

JACOB FYFFE

Kimper, KY • (606) 689-7111 • jacobfyffe16@gmail.com

github.com/jacobfyffe • linkedin.com/in/jacob-fyffe

SUMMARY

Computer Science student (B.S., expected December 2026, 4.0 GPA) with hands-on experience building, debugging, and testing software. Seeking a remote role in software development, QA, or technical support where I can apply Python, SQL, and debugging skills on a real product. Detail-oriented and reliable, with a proven record of resolving problems and improving processes under pressure.

TECHNICAL SKILLS

Languages: Python, Java, C++, C#, PHP, SQL, JavaScript, HTML/CSS, TypeScript

Concepts: Object-Oriented Programming, Data Structures, Algorithms, Debugging, Databases, Networking, Operating Systems, REST APIs, Client-Server Architecture, Authentication (OAuth), Input Validation & Sanitization

Currently learning: Unit Testing, CI/CD, Containerization (Docker)

Tools: Git/GitHub, Node.js, Supabase/PostgreSQL, Microsoft Excel & Access, Office 365, Google Workspace

Currently learning: React, Laravel

PROJECTS

ACME Customer Registration System — A full-stack form application with dual-layer validation that captures, validates, and persists customer data to CSV. HTML/CSS/JavaScript front end, PHP back end. [GitHub link *coming soon*]

- Built a customer registration form with client-side validation in vanilla JavaScript and a PHP backend that independently re-validates all 14 fields server-side, treating client validation as UX-only and never trusting incoming data.
- Hardened the backend against malformed and malicious input using `strip_tags` sanitization, allowlist validation for all choice fields (gender, contact method, interests), integer range-checking, and RFC-compliant CSV escaping to prevent injection into the output file.
- Designed an accessible error-handling flow: an ARIA role="alert" banner that aggregates all validation errors and auto-scrolls into view, plus graceful server-side fallback (HTTP 422) when JavaScript is disabled.

Big Brother Simulator — A browser-based reality-competition simulator that runs a full multi-week season with an autonomous social-strategy engine and cloud-synced season history. Vanilla JavaScript, HTML/CSS, Supabase (Postgres + Auth). [GitHub link *coming soon*]

- Built a ~4,900-line simulation engine in dependency-free JavaScript that models an entire season (HOH, nominations, veto, eviction, final 3, and jury vote) where every AI decision is driven by a per-houseguest relationship and grudge model rather than random selection; nominations and votes are computed from weighted "threat" and "danger" scores combining relationship level, accumulated grudges, competition wins, and alliance protection.
- Implemented a dynamic alliance system (formation gated on mutual-relationship cohesion thresholds, with fractures and betrayals), a social-encounter engine that shifts relationships over time, and 14 historically-accurate twists (Double Eviction, Battle of the Block, AI Arena, Pandora's Box, etc.) with configurable one-shot vs. multi-week persistence.
- Integrated Supabase for GitHub OAuth and Postgres persistence, archiving each completed season (cast stats, full week-by-week history, alliances, twists, and a settings snapshot) to power a stats dashboard with cross-season player records, a step-through season replay, and an owner-gated admin panel.

EXPERIENCE

Server / Bartender / Server Trainer Texas Roadhouse — Pikeville, KY

May 2021 - Present

- Promoted to Server Trainer (an extension of management); designed and implemented a standardized menu and service-skills study guide now used to train all new hires.
- Handle high-volume order accuracy and resolve customer issues in real time under pressure; directly transferable to customer-facing technical support.
- Recognized as Employee of the Month and Trainer of the Quarter (multiple times) for reliability and process improvement.

EDUCATION

B.S. Computer Science University of the Cumberlands (online)

Expected December 2026 • 4.0 GPA • *President's List (multiple semesters)*

Relevant coursework: Data Structures, Algorithms, Databases, Object-Oriented Programming, Operating Systems, Networking, etc.