ETHAN MIKOLAYCIK

+1 (737) 239-6840 | mikolaycikethan@gmail.com | github.com/EthanMik | youtube.com/@ethanmik2587

Education

Texas A&M University — **College Station, TX**
Bachelor of Science, Computer Science — *Aug 2025 – Expected 2029*

Lake Travis High School — **Austin, TX**
High School Diploma — *Aug 2021 – May 2025*

- Published robotics tutorial series

Experience

Software Developer — **Sep 2025 – Present**
WHOOP VEX U Robotics Team — *College Station, TX*

- Designed and planned a web path-planning app with slider-based simulation, X/Y field mirroring, and one-click auton code export
- Building app in React Typescript with Tailwind CSS
- Planned integration with existing motion library so each planned path is stored in current repository

Projects

mikLib | C++, VEXcode

- Open-source library for VEXcode adding odometry, motion control, and an on-brain custom UI framework for real-time PID tuning and autonomous selection
- Implemented two-tracker IMU odometry, wall resetting, with turn, swing and drive to point, drive to pose, and follow path drive motions
- Implemented reusable PID class and motion chaining, which values can be individually passed into each drive motion; SD-card save/load for tuning constants and selected autonomous
- Documented with setup tutorials and Doxygen-style comments; actively maintained
- Used by 5+ high school robotics teams in competition

Wall Reset Simulator | Javascript, HTML, CSS

- Built a browser-based simulator to validate a wall-reset localization algorithm, using virtual lasers to estimate robot (x, y, ϑ) from distances to field walls
- Implemented configurable front/left/right/rear lasers with sensor offsets, live visibility toggles, and field switching
- Deployed on GitHub Pages with documentation

Scene Maker | Java AWT/SWING

- Built a customizable scene maker; scene is built with background and particles, with velocity, size and amount parameters; snowflake and rain particles are pre-built
- Snowflakes are procedurally generated, and can be selected to rotate along the x, y and z axes
- Placed 1st in the CS3 winter scene competition

Minigame Roulette | C# Unity

- Developed a Unity game that has a collection of randomized minigames, selected via a roulette wheel
- Implemented game selection logic and unified scoring across multiple minigame types
- Designed and programmed multiple interactive minigames with custom physics, movement, event handling and camera scripts
- Added UI for real-time score display and transitions that contain minigame controls

Skills

Programming Languages C++, Javascript, Typescript, Java, Python, C#
Technologies Git, GitHub, VEXcode, PROS, raylib, Unity, React, Java AWT/Swing, VS Code, Fusion 360
Interests Mountain biking, Lego, Origami