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 building modern UI with React, TypeScript, Next.js, and Tailwind CSS. Strong creative-coding depth in Three.js + GLSL, including GPGPU simulation, procedural flow fields, and real-time shader effects. I build visually impressive projects that remain production-minded: versioned, deployed, and structured for iteration.

TECHNICAL SKILLS

Frontend: React, TypeScript, Next.js, Tailwind CSS, Vite, Webpack
Graphics: Three.js, GLSL (vertex/fragment/compute), GPUComputationRenderer, GPU particles, flow fields, GLTF/Draco, Blender, procedural noise
Testing/Docs: Vitest, Storybook | **Backend:** Node.js, Express, Python | **DB:** MongoDB
Workflow: Git, GitHub Actions, CI/CD basics, Agile/Scrum

EXPERIENCE

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

- Built a visitor-facing kiosk app by developing reusable React + TypeScript UI and implementing i18n language toggles.
- Improved maintainability by documenting components in Storybook and writing unit tests with Vitest.
- Delivered features in a 5-person Agile team by completing Jira tickets and submitting PRs with peer review.

PROJECTS

GPGPU Flow Field Particles (Three.js, GLSL, GPUComputationRenderer)
Live: https://adamjackczyk.github.io/THREE-gpgpu-flow-field-particles-shaders/

- Built a GPU particle simulation to animate thousands of particles by updating positions in a compute shader instead of CPU loops.
- Preserved model identity by extracting Draco-loaded GLB vertex positions and vertex colors into GPU textures and attributes.

- Achieved organic motion by generating a 3D simplex-noise flow field with tunable influence/strength/frequency via lil-gui.

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

- Built GPU-driven particle morphing by blending between multiple target geometries directly in GLSL.
- Normalized mismatched meshes by merging multi-mesh GLTF scenes and resampling vertex counts for consistent transitions.
- Increased visual clarity using per-particle attributes, noise-driven timing offsets, and additive blending.

Realistic Earth (Three.js, GLSL)
Live: https://adamjackczyk.github.io/THREEjs-earth-shaders/

- Built a physically-inspired planet render by blending day/night textures and limiting specular highlights to oceans.
- Improved realism using cloud masking and atmospheric rim glow with Fresnel-based shading.

Particles Cursor Animation Shader (Three.js, GLSL)
Live: https://adamjackczyk.github.io/THREEjs-particles-cursor-animation-shader/

- Built a cursor-reactive image-to-particles effect using a fading 2D canvas as a live displacement texture.
- Improved interaction feel by mapping cursor UVs via raycasting and scaling displacement intensity by cursor speed.

EDUCATION

Columbus State University – B.S. Computer Science (In Progress, Junior)
TripleTen – Software Engineering Certificate (2025)