





# Jack Schedel

## Artificial Intelligence Embedded Systems

-  <https://schedel.io>
-  [jack@schedel.io](mailto:jack@schedel.io)
-  [github/jackschedel](https://github.com/jackschedel)
-  [in/schedel](https://in/schedel)

## Education

### Bachelor of Science - Exp. May 2024

Computer Science (College of Eng.)  
University of Florida  
3.41 GPA

### Master of Science - Exp. May 2025

Computer Science  
University of Florida  
Combined Degree Program

## Coursework

- Digital Logic and Computer Systems
- Software Engineering
- Microprocessor Applications
- Operating Systems
- Mathematics for Intelligent Systems\*
- Machine Learning\*
- Advanced Data Structures\*
- Advanced Systems Programming\*

*\* completed by May 2024*

## Skills

- Rust
- Go
- C / C++
- JavaScript / TypeScript
- Python / Jupyter Notebooks
- TensorFlow / Keras
- React / React Bootstrap
- Tailwind CSS
- STM32
- MSP430
- SQL / PHP

## Work Experience

### Strategy Lead, FSGP 2023 Champions, Solar Gators Spring 2022 - Present

- Developed real-time telemetry data analysis tools for our team's solar-powered racecar to run live regression calculations in the racetrack pit using TypeScript.
- Used the analysis tools to manage and determine team race strategy during the 2023 Formula Sun Grand Prix, in which our team won the overall competition.
- Created static race physics simulations using Python, used in conjunction with an optimization solver, to optimize energy efficiency and find the ideal driving strategy for the complex 11-turn track.

### Embedded Systems Intern, Oak Ridge National Laboratory Summer 2023

- Wrote C and C++ code deployed to MSP430 chips on custom PCBs for a localized tracking system designed for use with containers for fissile nuclear material.
- Used light propagation latency from ultra-wideband communications between container devices and anchor devices to find one-dimensional distance.
- Implemented multilateration to determine container location using calculated distance information between the container device and multiple anchor devices with known exact placements around a target area.

## Projects

### AutoCalibr, Generative Autoencoder for 3D Meshes Summer 2023 - Present

- Created and trained a custom variational autoencoder model using Keras to generate new meshes using principle component analysis in the latent space.
- Handled comprehensive preprocessing of raw mesh data including face tri conversion, volume/positioning normalization, and face subdivision padding.
- Generated thousands of different random variations of each object during normalization and padding to artificially expand the dataset and enhance the model's comprehension of the intrinsic properties of the mesh data.

### KoalaClient, OpenAI API Client Interface Summer 2023 - Present

- Developed an advanced interface for the OpenAI API using TypeScript and Tailwind CSS to better integrate LLMs into my personal workflow.
- Added OpenAI Whisper speech transcription, and a searchable prompt pallet to insert frequently used prompts from a custom user-defined library.
- Implemented tweakable model configuration overrides, unrestricted conversation history editing, and syncing across devices via Google Drive.
- Deployed the project to a website ([client.koaladev.io](https://client.koaladev.io)) and Electron desktop app.

### EndGame2, UCI Chess Engine Summer 2023

- Developed a UCI-compatible chess engine written in Rust that uses minimax tree-traversal over a variety of custom board analysis algorithms.
- Implemented alpha-beta pruning, position hashing, piece position mapping, and multithreading to improve search performance times.
- Still actively being developed and expanded upon, with plans to further expand board analysis algorithms.

### Save the City, Mobile Game to Teach Sustainable Living Fall 2023

- Built in 24 hours by a team of 4 for the University of Florida AI Days Hackathon.
- Created minigames with automatically adjusting granular difficulties determined by a 4-dimensional linear regression model trained on past user data using scikit-learn with a custom REST API written in PHP.