Matthew Riggle

Madison, WI | mattriggle05@gmail.com | mriggle.dev
linkedin.com/in/matthew-riggle | github.com/mattriggle05

EDUCATION

University of Wisconsin-Madison College of Computing & Artificial Intelligence | Madison, WI **Expected May 2028**
B.S. in Computer Sciences & Data Science, Certificate in Mathematics | GPA: 3.91

- Honors: Dean's List for all 4 completed semesters (Fall 2024, Spring 2025, Fall 2025, Spring 2026)

EXPERIENCE

Morgridge Institute for Research | Web Development Internship | Madison, WI **Mar 2026 – Present**

- Built and maintained full-stack production web applications as part of a two-person web team for CHTC and the OSG Consortium, supporting 1,000+ researchers and 100+ institutions across the national OSPool.
- Utilized part of an $83,000,000 NSF grant to help build FabAID, a national AI data fabric, federating storage, caching, discovery, and workflows to support large-scale datasets for AI-driven research.

Epic Systems Corporation | Summer Internship | Verona, WI **May 2025 – Aug 2025**

- Optimized VBA employee data processing performance, replacing linear scans with binary searches and merging multiple data sources, cutting generation time for over 10,000 personalized emails from hours to minutes.
- Built and maintained workflows to generate FMLA and benefits correspondence, improving document accuracy, vastly reducing turnaround time, and combining three disparate systems into just one.

PROJECTS

OSG Topology Interface | Professional | Go, Kubernetes, PostgreSQL **Aug 2026 – Present**

- Utilized AI coding agents to migrate the OSG topology registry from a Python/Flask YAML system to a Go/PostgreSQL REST API with byte-for-byte compatible XML endpoints and round-trip YAML import/export.
- Built secure PostgreSQL change-approval workflows with OpenID Connect, RBAC, AES-256-GCM envelope encryption, atomic multi-entity updates, optimistic conflict detection, and immutable audit logging.
- Packaged the Go API and statically generated Next.js frontend into a single binary, automating multi-stage Docker builds through GitHub Actions to produce a minimal container without a separate frontend server.

CHTC User App (UI / API) | Professional | Python, FastAPI, GitHub Actions **May 2026 – Present**

- Built the web interface for CHTC's high-throughput computing platform, using Next.js, React, TypeScript, Material UI, and SWR to build interactive workflows for managing users, groups, projects, and resources.
- Developed asynchronous FastAPI backend services with PostgreSQL, SQLAlchemy, Pydantic validation, and Alembic migrations, implementing authentication and reusable query layers for filtering, sorting, and pagination.
- Created automated Dockerized CI testing infrastructure using GitHub Actions and pytest, running migration checks and black-box relational integrity tests against seeded, isolated databases on every push and pull request.

OSPool Maps | Professional | Next.js, TypeScript, AWS S3 **May 2026 – Present**

- Built a greenfield Next.js/TypeScript geospatial mapping of the OSG's research computing ecosystem, highlighting research at non-R1 universities and in EPSCoR states to make underrepresentation more visible.
- Created searchable project and institution maps with React, Mapbox GL, and Material UI, including interactive markers, filters, and usage statistics panels using data pulled through batched requests for static visualizations.
- Designed a CI/CD AWS S3 deployment pipeline for the statically generated site, enabling a scalable, serverless map application that could be embedded across other pages without a persistent backend.

Keyboard Layout Optimizer | Personal | C++, CUDA, SciPy **Mar 2026 – Present**

- Built a touchscreen measurement tool that calibrates screen coordinates to millimeters, captures resting finger positions and reach regions, and generates personalized inputs for keyboard geometry optimization.
- Developed a Python effort-modeling pipeline with NumPy, SciPy, and scikit-learn, combining DBSCAN outlier detection, radial basis interpolation, and Gaussian smoothing to generate finger effort fields.
- Implemented GPU-parallel simulated annealing in C++/CUDA to optimize physical key positions, minimizing modeled finger effort while enforcing non-overlap constraints across independent search chains.
- Built a corpus-driven key-layout optimizer that assigns characters and control keys to custom geometries, accounting for finger rolls, modifier chords, and layers across prose, programming, and LaTeX workloads.

Natal Chart Solver | Personal | Rust, WebAssembly, React **Jan 2026 – Present**

- Built a single-page web application using React and TypeScript that implements a custom-built Rust interface to perform alignment searches on 10+ planets over 100-year ranges in less than a second.
- Developed a Rust numerical search engine using VSOP87 ephemerides, progressive interval filtering, bisection root-finding, and retrograde-motion detection to refine matching time windows.
- Compiled the Rust source to WebAssembly to execute divide and conquer searches in parallel Web Workers, exchanging typed arrays to keep the React interface responsive during computation.
- Validated numerical correctness against NASA JPL Horizons reference events and randomized cases, with regression tests for angular wraparound, retrograde transitions, and interval boundaries ensuring reliability.

SKILLS

Languages: Python, Java, Go, C, C++, C#, Rust, TypeScript, JavaScript, HTML/CSS, Kotlin, Swift, Bash, Assembly, SQL, R, PHP
Frameworks & Libraries: React, Next.js, .NET, FastAPI, Django, SQLAlchemy, Playwright, NumPy, pandas, Elasticsearch
Platforms & Tools: Google Cloud, Docker, Kubernetes, Linux, Git, GitHub Actions, PostgreSQL, Gradle, CUDA, Claude Code