

Rajeev Atla

AI-Native Software Engineer Building Secure Backend Systems, using Automated Cloud Infrastructure

US Citizen | [732-209-3995](tel:732-209-3995) | rajeev.atla@gmail.com | linkedin.com/in/rajeev-atla | github.com/RajeevAtla | rajeevatla.com

EDUCATION

Princeton University (Exchange Student)/Rutgers University - School of Engineering Sep 2025 — Dec 2026
Master of Science in Computer Engineering (Specialization in Machine Learning) Princeton/New Brunswick, NJ

Coursework: Multimodal AI, LLMs, Reinforcement Learning, Machine Vision, AI for Precision Health, High Performance Computing

Rutgers University - School of Engineering Sep 2021 — May 2025
Bachelor of Science (Triple Major) in Computer Engineering, Computer Science, and Data Science New Brunswick, NJ

Recipient of the Eleanor and Samuel Sneath Endowed Merit Scholarship for Engineering Students

Coursework: AI, ML, Distributed Deep Learning, Data Science, Robotics and Computer Vision, Info and Network Security

SKILLS

- **Programming Languages**: Python, R, SQL, Java, C/C++/CUDA, JavaScript/TypeScript, Rust, Bash, Go
- **AI/ML**: NumPy, PyTorch, JAX, TensorFlow, Keras, Pandas, Scikit-Learn, OpenAI API, LangChain/LangGraph, OpenCV, DSPy, RAG, HuggingFace (Transformers, Tokenizers, Datasets, Diffusers), vLLM, pgvector, Pydantic, FastAPI, NLTK, spaCy
- **Visualization**: Matplotlib, Seaborn, Plotly, Tableau, PowerBI, React
- **Cloud & DevOps**: AWS, Microsoft Azure, OCI, GCP, GitHub Actions, Jenkins, Docker, Kubernetes, Slurm
- **Tools & Databases**: Jupyter, PySpark, Hadoop/Hive, Git, Linux, Flink, Kafka, Jira, PyTest, Codex, Claude Code

CERTIFICATIONS

- **AWS**: [Certified Cloud Practitioner](#), [Certified Machine Learning Specialist](#), [Certified AI Practitioner](#)
- **Oracle (OCI)**: [AI Foundations Associate](#), [Generative AI Professional](#), [Data Science Professional](#), [Vector AI Search Professional](#)

WORK EXPERIENCE

AI Engineering Intern May 2024 — Sep 2024
Atlait Inc. Remote

- Integrated DevSecOps within GitHub Actions, **improving risk mitigation by 13%** for production
- Engineered PyTorch inference models for real-time predictions, **optimizing latency by 96ms** and enabling faster decision-making
- Created a > **1TB** RAG-PySpark system, utilizing A/B testing to optimize AI-powered search and recommendation accuracy
- Optimized Airflow-Hadoop data pipeline to **speed up analytics delivery by 13%** in a cross-functional Agile environment

PROJECTS

DocuMint <https://bit.ly/DocuMint>
• Built a 5-agent LangGraph + Gemini API doc-modernizer with Gradio, achieved **90%+ modernization coverage** on sample docs, **cut manual edit time by 50%** with a **4-tab UX**, hardened with **8 deterministic Pytest cases** and network-safe skips
• Authored a modular multi-agent system with structured prompts and severity-prioritized research, **lifting modernization accuracy by 35%** and **trimming LLM API spend by 20%**

dexMCP <https://bit.ly/dexmcp>
• Engineered Model Context Protocol (MCP) server exposing **5+ reusable tools** and **5+ Pydantic models**
• Implemented parameter validation across **20+ typed fields** and **100% of tool inputs** using Pydantic
• Built asynchronous clients using **DSPy** and **LangChain** to auto-discover tools and execute multi-step agentic workflows

SuperconGAN <https://bit.ly/3z7JaqZ>
• Built a PyTorch-based GAN to create synthetic superconductivity data of various materials, enhancing generative AI applications
• Extracted and processed **80,000+ dataset entries** from the UCI ML Repository using Pandas efficiently
• Released Python package on PyPI, achieving over **80,000 downloads** and widespread adoption

Cityscape Mobile Tour App (Won 2nd Overall at HackExeter 2021) <https://bit.ly/3OZjJ07>
• Wrote controllers and models for MongoDB using Mongoose ODM to store **30+ kB of geographic data** in NoSQL schema
• Built mobile user interface allowing users to search, review, rank, and explore **100+ tours** using Flutter/Dart
• Constructed REST API using Express.js and nodemon to **increase development velocity by 20%** with hot-reloading