Christopher Colosimo

Cleveland, OH | Colosimocj3@gmail.com | github.com/odysian

SUMMARY

IT Programming & Development student with hands-on experience building and deploying full-stack web applications using Python, FastAPI, PostgreSQL, and React. Practical AWS infrastructure and Terraform experience from deploying production applications. Currently building an AI-powered document analysis platform using RAG architecture and vector embeddings.

EDUCATION

Cuyahoga Community College, Cleveland, OH   *Expected 2027*
Associate of Applied Business in IT Programming & Development
Relevant Coursework: Enterprise Database Systems, Java Programming, Interactive Internet Programming, Programming Logic, Web Programming

TECHNICAL SKILLS

Backend: Python, FastAPI, SQLAlchemy ORM, REST API design, JWT authentication, WebSockets
Databases: PostgreSQL, MySQL, pgvector, SQL (joins, aggregations, subqueries, indexing)
Infrastructure: AWS (EC2, RDS, S3, ElastiCache, CloudWatch), Terraform, Docker, CI/CD with GitHub Actions
Frontend: React, TypeScript, Next.js, Tailwind CSS (working knowledge for full-stack development)

PROJECTS

Quaero — Document Intelligence Platform | *FastAPI, PostgreSQL, pgvector, Next.js, TypeScript, OpenAI, Anthropic API*
quaero.odysian.dev
- Built a PDF processing pipeline that extracts text and splits documents into semantic chunks for retrieval
- Implemented vector embeddings with OpenAI and pgvector, enabling semantic search across uploaded documents
- Developed a RAG-based Q&A system using Anthropic API that answers questions with cited source passages
- Secured all endpoints with input validation, rate limiting, and XSS/SQL injection prevention

FAROS — Task Management Platform | *FastAPI, PostgreSQL, Redis, React, Terraform*
faros.odysian.dev
- Developed a full-stack task manager with JWT authentication, CRUD operations, commenting, and file uploads to S3
- Integrated Redis caching for task and user data, reducing redundant database queries on repeated page loads
- Defined AWS infrastructure in Terraform (EC2, RDS, S3, ElastiCache) for reproducible deployment; hosted on Render
- Implemented task sharing with role-based access control, allowing users to collaborate across accounts

Rostra — Real-Time Chat Application | *FastAPI, PostgreSQL, WebSockets, React, Redis*
rostra.odysian.dev
- Built a multi-room chat application using WebSocket connections for real-time bidirectional messaging
- Implemented user authentication with secure session handling and persistent message history in PostgreSQL
- Added unread message tracking with Redis caching, providing per-room notification counts without repeated queries
- Deployed frontend to Vercel and backend to Render with Supabase PostgreSQL for a fully hosted environment

EXPERIENCE

Self-Employed Landscaper, Brunswick, OH   *2020 – 2025*
- Managed project planning, budgeting, and client coordination for residential landscaping operations
- Applied systematic problem-solving and attention to detail transferable to software development