Noah Sun

(647) 700-5886 | nn2sun@uwaterloo.ca | legendaryark.github.io | github.com/LegendaryArk | /in/sunnoah

Education

University of Waterloo – Candidate for BASc. in Mechatronics Engineering — September 2025 – April 2030
Relevant Coursework: Digital Computation, Algorithms and Data Structures, Circuits

University of Toronto – DEEP Summer Academy — July 2024 – August 2024
Relevant Coursework: Self Driving Robots, Big Data and Algorithms

Technical Skills

Languages/Frameworks: C++, C, Java, Python, TypeScript, ROS2, Flutter, ReactJS, Vite, Matplotlib, SQLite, Pandas, TensorFlow
Hardware: NVIDIA Jetson Nano, Arduino, STM32
Developer Tools: Git, Docker, Firebase, Foxglove, Linux, Figma, MS Office

Experience

Co-founder and Lead Developer – Elapse | Source Code | *Dart, Flutter, TypeScript, Firebase* — June 2024 – August 2025
- Led 4 developers to create a VEX tournament app that boosted access to schedules, rankings, and match data by **80%**
- Designed adaptive match times and scouting forms, achieving **3500+ downloads** and **5.0 star App Store Rating**
- Integrated Firebase, Firestore, and Cloud Functions to sync data between team members, reducing scouting times by **30%**

V5RC Robotics Coach – Mi3L Schools | *C++, Fusion 360, CNC Machining* — July 2024 – August 2025
- Coached **2 world class teams**, guiding them to win **5+ awards** at both local and international events
- Taught odometry, vision systems, and modular libraries, boosting autonomous route scores by **20%** and win rate by **15%**

V5RC Team Captain and Programming Lead – Checkmate Robotics | *C++, Git, Fusion 360* — June 2019 – August 2025
- Led team of 6 in robot design and library structure winning the **Create (innovation) Award** at VEX Worlds 2024
- Engineered and deployed ArkLib, enabling the team to qualify for VEX Worlds in **5 consecutive years** and achieve the **Think (programming) Award** at an international event
- Designed and tuned PID and motion profiling algorithms to maximize drivetrain precision and autonomous reliability

FTC Team Software Lead – Bayview Secondary School | *Java, Solidworks* — July 2024 – June 2025
- Led 5 programmers to develop a library for precise 2-axis arm control and efficient mecanum drivetrain navigation, reducing task times by **15%** and winning the **Control (programming) Award**
- Trained 4 new programmers on control algorithms and drivetrain systems, driving effective team contributions

VAIC Team Embedded Systems Lead – Mi3L Schools | *C++, Git, Linux, Python* — May 2024 – June 2024
- Developed an API enabling the VEX Brain to process 3D spatial data from the Intel RealSense depth camera via the Jetson Nano for adaptive field navigation, winning the **Innovate Award** and **Skills World Champion** at VEX AI Worlds 2024
- Diagnosed and fixed critical communication issues, enabling **2 robots** to reliably navigate autonomously in **10+ matches**

Projects

Autonomous Robot Control System | Demo Video | Source Code | *C++, ROS2, Foxglove, Docker* — September 2025
- Fused LiDAR and odometry data into a **0.1 m resolution costmap** for obstacle detection and precise robot localization
- Developed path planning and following using A*, Pure Pursuit, and PID controllers, achieving **0.1 m accuracy** and **flawless obstacle avoidance** across **50+ simulation tests**

Personal Website | Website Link | Source Code | *Typescript, HTML, CSS, ReactJS, Vite, Figma* — August 2025 – Present
- Built a responsive site with Vite and React, featuring a modern design and seamless cross-platform navigation
- Leveraged GitHub Pages for CI/CD and open-source hosting, reaching **100% uptime** and cutting deployment time by **40%**

ArkLib | Demo Video | Source Code | *C++, Git* — June 2020 – July 2025
- Modularized odometry and control algorithms using OOP, enabling seamless integration of **5+ robot configurations**
- Enhanced motion algorithms for precision, achieving **1” and 1° accuracy** via improved PID tuning and settling conditions
- Designed reusable libraries and APIs, cutting code integration time by **30%** and standardizing code across multiple seasons

Boggle | Source Code | *Java, Git, Figma* — May 2024 – June 2024
- Led a team of 4 to develop a digital Boggle, meeting all deadlines by optimizing project timelines with Gantt charts
- Integrated backend modules (word validation, game logic, etc.) seamlessly with the frontend UI implemented with JavaFX
- Engineered an AI opponent using recursive searching algorithms to provide **4 difficulty levels** to enhance replayability