Jack Adamczyk
Columbus, GA
jackadamczykjs@gmail.com | (706) 325-6589
Portfolio: https://jackadamczyk.vercel.app
GitHub: https://github.com/Adamjackczyk
LinkedIn: https://www.linkedin.com/in/jack-adamczyk

SUMMARY

Frontend-focused developer shipping modern, performant UI with React, TypeScript, and Tailwind CSS. Strong creative-coding depth in Three.js and custom GLSL shaders, including GPU particle systems, procedural animation, and real-time visual effects. Comfortable building production-minded features with testing, documentation, and team collaboration.

TECHNICAL SKILLS

Frontend: React, TypeScript, Next.js, Tailwind CSS, Vite, Webpack
Graphics: Three.js, GLSL (vertex & fragment shaders), GPU particles, GLTF/Draco pipeline, Blender, procedural noise
Testing / Docs: Vitest, Storybook
Backend: Node.js, Express, Python (scripting)
Databases: PostgreSQL, MongoDB
Workflow: Git, GitHub Actions, CI/CD basics, Agile/Scrum

EXPERIENCE

Software Engineer Extern (Remote) – Electric City Aquarium & Reptile Den
TripleTen Externship | 2025

- Built an interactive React, TypeScript, and Tailwind kiosk application with i18n and flag-based language toggles for daily visitors.
- Documented UI components in Storybook and added unit tests with Vitest to improve maintainability and reliability.
- Collaborated in Agile sprints on a 5-person team, completing Jira tasks and submitting GitHub pull requests with peer reviews.

PROJECTS

Particle Morphing Shader (Three.js, GLSL, Blender)
Live demo: https://adamjackczyk.github.io/THREE-particles-morphing-shader/

- Built an interactive particle morphing system using custom GLSL shaders to blend between multiple 3D shapes entirely on the GPU.
- Created and exported custom morph-target models in Blender and loaded compressed assets via GLTFLoader with DRACOLoader.
- Traversed GLTF scenes to merge multi-mesh geometry into unified position buffers and normalized mismatched vertex counts using randomized sampling.
- Added simplex-noise-driven timing offsets, per-particle size attributes, additive blending, and noise-based color gradients for visual depth and motion control.

Galaxy Shader Generator (Three.js, GLSL)

- Implemented a real-time animated galaxy using custom vertex and fragment shaders, additive blending, and optimized particle attributes to maintain smooth performance at high particle counts.

WTWR Weather App (React, Node.js)

- Built a React and Node.js application that integrates live weather data to recommend weather-appropriate outfits, with browsing and item-saving functionality.

EDUCATION

Columbus State University – B.S. Computer Science (In Progress, Junior)
Track: Game Development / Software Development

TripleTen – Software Engineering Certificate (2025)
Full-Stack Web Development: React, Node.js, Express, MongoDB, SQL