

# Vanessa Ching

U.S. Citizen · 📩 vanessazqching@gmail.com · 📞 (678) 779-0547 · 💬 Vanessa Ching

## Education

---

### Johns Hopkins University, Whiting School of Engineering

B.S. in Computer Science, Minor in Film and Media Studies

Baltimore, MD

Aug 2024 – May 2028

**Relevant Coursework:** Computer Systems Fundamentals, Intermediate Programming, Gateway Computing: Python

**Other Activities:** Matriculate Advising Fellow, Pi Beta Phi Fraternity for Women

## Experience

---

### Social Cognitive AI Lab, Johns Hopkins University

Baltimore, MD

*Research Assistant*

Aug 2025 – Present

- Used Python scripting and Jupyter notebooks with SimWorld (Unreal Engine 5) to generate large-scale city environments for embodied AI navigation experiments
- Constructed urban simulation scenarios with dynamic human obstacles, environmental variation, and audio cues
- Supported embodied AI research using Unreal Engine workflows, simulation assets, and multimodal experiment design

### Frenalytics

Remote

*Software Engineering Intern*

May 2025 – Aug 2025

- Built an AI-powered reading assistant WordPress plugin for 1500+ children with literacy challenges
- Designed backend REST APIs in PHP for avatar customization, session management, and AI interaction storage
- Modeled and managed MySQL database to enable personalization, analytics, and educator feedback; integrated frontend JavaScript with WordPress backend for CRUD operations and smooth UX

## Projects

---

### AI Readezzy Reading Assistant | PHP, SQL, JavaScript, CSS

- Architected a full-stack WordPress plugin with REST APIs and persistent session management to support AI-assisted reading
- Designed relational MySQL schemas to track user state, avatar configuration, and AI interaction logs
- Implemented dynamic avatar customization and session tracking for personalized learning experiences

### Multi-Client Chat Server | C++

- Implemented a multithreaded TCP/IP chat server with room-based publish/subscribe messaging
- Ensured thread-safe shared state using mutexes, semaphores, RAI guards, and per-client message queues

### Parallel Quicksort | C++

- Implemented a fork-join parallel quicksort using POSIX processes and shared memory (mmap)
- Evaluated execution time across varying parallelism thresholds and analyzed scalability tradeoffs

## Leadership & Community Involvement

---

### Hopkins Student Organization for Programming (HOP)

Baltimore, MD

Fine Arts Committee Member

- Plans and organizes campus-wide fine arts events for 5600+, including catering and supplies

### Student Government Association (SGA) Programming Council

Baltimore, MD

Sophomore Representative

- Executes large-scale initiatives for a class cohort of 1,000+ students, coordinating logistics and outreach

## Skills

---

**Programming Languages:** Python, C++, C, Java, JavaScript/TypeScript, SQL, PHP

**Systems & Concurrency:** Multithreading, TCP/IP, POSIX APIs, Shared Memory, Synchronization

**Web & Backend:** React, HTML/CSS/Tailwind, WordPress, REST APIs, MySQL, GitHub Pages, Vite

**AI & Simulation:** PyTorch, TensorFlow, Unreal Engine 5

**Tools:** Git, GitHub, Linux, VS Code, Jupyter Notebooks, XAMPP