

Winning a Hackathon for Modernizing Tens of Legacy Applications

During a recent three-day hackathon, SORINT put its AI-augmented Software Development Lifecycle (SDLC) practices and toolchain to the test under high-pressure conditions. Driven by a short-notice participation request from a prominent European client in the Banking & Finance sector, the team demonstrated how AI-driven workflows can dramatically accelerate legacy modernization.

React Node.js Figma AWS Terraform Docker PostgreSQL Python

Objective

- 1 Define and validate an AI-native SDLC operating model capable of modernizing a portfolio of dozens of legacy software solutions.
- 2 Teams must apply this model to refactor and modernize, on spot, one of the assigned applications.

Assigned Application

Rewrite the core subset of features to rebuild a 10-year-old legacy financial application using modern tech stack across infrastructure and application layer.



Zero-prior Knowledge

The team had no pre-existing familiarity with the business intent, the proprietary legacy solution, architecture, or underlying logic.



Minimal Documentation

Technical onboarding is strictly limited to a raw code repository and a high-level briefing document.



Strict UI/UX Compliance

Deliverables require 100% adherence to client's predefined enterprise design system and brand guidelines.

To tackle this, we assembled a cross-functional, multidisciplinary team ready to blend product intuition with engineering execution. Mirroring the business context by orchestrating several agentic AI solutions.



The “Signal Out of the Noise” Playbook

The winning formula relied on a well-structured governance model, effectively blending human judgement and collaboration with multi-agent workflows and tools.

Legend:



Requirement Gathering

A multi-agent workflow converted undocumented legacy code and fragmented domain knowledge into a structured, human-verified Product Requirements Document (PRD).



Code Analysis & Logic Extraction: Parsed the legacy codebase to reverse-engineer functional logic, underlying business rules, and technical dependencies.



Technical Audit: Stress-tested extracted specifications against edge cases and system constraints from a strict engineering perspective.



PRD Synthesis: Consolidated findings into 40 coherent Epics, which were subsequently reviewed and approved by key business and technical stakeholders.

Verified user stories



UX/UI Design

Advanced from approved User Stories to production-ready mockups in under an hour through a three-step accelerated workflow that ensured strict adherence to the client's design system.



AI Wireframing: AI co-pilot to parse User Stories and auto-generate interactive, low-fidelity prototypes.



Design System Alignment: Executed human-in-the-loop refinements to align wireframes with the client's enterprise UI kit.



Visual Validation: Conducted rapid design reviews for instant sign-off from both our team and the client's.

Navigable HTML mockups



Code Curation

Deployed specialized code-generation agents to build target components directly from the verified PRD, User Stories, and UI wireframes.



Prompted agents with strict architectural rules, coding standards, business logic, and security parameters to ensure high-quality outputs.



Developers shifted from writing raw code to curating and optimizing AI-generated modules.

Production-Ready Codebase



Automated Quality Assurance

Replicated and modernized the previous test strategy by generating comprehensive functional, integration, and end-to-end (E2E) test suites.



Analyzed legacy test patterns and business rules to auto-generate modern test scripts.



QA Engineer validated AI-generated test coverage, executed acceptance testing to verify that modernized features functioned exactly as intended.



Infrastructure as a Code

Evaluated the legacy hosting environment and provisioned a modern, cloud-native deployment pipeline in minutes using multi-agent DevOps workflow.



Parsed the client's infrastructure documentation to extract naming conventions, security policies, networking rules, and compliance constraints.



Architected and generated production-ready Infrastructure as Code (IaC) templates using Terraform on AWS.

Pitching Results. We Won!

Successful blending of AI expertise, foundational software engineering skills, and foundational human skills. To ensure client's teams could autonomously tackle the remainder of its aging application suite, we finalized the engagement by delivering a comprehensive framework and tangible proof of concept.

1. Designing a Playbook

A standardized, ready for adaption, AI-driven SDLC modernization framework that balances AI technologies, with human cognition. AI agents to ingest, map, and document unrecorded legacy codebases, effectively reverse-engineering business logic before development begins.

Check marking business validation, equivalency, and industry regulations.

2. One Rebuilt Solution

Deployed in 3 days and compresses both, modernization timelines, and associated risks, exponentially. Full modern scalable technical stack in terms of the software architecture, user interface, codebase, containerized infrastructure and met security standards.

Case study

Application Modernization - Agentic AI & Human Engineering - Banking & Finance



BUILDING GREAT TECHNOLOGY
www.sorint.com