Caleb L'Italien

+1 934-500-4492 | litaliencaleb@gmail.com | linkedin.com/in/caleblitalien | github.com/CalebLItalien

Education

Union College — Schenectady, NY
Bachelor of Science in Computer Science, Minor in Mathematics — *Sep. 2020 – Jun. 2024*

Skills

Languages: C, JavaScript/TypeScript, Java, Python, HTML/CSS
Frameworks & Libraries: React, Angular, Express.js, Node.js, Mongoose, Bootstrap
Databases : MongoDB, Firebase, MySQL
Tools & Services: Git, Amazon Web Services (EC2), Azure (Cosmos DB), Docker, VMware

Experience

Full Stack Applications Developer — Jun. 2023 – Sep. 2023
Diamond Diagnostics — *Holliston, MA*

- Worked in a fast-paced business environment with a legacy codebase and deployed client needs to cloud instances
- Spearheaded development and wide-scale integration of robust middleware responsible for deserializing user data and managing HTTP requests using Express.js and Node.js
- Created efficient data pipelining, automating internal tasks using MongoDB and Mongoose
- Introduced new features, including label generators, Microsoft Outlook integration, and security management
- Overhauled existing routes, simplifying and generalizing for enhanced accessibility to future developers

Robotics Research Assistant — Sep. 2022 – Present
Union College — *Schenectady, NY*

- Worked in a large lab (20+ students) to simulate soft robotics movement in a 2D environment
- Optimized genetic algorithms for calculating ideal movement strategies
- Implemented Docker containerization and configured VMware setups for Ubuntu Linux virtual machines
- Managed resource allocation using Chameleon cloud services

Fundraiser — Dec. 2021 – Nov. 2022
Hudson Bay Company — *Farmingdale, NY*

- Fundraised over $15k canvassing door-to-door on clean water initiatives
- Worked alongside program directors and executives to help revitalize headquarters' canvassing operation
- Trained three new fundraisers and held them accountable to meet standard fundraising goals

Projects

OS/161 | *C*

- Worked in a team of three students to design a fully functional operating system
- Optimized process/thread implementation and memory management
- Implemented essential system calls, including fork(), waitpid(), and exit()

ToyPL | *Scheme*

- Designed a toy programming language, with a lexer, parser, and interpreter
- Implemented advanced features, such as automatic heap management and references
- Implemented basic primitives, immutable/mutable variables, and block expressions

Cherokee Language Exercises | *TypeScript, React, Python, Firebase*

- Transformed application into a Progressive Web Application (PWA), creating offline functionality for mobile and desktop devices with audio caching
- Reconfigured and standardized data pipelining processes using UUIDs
- Configured GitHub actions to run tests upon pull requests