

# JARED MAHOTIERE

Jared Mahotiere | Bear, DE | (302) 803-7673 | [jmahotie@purdue.edu](mailto:jmahotie@purdue.edu) | [linkedin.com/in/jared-mahotiere](https://www.linkedin.com/in/jared-mahotiere) | [github.com/jmahotiedu](https://github.com/jmahotiedu) | [jmahotiedu.github.io/jmahotiedu](https://jmahotiedu.github.io/jmahotiedu)

## SUMMARY

.NET and industrial software developer with production Blazor experience, SQL Server optimization, automated reporting, real-time dashboard delivery, AWS ECS platform deployments, and open-source backend contributions.

## 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

## LEADERSHIP & ORGANIZATIONS

**Delta Tau Delta (Campus Chapter):** DEI Chair | **National Society of Black Engineers (NSBE):** Member

## 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 (ECS Fargate, ALB, RDS, ElastiCache)

## EXPERIENCE

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

- Led system integration projects: scoped, specified, and coordinated implementation of new automation systems, ensuring seamless startup, cross-team adoption, and operational reliability.
- Developed and maintained Blazor/.NET real-time operator dashboards and robust back-end services, enhancing process transparency and improving steel production workflows.
- 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.

## PROJECTS

**Event Stream Platform - C#, .NET, WebSocket, WAL, Prometheus**

- Built a durable event ingest/replay platform with WAL-backed persistence, materialized views, deterministic backfill tooling, and production observability.

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

- Built multi-tenant control-plane patterns including deterministic rollout logic, RBAC, idempotency, operational observability, and a reproducible load-test harness; deployed on AWS ECS Fargate with ALB, RDS Postgres, and ElastiCache Redis.

**workflow-orchestrator - TypeScript, Redis Streams, Postgres**

- Implemented queue-driven orchestration with DAG validation, durable run/task state, worker retries, and dead-letter handling for reliability.

## OPEN SOURCE CONTRIBUTIONS

- PicoClaw (Go): 3 merged PRs on a 14.5k-star project, including provider protocol-family refactor (#213) and security/model hardening follow-ups; invited to Dev Group.
- PRs Under Review: Databricks CLI (#4504) auth-resolution fix; Google langextract (#359) cache-key hashing fix.