



A High AI-Q[™]
Company



GenAI-Powered Legacy Modernization for Scalable Growth

Reimagining a legacy monolithic system with AI-driven microservices to enhance scalability, accelerate delivery, and ensure long-term maintainability.

Overview

- Migrated legacy ASP Classic monolith to .NET-based microservices and Vue.js microfrontends
- Integrated Cursor AI agents to automate code generation and enforce architectural standards
- Accelerated development velocity by 80%, significantly improving scalability and maintainability



Client Profile

A cloud software provider specializing in end-to-end research grants management for universities, funding agencies, and research organizations. The platform powers the entire grant lifecycle from submission to reporting and analytics.

Challenges: Overcoming Monolithic Limitations

- **Tightly Coupled Architecture:** Existing monolith made updates complex and time-consuming.
- **Scalability Bottlenecks:** Inability to scale individual modules independently.
- **Technical Debt:** Legacy tech stack restricted adoption of modern tools and slowed innovation.
- **Performance Issues:** Slow response times and inefficient data handling affected user experience and real-time reporting.
- **Operational Inefficiency:** High maintenance overhead and long release cycles impeded growth.

Solution: AI-Accelerated Legacy Modernization

We implemented a three-phase modernization roadmap powered by Cursor AI agents, combining human expertise with AI automation to deliver a future-ready, modular architecture.

- **Foundation Architecture Setup:** Established a scalable microservices framework with coding patterns, Cursor AI rules, and architectural enforcement mechanisms.
- **AI Integration:**
 - Defined reusable rules and patterns for CQRS, controllers, and frontend components
 - Built automated compliance and migration guides for quality and consistency
 - Integrated Figma MCP for extracting UI components directly from designs
- **AI-Driven Development Workflow:**
 - Automated migration of ASP pages to .NET microservices and Vue components
 - Context-aware code generation using “one-shot” prompts to create APIs, components, tests, and documentation

Implementation Approach: Modular, Automated, Future-Ready

- **Strangler-Fig Migration Pattern:** Ensured risk-free, incremental modernization while maintaining business continuity.
- **AI-Guided Code Generation:** Reduced manual development effort through automated, pattern-consistent code creation.
- **Microservices Architecture:** Enabled independent service deployment and fault isolation for better scalability.
- **Standardized Patterns and Prompts:** Ensured code consistency and maintainability across services.
- **Developer Onboarding Efficiency:** AI-driven documentation and contextual automation accelerated onboarding for new developers.

Impact

- **Accelerated Development:** 80% faster development velocity with AI-assisted code generation and architectural enforcement.
- **Independent Domain Scalability:** Enabled independent scaling of business domains such as Auth, Committees, and Applications.
- **Lower Operational Costs:** Reduced operational costs through modular design and automation-led efficiency.
- **Improved Maintainability:** Improved maintainability and developer productivity with enforced coding standards.
- **Enhanced Performance:** Improved performance and user experience through modern, loosely coupled architecture.