ALESSANDRO LORENZO
lorenzoa@umich.edu • (786) 405-3280 • https://www.linkedin.com/in/alessandrolorenzo • https://lorenzoa3.github.io/portfolio

EDUCATION

University of Michigan **Ann Arbor, MI**
Bachelor of Science in Engineering in Computer Science *December 2024*
GPA: 3.42 / 4.00
Relevant Coursework: Software Engineering, Data Structures and Algorithms, Operating Systems, Web Systems & Development

SKILLS

Programming Languages: C++, Python, C#, JavaScript
Technologies & Tools: Git, HTML, CSS, Unity, SQL, Docker, Linux, .NET, Jira
Languages: Spanish (Native Proficiency)

WORK EXPERIENCE

Outlier **Remote**
Software Developer *December 2024 – Present*
- Developed and executed numerous robust unit tests to rigorously evaluate AI-generated solutions, ensuring 100% adherence to prompt constraints, functionality requirements, and edge case scenarios, significantly amplifying model reliability
- Improved artificial intelligence model performance and usability, decreasing output errors by over 30%, identifying and addressing logic gaps in code design, and creating detailed test plans to optimize AI response accuracy and user satisfaction
- Enhanced prompt accuracy by 25% through the refinement of 200+ coding prompts, providing AI with clear pseudocode, context-specific hints, and detailed explanations improving comprehension and higher quality AI-generated solutions in C++ and Python

University of Michigan **Ann Arbor, MI**
Researcher, Machine Learning *September 2022 – May 2023*
- Annotated 2500+ syllabi in Label Studio, validating a specific definition for significant writing component; trained machine learning model for annotation with an 80/20 test split, utilizing natural language processing to scale human annotation to corpus
- Web scraped 345,289 syllabi from public Texas universities in Python for research aimed at determining courses with a significant writing component, employing a text-as-data approach, achieving higher accuracy in identifying criteria
- Programmed 50+ data visualizations with Python to articulate model performance and insights to faculty; findings aligned with expectations of a lower number of robust writing courses at under-developed institutions

TECHNICAL PROJECTS

Trading Engine Server | C#, .NET *December 2024 – January 2025*
- Engineered a multi-instrument trading engine from scratch using C#, implementing core features such as order book management, bid-ask spread calculation, and support for order addition, modification, and cancellation with extensible interfaces for scalability
- Built a custom logging system, including text, database, trace, and console loggers with support for asynchronous logging and thread safety, directing detailed insights into system operations, errors, and performance metrics to local .log files
- Implemented a price-time priority matching algorithm to efficiently execute trades by prioritizing orders based on price competitiveness and submission time, ensuring order integrity through username validation, conflict resolution, and error handling

Network File Server | C++, Docker, Linux *June 2024 – August 2024*
- Designed and implemented a multi-threaded, secure network file server in C++ capable of handling 500+ concurrent file operations, optimizing system performance under high traffic and ensuring reliable request handling without bottlenecks.
- Devised innovative measures to integrate advanced security protocols between clients and server; assured protection of sensitive directories while handling over 500 concurrent user requests for elevated data integrity
- Optimized file system performance by implementing synchronization techniques, including locks and condition variables, resulting in a 65% increase in system throughput and no deadlocks

Full-stack Social Networking Platform | Python, JavaScript, HTML, CSS, SQL *May 2024 – July 2024*
- Built a full-stack social media platform leveraging dynamic server and client-side web pages with Python, HTML, CSS, and JavaScript, enabling asynchronous updates via AJAX resulting in seamless user experience with real-time page interactions
- Integrated REST APIs and honed SQL database queries to dynamically manage and retrieve user data, enabling core features such as likes, comments, shares, and follows, improving application responsiveness by more than 50%
- Developed and deployed backend API and database architecture, reduced data retrieval latency by 10%, while securing user data through a robust session-based login mechanism

Rent-A-Bot | C#, Unity, Jira *September 2023 – December 2023*
- Collaborated with a team of four adopting Agile development to build a co-op game with Unity and C#, applying game design concepts such as guidance, progression, and game physics, resulting in an engaging multiplayer experience
- Designed and pioneered gameplay mechanics, utilizing a TDD workflow to maintain quality and consistency, with a focus on novelty, theming, and game juice, resulting in a 20% increase in game attraction and enhanced gameplay engagement
- Led game testing and debugging in Jira for project management to strengthen technical soundness and smooth performance, resolving critical issues and delivering a polished, high-quality final product acquiring 2nd place at Michigan's Game Showcase