JARED MAHOTIERE

Jared Mahotiere | Bear, DE | (302) 803-7673 | jmahotie@purdue.edu | linkedin.com/in/jared-mahotiere | github.com/jmahotiedu

SUMMARY

Backend and systems software engineer focused on distributed services, C systems programming, .NET backend development, and SQL performance optimization.

EDUCATION

Purdue University - B.S. Electrical Engineering Technology (Computer Engineering Technology)

Minor: Computer & IT | Certificate: Entrepreneurship & Innovation | Expected May 2026

SKILLS

C, C#, TypeScript, Python, Java | PostgreSQL, SQL Server, Redis Streams | .NET, ASP.NET Core, Node.js, Express, React | Docker, GitHub Actions, Prometheus, Grafana, AWS | REST, gRPC, concurrency, POSIX networking

EXPERIENCE

Nucor Corporation - Software/Automation Engineering Intern | Darlington, SC | May-Aug 2024 and May-Aug 2025

- Developed and maintained Blazor/.NET real-time operator dashboards and robust back-end services, enhancing process transparency and improving steel production workflows.
- Managed and analyzed production data in SQL Server/QMOS databases; developed optimized queries and recommended new tables/columns to support process improvement.
- Built automated reporting and alert systems using Quartz.NET with real-time email notifications for maintenance and quality events, reducing manual monitoring and accelerating response.
- Migrated legacy Visual Basic applications to .NET/Blazor, reducing technical debt; utilized Git for version control, peer code reviews, and codebase integrity.
- Collaborated with production teams and led project meetings; conducted comprehensive testing and validation with multi-disciplinary stakeholders while prioritizing deliverables and shipping on time with high safety and quality standards.

PROJECTS

workflow-orchestrator - TypeScript, Node.js, Redis Streams, Postgres

- Executed 25/25 benchmark runs in 15.94s (1.57 runs/s) with DAG validation, consumer groups, idempotency, and run-state durability.

cachekit - C (C11), POSIX, RESP

- Built networked in-memory caching with RESP protocol support, low-level socket handling, and persistence primitives.

Telemetry Node - ESP32, FreeRTOS, C

- Delivered embedded telemetry firmware to demonstrate low-level debugging, protocol design, and systems reliability skills.