

Fabricio Rodriguez

☎ 240-439-5354 | ✉ frodrig3@umbc.edu | 🔗 linkedin.com/in/fabricio-rodriguez-816676255 |
📁 github.com/rodriguezzfabricio

EDUCATION

University of Maryland Baltimore County

Bachelor of Science in Computer Science GPA: 3.7

Baltimore, MD

Aug. 2022 – May 2026

EXPERIENCE

Incoming Software Engineer Intern

Finra

March 2025 – Present

Rockville, MD

Data Science Research Assistant

American University

Jan. 2024 – Sept. 2024

Washington, DC

- Achieved **97% accuracy** in predictive modeling by developing a **Machine Learning model** using **Python, TensorFlow, and scikit-learn**.
- Created and deployed a **REST API** using **Spring Boot and PostgreSQL** to provide **real-time** data access for team analysis.
- Developed an efficient **data cleaning script** in **Python**, improving ML model training time.

PROJECTS

Shell Implementation | *C++, Filesystem, POSIX*

- Built a **Unix-like shell from scratch**, implementing core commands **cd**, **pwd**, and **type** for file system navigation.
- Utilized the **C++ Filesystem library** to manage **directory structures and file operations**.
- Implemented **process control** with **POSIX APIs fork, execvp** to reliably execute external commands.
- Enhanced user experience by designing features for **absolute path navigation** and **current working directory retrieval**.
- Ensured **robust error handling** and adherence to **shell scripting conventions** through iterative testing.

Retriever Study | *Java, Spring Boot, MongoDB, React, Google Calendar API*

- Designed an **online platform** connecting students for study groups using **Spring Boot backend** and **React interface**.
- Developed **REST APIs** for **user authentication, group management, and scheduling**, integrating the **Google Calendar API** to streamline study sessions.
- Improved **accessibility for 100+ students** by optimizing **authentication and course selection** features.

Kernel-Level CPU Scheduler | *C, Linux Kernel, Multithreading*

- Implemented a **custom CPU scheduler** as a **Linux kernel module** with **syscalls** for **task creation, resource management, and scheduling**.
- Built and tested user-space interfaces to invoke kernel **syscalls** and validated thread safety using **mutexes, semaphores, and dmesg logging**.
- Designed **priority queue structures** and implemented a **consumer thread** in kernel space to simulate **preemptive task execution**.
- Integrated user-space test scripts and **Valgrind tools** for memory safety, while leveraging **kmalloc** and **synchronization primitives** in kernel space.

TECHNICAL SKILLS

Languages: Python, C/C++, SQL, Java, JavaScript, HTML/CSS, R, Swift, TypeScript

Frameworks: React, Node.js, Flask, Spring Boot, AWS, Django, Agile

Databases: MongoDB, PostgreSQL, MySQL

Developer Tools: Git, Docker, VS Code, Visual Studio, PyCharm, R Studio, Excel, Word, PowerPoint, Unix, Linux

Libraries: pandas, NumPy, Matplotlib, Seaborn, TensorFlow