Dylan Jones | Software Engineer (AI/Cloud)

540-488-2999 | Roanoke, Virginia | djones111797@gmail.com | [LinkedIn](#) | [GitHub](#) | [Portfolio](#)

SUMMARY

Software Engineer focused on ML/AI and cloud-based applications, with experience building and deploying a healthcare-focused generative AI product using Python, Flask/FastAPI, Google Cloud, AWS, Docker, and Flutter. Since September 2023 I've worked full-time at a digital health startup, integrating LLM services into HIPAA-aligned medical workflows and shipping cross-platform mobile and web features. Outside of work I've built ML projects using scikit-learn, fastai, and PyTorch (including a cardiovascular risk predictor and an image classifier using CNNs and transfer learning), and I'm now looking to go deeper into ML/AI systems and MLOps, focusing on scalable pipelines, model orchestration, and production AI infrastructure in a cloud-first environment.

TECHNICAL SKILLS:

Languages & Frameworks: Python, Flask, FastAPI, Django, JavaScript, Node.js, Express.js, SQL

ML & Data: scikit-learn, fastai, PyTorch, CNNs & transfer learning, Jupyter Notebooks, Kaggle, MySQL, PostgreSQL, MongoDB, Mongoose

Cloud & DevOps: Google Cloud Platform (GCP), AWS (S3, IAM, EC2), Docker, CI/CD, CLI-based deployment workflows, GitHub

Tools & Platforms: Git, GitHub, Android Studio, Xcode, VS Code

INTERPERSONAL SKILLS: Good Communication, Teamwork, Problem Solving

---

Software Engineering Experience

TeachMeMedical | Software Engineer (AI/Cloud) | September 2023 - Present | Python, Flask, FastAPI, GCP, Docker, MySQL,   Pinecone, Cloud Healthcare API (FHIR)

TeachMeMedical is a digital health product that utilizes generative AI to explain medical records in clear, accessible language. It securely analyzes personal medical data to generate accurate explanations for patients.

- Built secure cloud-based AI services in Python/Flask (and FastAPI where applicable), integrating LLM-driven medical explanation workflows on Google Cloud.
- Designed token-safety controls with automatic summarization and session refresh to prevent context overflows in production.
- Integrated PHI de-identification into LLM request pipelines to reduce privacy risk in healthcare AI use cases.
- Implemented service-account-based access patterns and secure secret/config handling for healthcare AI services on GCP.
- Used Git, GitHub, Docker, and CLI-driven workflows to support consistent environments, code review, and iterative delivery of new AI features.
- Contributed to data modeling and MySQL schema design to support secure storage and retrieval of medical content and AI-generated explanations in an auditable manner.
- Integrated Google Cloud Healthcare API (FHIR) for structured patient-related data storage and retrieval.

Patient Risk Analysis (ML) | personal project | [App](#) | [GitHub](#) | Python, scikit-learn, FastAPI, pandas, Docker

End-to-end ML web app that predicts 10-year coronary heart disease (CHD) risk using the Framingham dataset with a deployed FastAPI inference API and lightweight UI.

- Built a scikit-learn Pipeline with median imputation, standardization, and logistic regression to predict 10-year CHD risk from clinical features.
- Persisted the trained model + feature metadata with joblib and loaded it at API startup for consistent preprocessing in production inference.
- Developed a FastAPI service with Pydantic request/response models and a health endpoint for structured, reliable model serving.
- Containerized the app with Docker and deployed to Render, providing a minimal UI for probability, risk banding, and top-factor explanations.
- Documented reproducible training + serving workflow (train ➡ export ➡ load ➡ predict).

AI/ML Forest ID | Personal project | [GitHub](#) | Python, fastai, PyTorch, torchvision, Jupyter Notebook, NumPy, Git/GitHub

End-to-end fastai/PyTorch computer vision project that classifies forest images into giant sequoia, oak, and pine with a trained ResNet34 model and a local CLI for reproducible inference.

- Trained and evaluated a ResNet34 model with fastai, using data loaders, augmentation, and confusion-matrix-based analysis.
- Worked under offline/SSL-restricted constraints, documenting reproducible training steps and clear improvement paths.
- Structured the repo to mirror real-world MLOps practices, keeping code reproducible while excluding large data/model binaries from git.

ADDITIONAL EXPERIENCE

Data Analytics, Holly Insurance, October 2021 - May 2023
- Queried relational datasets in SQL to support performance reporting, anomaly detection, and business decision-making
- Developed Excel models and dashboards to monitor claims/policy trends and operational KPIs.

EDUCATION

General Assembly, Remote, Software Engineering Immersive, June 2023 - August 2023
Radford University, Radford, Virginia, Completed coursework in Business Management, 2016-2018, 2020-2021