

## Work Experience

---

### Amazon Web Services, Software Development Engineer

*April 2022 - August 2024*

- Develop Tying, a Java-based engine for automated patching and OS migration of Amazon hosts.
- Design and author integration and end-to-end load tests for the Tying engine.
- Improve dev-ops tooling and monitor service metrics proactively and in on-call capacity.
- Expand and maintain CI/CD and distributed cloud computing infrastructure-as-code.

### Epic Systems, Software Developer

*August 2019 - August 2021*

- Develop and design the App Orchard website, a marketplace for third-party healthcare apps.
- Implement RESTful web APIs to expose health record data according to the HL7 FHIR standard.
- Lead transition of App Orchard site from ASP.Net framework to single-page React app.

## Education

---

### University of Pennsylvania, 2025 (expected)

- M.S. in Computer Graphics and Game Technology

### Rice University, 2019

- B.S. in Computational Physics; Minor in Computational and Applied Mathematics.

## Projects

---

### Personal Blog Site, <https://mattzschwartz.web.app/>

- A personal website, portfolio, resume, and blog built on the React framework.
- **Technologies:** React, Recoil, Typescript, Firebase, HTML, CSS, Material UI.

### Fluid Surface Mesh Shader, <https://github.com/mzschwartz5/Breakpoint>

- A compute and mesh shading pipeline for triangulating fluid surfaces from point cloud data.
- **Technologies:** C++, DirectX 12, HLSL

### CUDA Pathtracer, <https://github.com/mzschwartz5/CUDA-Path-Tracer>

- A pathtracer built on CUDA with support for various PBR materials, GLTF model loading, denoising, and various performance optimizations.
- **Technologies:** CUDA, C++, OpenGL

### Node Based SDF Modeling Tool, <https://github.com/mzschwartz5/Node-Based-SDF-Modeling-Tool>

- A node-graph tool for quickly prototyping procedural 3D models, via raymarched SDFs.
- **Technologies:** C++, OpenGL, GLSL, ImGui

### Mini Minecraft, <https://github.com/mzschwartz5/Mini-minecraft>

- A Minecraft recreation with support for multithreaded terrain (de)loading, dynamic shadow mapping, and much more.
- **Technologies:** C++, OpenGL, GLSL, QT