Sayed Hayat Ahmad

647-938-5883 | sh2ahmad@uwaterloo.ca | linkedin.com/in/hayatahm | hayatahmad.xyz

Education

University of Waterloo **Present**

BASc in Honors Computer Engineering *Waterloo, ON*

- **Relevant Coursework:** Electricity and Magnetism - Linear Circuits - Digital Circuits

Technical Skills

Languages: Python, C++, C, C#, VHDL, JavaScript, HTML, CSS, GDScript
Developer Tools: Altium Designer, KiCad, COMSOL, Intel Quartus Prime, Solid Works, Blender, LTSpice
Framework & Technologies: React.js, Three.js, Node.js, RESTful API, CircuitPython, MicroPython

Experience

NETSOL Technologies | *AI/ML Engineering Intern* **May 2025 - Present**

- Spearheaded the development of an **Agentic** reimbursement system, automating receipt scanning, intelligent data extraction, and PDF form auto-submission; successfully pitched and deployed in-house, reducing manual processing time by over **65%** (from 15 to 5 minutes per application)
- Increased speed and accuracy of financial data analytics by developing an **LLM**-powered pipeline to automatically extract and analyze financial ratios from complex statements, streamlining reporting for internal teams
- Developed an **Agentic** AI Caller Assistant with speech recognition and **LLMs**, automating dynamic, multi-turn client interactions; decreased manual call handling hours by **50%** and enabled real-time operational analytics on call data

National University of Science and Technology | *Full-Stack Developer* **Sept 2023 - Dec 2023**

- Designed and developed dynamic frontend features in React, such as research filters and notification panels, increasing successful student-professor connections by **20%** through improved discovery UX
- Integrated **RESTful APIs** with real-time polling mechanisms to display live research postings, reducing outdated listings and enhancing user navigation speed and accuracy
- Boosted user engagement by **30%** by implementing fully responsive profile pages using **CSS** media queries, adaptive grids, and image scaling for cross-device compatibility

Tetra Pak Ltd. | *Technical Team Intern* **May 2023 - Aug 2023**

- Achieved incident resolutions in under **30 minutes** by diagnosing sensor, motor, and control system issues on over **50** packaging machines, reducing repair times and improving uptime
- Reduced unscheduled downtime by **15%** by implementing comprehensive preventive maintenance and calibration programs for over **50** packaging machines, enhancing overall production efficiency
- Increased overall performance by **10%** by collaborating with cross-functional teams to implement targeted efficiency upgrades, boosting production throughput

Projects

Desktop Spotify Assistant | *ESP32* | *C++* | *Python* | *ArduinoIDE* | *RESTful API* | *SPI/I2C*

- Built an **ESP32**-based IoT device in C++ for real-time Spotify metadata visualization (e.g., song title, artist) using **RESTful APIs** and **SPI/I2C** for TFT screen integration
- Implemented direct music control (play/pause) functionality without accessing the Spotify app by developing **HTTP** request handlers for rotary encoder inputs
- Designed a secure Spotify **OAuth 2.0** flow with automated token refresh and encrypted local token storage for uninterrupted API access

Custom Macropad PCB | *Raspberry Pi Pico* | *CircuitPython* | *Altium Designer* | *AutoCAD*

- Designed a 3x3 matrix **PCB** using **Altium** Designer and **KiCad** with short trace routing and ground plane isolation for **EMI** reduction and signal integrity
- Validated Macropad PCB design using SPICE simulations, ensuring optimal circuit functionality and signal integrity
- Boosted productivity by **40%** by scripting customizable key macros in **CircuitPython** for task automation and quick command access

Portfolio Website | *HTML* | *CSS* | *React.js* | *Javascript* | *Three.js* | *JSON*

- Developed a modular, scalable portfolio site using **React.js** and **CSS** modules with a component-driven architecture for easy maintenance
- Engineered reusable **React** components to dynamically render content from **JSON** sources, reducing code redundancy
- Achieved cross-browser 3D model rendering using Google's Model Viewer for glTF/GLB files, avoiding complex WebGL boilerplate