateway to Fidelity's Enterprise-wide API platform, including careful documentation for team-internal as well as customer usage
  - Anticipate the needs of, and potential issues within, our code and tooling before they arise

## Senior Software Engineer - Garner Health

August 2022 - February 2023

- Systems and Tech:
  - Python 3: connexion/flask, psycpg2, sqlalchemy, boto3, pytest, alembic, asyncio, behave
  - AWS: S3, VPC, IAM, RDS, ECR, EKR, Step Functions, Lambda, Transfer Family, Secrets Manager
  - Docker, Kubernetes, Terraform
- Projects and Responsibilities:
  - PGP decryption module for the file ingestion system
  - Designed and implemented Client Data Configuration Versioning
  - System testing of RESTful APIs using Behavior Driven Design (behave library)
  - Product Requirement Documents for engineering/management communications

## **Software Engineer - Actalent Services**

April 2019 - July 2022

- Systems and Tech:
  - Microservices architecture for Bridgestone's tire-design CAD web application
  - Python 3, react.js, numpy, logging, pytest, mongoDB
- Projects and Responsibilities:
  - Designed and built a Dynamometer Dashboard for engine test data analysis at Ford Motors
  - Contributed to Bridgestone tire design CAD/analysis system written in Python 3 and React.js
  - Added 'overlays' to the CAD system, allowing tire engineers to view a transparency of one tire design overlaid on to another
  - Refactored the automatic Excel report generation - tire engineers click a button in the CAD system and the Python back-end generates a downloadable report
  - Converted and upgraded FMAT, a tire image analysis tool, from MATLAB to Python 3 and introduced 'alpha-shapes' computational geometry technique to improve the boundary estimates of complex 2-D shapes

## **Junior Software Engineer - Renaissance Computing Institute**

June 2018 - March 2019

- NIH National Center for Advancing Translational Sciences (NCATS) - Biomedical Data Translator
- Wrote and deployed APIs using Marathon, Mezos, Github, Jenkins, Docker, nginx, and python3

## **Research Assistant, NCSU Physics**

May 2013 - April 2018

- Designed and coded ALAI, a MATLAB application for automating fractal analysis of nanoscopic images
  - Reduced the user's active analysis time, per image, by a factor of ~50
- ANSYS Maxwell mesh-calculation to simulate 3-D electromagnetic fields
- LabVIEW software systems for instrument control and data collection
- Author/co-author of 3 peer-reviewed papers in experimental surface physics