

Tristan Shabbick

tshabbic.com | New Jersey | tshabbic@gmail.com

EDUCATION

Stevens Institute of Technology

Hoboken, New Jersey

Master of Science in Computer Science | GPA: 3.95/4.0

January 2026 – Expected December 2026

Bachelor of Science in Computer Science

September 2022 – December 2025

Specialized Courses: Modern Parallel and Distributed Computing on Cluster and Supercomputers, Parallel Programming for Many Core Processors, Concurrent Programming, Advanced Programming in the Unix Environment, Deep Learning.

SKILLS

Programming Languages: ARMv8 Assembly, C, Promela, CUDA, C++, C#, Java, Groovy, R, Python, JavaScript, Bash / Shell, Scheme, OCaml, Erlang, SQL

Technologies & Tooling: VirtualBox, KVM/QEMU, Docker, ssh(1), git(1) & GitHub, make(1), Gradle, gdb(1), Valgrind, jSpin, dotMemory, dotTrace, OpenCL, OpenMP, Thrust | LaTeX, HTML, CSS, and vi(1)

Operating Systems & Cloud: Linux, NetBSD, Windows 10 & 11, Amazon Web Services (AWS; EC2, S3)

Protocols & Standards: TCP/IP, UDP, HTTP, DNS, SMTP

Management & Communication: Agile Methodology, CI/CD, Jira, Backlog, Notion, Linear | Microsoft Teams, Slack

Currently Learning: Keras, NumPy, PyTorch, TensorFlow, MPI, GPU-aware MPI (via NCCL), Slurm batch scripting

PROJECTS | *See tshabbic.com/projects for additional, detailed projects & source code*

Simple Projectile Simulator — GPU-accelerated particle simulator using **OpenGL**, **OpenMP**, and **CUDA**, to explore memory-hierarchy optimization, spatial binning, and latency hiding, achieving an overall ~86x performance increase.

Simple Web Server — HTTP/1.0-compliant web server with CGI support, written in **C** to strict RFC design specifications and for portability, and debugged across Linux, NetBSD, and OmniOS.

Senior Design Project — Led a 6-person team through the full software development life cycle — planning, architecture, delivery — of a **Java/Gradle** Minecraft–Discord sync plugin; mentored peer teams, resolved production issues remotely.

Multi-Server Chat / Nexus — Low-level, multiplexed I/O message queue bridging game servers in **C** and **C#/.NET**; performance-optimized to a sub-0.5MB memory footprint sustained over 230+ days of uptime monitoring.

Discord Rank Up Bot — Discord bot written in **Python**, providing a server-wide leaderboard, scoring system, and dynamically generated rank-card graphics.

Personal Website (tshabbic.com) — Personal site with a Unix man-page-inspired design, handling backend routing in **Node.js/Express** and server-side templating with **Pug/Jade**.

EXPERIENCE

RenCorner Network

Pittsburgh, Pennsylvania

Lead Game Server Developer for a community-run multiplayer game server network

March 2023 – Present

- Led design and development of a large-scale raid boss system spanning numerous backend systems, optimizing per-connection network performance by at least 60x and reworking ally-projectile handling for further gains.
- Perform the core functions of a software engineer for the network, designing, developing, and debugging backend systems and dozens of custom plugins primarily in C#, as well as Python, C, and Node.js.
- Built a secure, API-based VPN detection system and managed server infrastructure across multiple cloud hosts, making it easier for staff to identify and act on cheaters and other bad actors.

Re-Logic

Floyd Knobs, Indiana

*Beta Tester & Quality Assurance for **Terraria** (Reference from CEO available upon request)*

May 2020 – Present

- Collaborate with creative leads and programmers over Slack across 6+ years of pre-release cycles, actively listening to design intent and communicating structured feedback, to refine features and ensure high-quality releases.
- Identify and document hundreds of bugs and game imbalances across pre-release and live builds using Jira and Backlog, communicating findings with developers to improve gameplay mechanics and player experience.

Stevens Institute of Technology

Hoboken, New Jersey

*Course Assistant for **Computer Architecture & Organization, Systems Programming***

August 2024 – May 2025

- Developed automated grading scripts in bash, applying consistent, criteria-based checks across every submission and cutting grading time per assignment set from several hours to just a couple minutes.
- Explained compilation, linking, system call, file system, and virtual memory space concepts on Linux to students, breaking material into clear, structured, step-by-step guidance, improving their comprehension and course outcomes.