John Judge
(845)-825-1771 | judgejohn17@gmail.com | https://github.com/johnjjudge | Shoreline, WA

Education **Rochester Institute of Technology** (Graduation May 2019)
Bachelor of Science in Computer Engineering

Certifications Microsoft Certified: Azure AI Fundamentals (January 2026)

Experience **Software Engineer II, Microsoft, Redmond, WA** (Mar. 2022 – Present)
Hardware Dev Center (Mar. 2022 – Present)

- Designed and delivered features across driver certification, signing, and publishing workflows that support the Windows driver ecosystem used by ~1.4B active Windows devices, improving stability, usability, and security for hardware partners. Worked on several distributed event-driven services.
- Co-designed and shipped an Azure event-driven, domain-driven service to re-scan Windows Update–published drivers; processed 250,000+ drivers and helped surface hundreds of suspicious drivers for further security investigation.
- Improved API performance by ~35× by designing and implementing a reusable filtering algorithm used across authorization/provider paths, reducing latency for a high-traffic service.
- Operational excellence at scale: led monitoring/SLO work and reduced on-call burden by tuning noisy alerts, diagnosing and mitigating production issues, and documenting playbooks to shorten time-to-mitigation and prevent repeat incidents.
- Architected and delivered an Azure-hosted synthetic E2E availability testing solution adopted across 5+ production services, validating critical user workflows and downstream dependencies to catch regressions before customers do.
- Drove team execution and engineering standards: served as Scrum Master for a sister team; authored engineering excellence standards adopted by the team; converted peer feedback into team action plans; organized hackathon teams; led team/org-level knowledge sharing via technical workshops and presentations; mentored/onboarded 8+ engineers.

Software Engineer, Microsoft, Redmond, WA (Jun. 2019 – Mar. 2022)
Hardware Dev Center (Jul. 2020– Mar. 2022)

- Planned and executed zero-downtime OS upgrades across the team's VM fleet in all environments (including prod); balanced dev + cloud cost tradeoffs, validated unknowns via spikes, and completed rollout within one sprint without service interruption.
- Partnered with engineering, security, and legal stakeholders to close a critical EV certificate validation gap; gathered requirements and implemented the validation algorithm to harden identity verification in the driver signing workflow.

Services Hub (Jun. 2019 – Jul. 2020)

- Designed and shipped a new Azure-hosted REST API microservice handling tens of thousands of requests/day, with production telemetry and operational readiness.
- Independently delivered a customer-facing workflow improvement adopted by thousands of enterprise customers, saving tens of hours for customers and Technical Account Managers through streamlined case/engagement handling.
- Owned infrastructure automation for a new service: built ARM-based IaC and CI/CD pipelines enabling consistent, automated, test-gated deployments from initial build through MVP.

Projects **Fantasy Football League Analyzer** (June 2025 - Present)
Built Python tooling to export ESPN Fantasy Football data to CSV and generate lightweight analyses (draft value, start/sit efficiency) for offline evaluation; available on GitHub.

Lead Embedded Control Systems Engineer, IREC Hybrid Rocket Engine (August 2018 – May 2019)
Lead control systems engineer on a multidisciplinary team to create a tested, functional, and reusable hybrid rocket engine for use in the 2020 IREC. Responsibilities included the design and building of RTM monitoring systems. Skills include high-frequency sensor reading, analog to digital conversion, high speed data acquisition, and printed circuit board design.

Interface & Digital Electronics NXP Cup (Spring 2017)
Developed an autonomous car for competition at Imagine RIT 2017. Designed and built hardware and software to capture and process sensor data. Developed an autonomous driving algorithm based on signal data.

Languages C#, .NET, Python, TypeScript, JavaScript, Bicep, YAML, Kusto, C/C++, Assembly, Java, VHDL, Verilog