

# Joshua Liu

484-758-7935 | [liu.joshualiu555@gmail.com](mailto:liu.joshualiu555@gmail.com) | [LinkedIn](#) | [Portfolio](#)

## EDUCATION

### Duke University

Durham, NC

*BS in Computer Science, Certificate in Entrepreneurship*

*Aug 2024 – May 2028*

- Coursework: Data Structures and Algorithms, Java OOP, Linear Algebra, Calculus II, Robotics

## EXPERIENCE

### Software Engineer

Nov 2024 – Present

*MyGymBuddy*

*Remote*

- Created a robust messaging system with React Native and Pocketbase, enabling seamless in-app communication
- Implemented live push notifications using Expo's notification API, providing users with real-time distance updates
- Enabled cross-platform map integration (Google / Apple), reducing manual address input to one-tap navigation
- Integrated location search with Google Places API autocomplete, supporting fuzzy matching for gym discovery
- Debugged 8 issues across route navigation, API calls, and SQLite queries, improving application stability

### Software Engineer Intern

Mar 2024 – Jun 2024

*Trio Labs*

*Morrisville, NC*

- Enhanced 9 ERP pages by adding tooltips, tables, and auto-generated notes using Jinja and DataTable
- Strengthened validation on 3 Flask forms using Python, HTMX, and SQLAlchemy / PostgreSQL database migrations, preventing invalid data entry and minimizing future cleanup time
- Wrote 65+ PyTest cases for page access, form input, PDF exports, and slicer bugs, improving code coverage
- Optimized pricing model visualization through lazy loading and correlation analysis using Pandas / Numpy, achieving 4x faster chart rendering for monthly shipping data insights
- Built a Rust tool with an egui interface to configure PVT profile editing with cubic spline interpolation, enabling precise control over pump speed and coating thickness while eliminating error-prone manual JSON edits

## PROJECTS

### Real Life “Among Us” Game

- Built a MERN webapp to automate Duke's “BioSci Assassins” game
- Stored global player and game state using Zustand to reduce passing data across the component tree
- Handled session management by storing and retrieving player IDs in Redis via cookies sent by the client
- Implemented real-time multiplayer with Socket.io to synchronize game state across all players
- Deployed the monorepo to Heroku, ensuring for continuous integration with every new commit

### Competitive Programming Stress Tester

- Developed a tool to compare solution algorithms and brute force algorithms against generated test cases
- Engineered a multithreaded process to compile, execute, and compare the outputs without disrupting the UI
- Utilized file I/O and the JavaCompiler API to dynamically execute user-inputted code
- Containerized CLI tool with Docker and bash script for consistent, portable execution across environments

### Pixel Art Maker

- Engineered a pixel art maker using Pygame and OOP principles with features such as drawing, color selection, pen sizes, eraser, trash, shape creation, and screenshots
- Implemented the bucket fill feature using DFS and the undo / redo feature using a stack
- Converted the program into an executable using PyInstaller for easy distribution without any setup

## SKILLS

**Languages:** JavaScript / TypeScript, Python, Rust, C++, Java, HTML / HTMX, CSS

**Databases:** MongoDB, PostgreSQL, SQLAlchemy, Pocketbase

**Frameworks and Libraries:** React, React Native, Node.js, Express.js, Flask, Socket.io, Redis, Zustand, Jinja, Pandas, Numpy, egui, egui plot

**Tools, Cloud, and Platforms:** Git, GitHub, PyTest, Docker, AWS EC2