Nolan Ley Custodio

Email: custodionolan@gmail.com | Website: nolancustodio.xyz | Github: https://github.com/NolanCustodio

SKILLS

Tools: Python, PHP, HTML, CSS, Javascript, BASH, Git, MySQL, AMQP, RabbitMQ, Trello, Cron, Bootstrap, MVC, Docker, ReactJS, Nginx, Express, Cloudflare, MongoDB, Prisma, SolidJS, Typescript

Operating Systems: Windows 10, Linux (Ubuntu, Alpine)

Hosting: AWS (Elastic Beanstalk, EC2) & DigitalOcean (Droplet)

EDUCATION

New Jersey Institute of Technology (NJIT) Attended between Jan 2018 - May 2021
Bachelor of Science - Information Technology

EXPERIENCE

NJIT, Ying Wu College of Computing September 2019 - December 20, 2021
Adjunct Professor(Ending Position)

- Introduced basic programming concepts through Python
- Graded and reviewed assignments to over 25 students every semester
- Gave supplementary lectures/lessons to students every week ranging from individual to group lecture

PROJECTS

Portfolio Website: Static Website Hosted on the DigitalOcean - Winter 2025
Updated with features as a testing ground for old and new tools.

- Docker containers for a React front-end and an Nginx web server proxy
- Docker-Compose used to handle creation, usage, and new instances of containers
- Docker Network is used to communicate within the machine and the two containers
- Nginx handles all incoming TCP requests and creates pages from React build
- Cloudflare is used to secure the connection and provides an SSL Certification

Synchat: MERN Stack Application - Ongoing
Web Application with group chat and scheduling as a focus.

- Uses Docker to create 5 mico-services
- Implemented RabbitMQ with RPC to handle data fetching
- SolidJS front-end for lightweight and quick page creation
- Prisma and MongoDB for database and query
- Express used to handle API calls to backend and session/cookie checking

NewsBox Back End Developer: Systems Integration Project - Fall 2020
A group project which involved obtaining data through a news API, using multiple virtual machines that used RabbitMQ to communicate on a virtual LAN, and providing a dynamic front-end.

- Virtualbox was used to create virtual machines for each service
- The VMs all used Ubuntu and LAMP was the tech stack that we used
- Linux tools such as Systemd, cronjobs, iptables, and more were used to secure, facilitate, and automate
- Utilized Apache as the web server and made sure to create a secure connection with SSL
- MySQL database formation and data backup was scripted using BASH
- Created an MVC from scratch for this project to handle navigation and page data
- PHP was used throughout the project it was used in handling both front-end and back-end operations
- RabbitMQ facilitated communication between VMs
- Used a News API to dynamically populate custom homepages for users