Kevin Liedtke

SOFTWARE ENGINEER | SOFTWARE ARCHITECT | USER EXPERIENCE ARCHITECT
Portland, OR | (503) 936-5005 | kevindliedtke@gmail.com | LinkedIn | Portfolio

SUMMARY

- Engineer and architect with 20+ years building production software across drivers, middleware, developer tooling, and user-facing applications.
- Defined user experience architecture for Intel's wired Ethernet software portfolio, aligning APIs, GUIs, CLIs, and engineering workflows used by network administrators and a 1,000+ engineer organization.
- Improves reliability, usability, and developer productivity through architecture, automation, and workflow optimization.
- Applies user experience thinking to engineering architecture, aligning APIs, tools, and interfaces to customer workflows.
- Experienced across the full software lifecycle, from early architecture and requirements through implementation, validation, and production support.

CORE COMPETENCIES

Software Architecture | Full-Stack Development | Developer Tooling | Workflow Automation
User Experience Strategy | Human Factors Engineering | Prototyping | Usability
Technical Leadership | Mentorship | Cross-Functional Collaboration | Design–Engineering Partnership
Process Optimization | Lifecycle Governance | Data-Driven Decision Making

TECHNICAL SKILLS

Programming: C++, C#, Python, SQL, JavaScript, TypeScript, PowerShell
Web & APIs: REST APIs, Node.js, React, Next.js
Databases: SQL Server, PostgreSQL, MySQL
Operating Systems: Windows, Linux
Networking: TCP/IP, UDP, DNS, DHCP, ARP, routing protocols, switching
Tools: Git, Docker, Visual Studio, VS Code, Jira, Figma
Processes: CI/CD practices, automated validation, code review, Agile development

PROFESSIONAL EXPERIENCE

INTEL CORPORATION – Hillsboro, OR | 2005 – 2024
Global semiconductor company delivering networking hardware and software platforms.

Senior Systems Software Architect | Mar 2022 – Sep 2024 | Remote
UX architecture lead for Intel's wired Ethernet portfolio, defining experience direction across customer-facing products and internal engineering platforms and representing UX priorities in architecture committees.

- Reduced driver triage time 30% and maintenance overhead 25% by designing an automated validation framework using Python, C#, and SQL that embedded metadata for version and device traceability across production driver releases.
- Improved development efficiency 15% across four global engineering divisions by standardizing development workflows and aligning requirements practices across engineering teams.
- Cut requirements review cycles 50% and increased SLA compliance 45% by building automation pipelines, dashboards, and monitoring tools that surfaced workflow bottlenecks and enforced review timelines.
- Led architectural reviews across hundreds of thousands of lines of C++, C#, JavaScript, ASP.NET, and Python, guiding system design decisions, maintainability standards, and cross-team development practices.
- Increased engineering participation in requirements processes 200% by presenting data-driven process improvements at a global management summit and aligning stakeholders across engineering, QA, and support.

Senior Human Factors Engineer, June 2012 – March 2022
Owned UX architecture and design strategy for Intel's wired Ethernet software ecosystem, aligning APIs, GUIs, CLIs, and hardware–software interaction for both external network administrators and internal engineering teams.

- Integrated UX architecture into early hardware and software requirements to drive consistent interaction models across the product portfolio.
- Reduced design iteration cycles 30–40% by prototyping workflows in Figma and C#, accelerating validation and feedback across distributed teams.
- Streamlined support operations, reducing tickets 20% and resolution time 15% by designing and implementing a centralized debugging and documentation hub used across multiple organizations.
- Built and led a distributed UX team in Poland providing continuous design coverage across global engineering teams.
- Generated dozens of patent submissions by leading innovation workshops and patent training for engineers.
- Influenced more than $50M in product retirement decisions as representative on Intel's End-of-Life committee.

Network Software Engineer, August 2005 – June 2012

Delivered production networking software across Intel's wired Ethernet stack, building components from middleware and hardware interfaces to configuration tools and user interfaces.

- Delivered full-stack networking tools across 40+ product releases contributing to over $1B in product revenue.
- Developed installers, middleware, services, and UI layers using C++, C#, and PowerShell.
- Built Windows configuration interfaces using C++, C#, WTL/ATL, and scripting frameworks.
- Managed a $250K localization budget and coordinated with globalization teams supporting worldwide product releases.

PERSONAL PROJECTS

- Engineered a Node.js module with an HTML/CSS/JavaScript front-end for MagicMirror.
- Developed a Python-based Discord bot for stock and cryptocurrency data retrieval.
- Created an open-source OctoPrint plugin using Python and Jinja to interface with custom thermal sensor.
- Wrote a responsive portfolio site using Next.js, Tailwind CSS, and TypeScript.

EDUCATION

Bachelor of Science (BS) in Computer Science; Minor in Business Administration
OREGON STATE UNIVERSITY, Corvallis, OR

AWARDS & RECOGNITION

Intel Technical Lead – *Awarded April 2020*
Recognized by Intel's Principal Engineer committee for demonstrated impact in technical leadership, innovation, strategy, and mentoring. Selection based on contributions to UX and Human Factors Engineering across Intel's software division.

PATENTS

Granted

- US-12001826-B2 - Device Firmware Update Techniques
- US-12254304B2 - Firmware update techniques

Pending

- US-20230124192-A1 - Heating and Cooling Systems for Edge Data Centers
- US-20220114011-A1 - Methods and Apparatus for Network Interface Device-Based Edge Computing
- US-20210041929-A1 - Dynamic Network Controller Power Management
- US-20200322287-A1 - Switch-Managed Resource Allocation and Software Execution
- US-20200177660-A1 - Offload of Streaming Protocol Packet Formation