THOMAS HART

thomas.hart@uwaterloo.ca thomashart.tech 778-387-0195 thomashart17 thomashart17

SKILLS

Languages: C++, Java, Python, C, Verilog, Rust, HTML, CSS, JavaScript, LaTeX, VHDL, ARM Assembly, SQL
Frameworks/Libraries: Langchain, Android SDK, AOSP, React.js, Django, SeaHorn
Tools: Linux, Git, CMake, VS Code, Android Studio, Visual Studio
Hardware: Arduino, Raspberry Pi, STM32

EXPERIENCE

Founder
AI Docs · Jan. 2024 to June 2024 · Vancouver, BC

- Developed an AI-powered tool to automate the process of writing codebase documentation and enhance developer productivity.
- Leveraged prompt engineering and fine-tuning techniques to ensure the tool delivered consistently accurate and detailed documentation.
- Implemented seamless integrations with platforms like GitHub, enabling automatic documentation updates with every code change.
- Engaged with potential customers to identify common issues in existing documentation practices, using feedback to guide design decisions.
- Awarded the Overbeeke Family Entrepreneurship Excellence Award from the University of Waterloo's enterprise co-op pitch competition.
- Made the strategic decision to discontinue the project after determining that existing LLMs lacked the necessary accuracy and contextual understanding, and that developing a new LLM was not feasible due to financial constraints.

Research Assistant
University of Waterloo · May 2023 to Aug. 2023 · Waterloo, ON

- Collaborated with a team of researchers supervised by Professor Arie Gurfinkel to verify Rust code using the SeaHorn verification framework.
- Investigated alternatives to Rust's standard library vector class in order to improve runtime performance of verification jobs.
- Demonstrated SeaHorn's effectiveness by creating verification jobs to detect critical errors in legacy versions of popular Rust crates.
- Developed a custom Python script to automate the generation of boilerplate code for verification jobs, accelerating development times.

Software Engineer
Peraso Technologies · Sept. 2022 to Dec. 2022 · Toronto, ON

- Contributed to the development of firmware and internal tools for 5G radio devices using C++ as a member of the Device Software team.
- Designed a custom XML parsing and generation tool, streamlining data input for an EEPROM programming application.
- Improved CLI code stability and efficiency by improving error handling and eliminating all crashes caused by invalid user input.
- Optimized CLI command outputs by revising commands to improve clarity and eliminate redundant information, enhancing usability for users.

Autonomous Vehicle Android Developer
Ford Motor Company · Jan. 2022 to Apr. 2022 · Remote

- Built high-quality Android applications for an in-vehicle infotainment system using Java, optimizing for both performance and user experience.
- Demonstrated technical expertise by leveraging hidden AOSP classes to implement a critical feature, despite minimal documentation.
- Migrated key features to the latest version of Android, enabling the adoption of new technologies and ensuring long-term compatibility.
- Implemented key UI components for the system's flagship application, improving the overall system functionality and ease of use.

PROJECTS

Finance Translator (Python, Django, HTML, CSS)

- Developed a web application using Django to simplify complex financial text using NLP through the Cohere API.
- Implemented a trie-based search algorithm to quickly identify relevant keywords in the text and provide links to corresponding definitions.
- Integrated real-time stock price display for companies mentioned in the input text, utilizing the TradingView API.

Arduino Stock Ticker Display (Arduino, C++, Python)

- Developed a real-time stock ticker display utilizing an Arduino Uno WiFi, delivering live stock prices via a connected LCD display.
- Designed an algorithm to periodically retrieve data from the Yahoo finance API, enabling users to track a custom portfolio of stocks.
- Created a Python script to automatically update the code when users modify their portfolio, eliminating the need for manual updates.

Crypto Terminal (Python)

- Designed and developed a user-friendly Python command line application to check various cryptocurrency statistics.
- Integrated the Coin Gecko API to retrieve real-time price data for the top 1000 cryptocurrencies, offering users up-to-date information.
- Utilized the mechanize and BeautifulSoup4 libraries to scrape Minerstat, allowing users to check the current profitability of mining hardware.

EDUCATION

University of Waterloo · Sept. 2021 to Apr. 2026
Candidate for Bachelor of Applied Science in Computer Engineering | GPA: 3.74
Relevant Courses: Algorithms and Data Structures (C++), Systems Programming and Concurrency (C), Real-Time Operating Systems (C, ARM), Compilers (Java), Computer Architecture (Verilog), Computer Networks (Python), Database Systems (SQL), Digital Computers (ARM)