

William Goodall

Education

2019-2023 **Georgia Institute of Technology**, College of Computing
(est.) Bachelor of Science, major in Computer Science, 3rd-year, GPA 3.81.
Relevant coursework: CS 1332 (*Data Structures and Algorithms*), CS 2110 (*Systems Architecture*),
CS 3210 (*Operating Systems*), CS 3220 (*Processor Design*), CS 3251 (*Computer Networking*)

History

2019-2021 **Mesmer Eyes, Inc** Remote
Developer (2019-2021), Intern (2019)

Worked to automate mobile app user experience and accessibility testing. Spearheaded several projects:

Mesmer Gherkin: a compiler for English-language descriptions of app tests (*Rust*)

- Built a compiler for a test description language similar to English, empowering customers to precisely define app tests, across Android and iOS, without technical knowledge
- Unified many of Mesmer’s different internal representations of test logic into one common syntax tree
- Collaborated with the Customer Success Team, writing Gherkin code to rapidly solve client problems
- Employed coverage-guided fuzzing and extensive snapshot testing to find new bugs and stop regressions

The Mesmer Sidecar: instant accessibility audits for Android apps (*Rust, Android/Java*)

- Built system to locally audit Android apps for accessibility issues, attaching to target apps at runtime and evaluating a proprietary ruleset against data extracted from Android APIs and in-house heuristics
- Built system to attach to a running Android app, extract accessibility data, and run in-house heuristics
- Created WCAG 2.0 rules engine to find accessibility issues and explain what’s needed to fix them
- Designed marketing materials, website, and pitch video for release on Product Hunt (#3 App of the Day)

The Mesmer CLI: stable command-line interface to Mesmer tools (*Rust*)

- Created a tool to expose most Mesmer platform functionality to developers working from the command line.
- Handled frequent changes in unstable internal APIs to maintain a stable “evergreen” public scripting interface
- Implemented CI/CD and a comprehensive test suite, auto-importing user error reports into failing tests, and deploying fixes across Mac, Windows, and Linux auto-updaters in 20min
- Provided front-line support, dealing directly with five Fortune 500 clients and many other smaller businesses

2019 **Wolfram Research** Waltham, Massachusetts
Student, Wolfram Summer Camp

Researched both traditional and machine learning approaches to algorithmically determine the physical scale of satellite images. Final model achieved $r^2 = 0.73$ using feature extraction and a small dense neural net.

2017-2018 **Transparenssee Systems** New York, NY
Intern, Developer

Created a client-facing customization tool for web embeds. This project was deployed to more than 60 local newspapers across the US, allowing non-technical users to integrate real estate platform `enclosure.io` and agricultural classifieds site `agrisearch.com` with their own properties.

- Designed and built full-stack web applications (React/Redux frontend, Express.js backend)
- Created an automated CI/CD pipeline to build, test, and deploy the application to Kubernetes

Skills and Projects

Languages Rust, Java, C, C++, JavaScript/TypeScript, Go, Python, Wolfram Language, VHDL, Verilog
Frameworks and Tools Git, CI/CD (Jenkins, GH Actions), Docker, GraphQL, React (+Native), Next.JS, WebAssembly
Platforms Google Cloud, AWS, Android, Firebase, Kubernetes, Heroku, Wolfram Cloud

Motor Controller (2021) FPGA-based closed-loop proportional servo controller. Used feedback from a quadrature encoder to position an output shaft with sub-degree precision. *Designed in VHDL, coursework for ECE 2031, Digital Design Lab.*

Boolean (2020) Symbolic computer algebra system for rule-based simplification of Boolean expressions. Contains a rudimentary SAT solver. *Written in Rust, compiled to WebAssembly, runs in-browser at `boolean.w01.dev`.*

Cookie (2019) Dynamic cookbook. Based on the user’s cooking preferences, construct a set of recipes as dependency graphs, simplify them by merging similar steps, and sequence them in parallel as the user cooks the meal. *Android, iOS app implement with React Native and TypeScript. Released to Google Play: `bit.ly/2muVZYY`.*