JARED MAHOTIERE

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

SUMMARY

.NET and industrial software developer with production Blazor experience, SQL Server optimization, automated reporting, and real-time dashboard delivery in steel manufacturing environments.

EDUCATION

Purdue University - B.S. Electrical Engineering Technology (Computer Engineering Technology)
Minor: Computer & IT | Certificate: Entrepreneurship & Innovation | Expected May 2026
Relevant Coursework: Industrial Controls, DAQ, Systems Development, IT Architecture, Network Engineering

SKILLS

C#, .NET 8, ASP.NET Core, Blazor, Entity Framework | SQL Server, T-SQL, PostgreSQL, Redis | xUnit, CI/CD, GitHub Actions | TypeScript, React, Next.js | Docker, AWS

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

Feature Flag Platform - TypeScript, Node.js, Redis, React

- Built multi-tenant control-plane patterns including deterministic rollout logic, RBAC, idempotency, and operational observability.

workflow-orchestrator - TypeScript, Redis Streams, Postgres

- Implemented queue-driven workflow execution with durable state, retries, and distributed worker coordination.

syncboard - Next.js, Socket.IO, PostgreSQL

- Built real-time full-stack collaboration features with optimistic UI updates, presence tracking, and conflict handling.