

# KAREEM ELSABROUTY

## Portfolio & Contact

### PROFESSIONAL SUMMARY

---

I build systems that connect hardware, databases, and user interfaces - often delivering solutions end-to-end. At Coinadrink, I joined as a new hire and rebuilt the company frontend consisting of dozens of pages providing critical data and services, developed a contactless payment and access control platform in Rust/React/Python that runs unattended on Raspberry Pi devices with a touchscreen, implemented a C++/Arduino weight-tracking system for smart fridges, and created business intelligence dashboards that replaced static reports with live operational data.

I focus on choosing the right tools for the problem and have shipped production systems in Rust, TypeScript, Next.js, PHP, and JavaScript within the same year. I learn by building and aim to create systems that last through clean architecture, continuous support and updates, failure recovery, and attention to performance when resources are constrained.

### TECHNICAL SKILLS

---

<b>Languages</b>	JavaScript, TypeScript, Rust, PHP, SQL, C
<b>Frameworks &amp; Libraries</b>	React, Next.js (learning), React Native, Tauri, Axum, Tokio, Flask
<b>Hardware &amp; Embedded</b>	Raspberry Pi, Arduino, MDB Serial Protocol, USB Serial Communication, HX711 Load Cells
<b>Databases &amp; Tools</b>	MySQL, SQLite, MongoDB, DB2, Docker, Git, Linux

### WORK EXPERIENCE

---

**Software Developer** | Coinadrink Ltd

*August 2025 - Present*

Joined as a new hire straight out of university and delivered five production systems end-to-end, from frontend interfaces to embedded hardware.

#### *Key Systems Delivered*

- **Contactless POS & Access Control System** - Full-stack payment platform (React, Tauri/Rust, Python Flask, SQLite) running on Raspberry Pi. Integrates via MDB serial protocol with PicoVend hardware to handle contactless card payments, unlock doors upon successful transactions, manage real-time inventory, and control automated dispensing. Features OTA updates, 70MB RAM footprint, and unattended 24/7 operation with automated failure recovery.
- **Cartesian Weight & Coordinating System** - C++/Arduino smart weighing system with HX711 load cell amplifier for real-time weight and positional tracking. Built bidirectional USB serial communication protocol between Arduino hardware and React Native mobile app, enabling instant billing based on products removed from specific shelf positions - eliminating manual checkout entirely.
- **Legacy System Overhaul** - Single-handedly rebuilt the company's entire undocumented PHP frontend in React, implementing authentication, a reusable component library, and optimised backend APIs/SQL queries. Reduced page load times from 30+ seconds for some complex data pages to under 4 seconds. Documented every component, helper, and enforced clean code practices for easy team adoption, replacing an unmaintained legacy codebase with a clean, well-documented system.
- **Customer Service Management System** - Task assignment and customer tracking platform with Teams integration, automated alerts, image storage, calendar sync, and live status updates
- **Food Routes Scheduling System** - Drag-and-drop route planner using Google Matrix API for optimal delivery pathfinding with visual scheduling interface

### FEATURED PERSONAL PROJECTS

---

**Editor Lite** - Rust, JavaScript, Tauri

Lightweight code editor built with Rust, React and Tauri as a VS Code alternative for day-to-day editing. Designed to run on minimal hardware (3.3MB RAM idle). Features clean, focused UI with keyboard-first interactions for file opening, terminal toggles, and navigation. End-to-end built and optimised with modular codebase focused on performance and ergonomics. Available for download on Windows.

**Auction Platform** - Rust, Axum, Tokio, React, Docker

Production-ready auction platform with Rust/Axum backend and React frontend, deployed on Render and Vercel. Implemented real-time bidding, session-based authentication, admin controls, and automated auction lifecycle management. Features clean architecture with RESTful API design, database normalisation across five relational tables, and containerised deployment via Docker.

**Jarvis Jr** - Python, JavaScript, Ollama, FastAPI

Local frontend UI communicating with backend API to interact with locally hosted LLM. Built with React, Ollama for the LLM, and FastAPI for the backend. Features chat history persistence via local storage, Redux state management, and real-time streamed responses.

**C Backend** - C, SQLite, JavaScript

Backend HTTP server built from scratch in C to learn how servers, memory management, and databases work at a low level. Implemented request handling, JSON responses, SQLite database integration, and frontend connectivity.

## **EDUCATION**

---

**BSc Computer Science (First Class Honours)** 2024 - 2025

Staffordshire University

**Diploma in Computing (Merit)** 2021 - 2024

Walsall College

**Software Development Bootcamp Certificate** 2025

Just IT

## **PREVIOUS EXPERIENCE**

---

Incident Manager | RAC March 2025 - May 2025

Team Member | Asda October 2024 - January 2025

Data Entry | Lidl Warehouse September 2022 - January 2023

Call Centre Agent | Vodafone January 2019 - September 2020