

Vincere un hackathon per modernizzare decine di applicazioni legacy

Durante un recente hackathon di tre giorni, SORINT ha messo alla prova le proprie pratiche e la toolchain del ciclo di vita dello sviluppo software (SDLC) potenziate dall'IA. Rispondendo a una richiesta con breve preavviso da parte di un importante cliente europeo nel settore bancario e finanziario, il team ha dimostrato come i flussi di lavoro basati sull'IA possano accelerare drasticamente la modernizzazione dei sistemi legacy.

React Node.js Figma AWS Terraform Docker PostgreSQL Python

Obiettivo

- 1 Definire e validare un modello operativo SDLC nativo per l'IA, in grado di modernizzare un portafoglio di decine di soluzioni software legacy.
- 2 Applicare questo modello per effettuare il refactoring e modernizzare, in tempo reale, una delle applicazioni assegnate.

Applicazione assegnata

Riscrivere il sottoinsieme di funzionalità principali per ricostruire un'applicazione finanziaria legacy di 10 anni fa, utilizzando uno stack tecnologico moderno a livello infrastrutturale e applicativo.



Nessuna conoscenza pregressa

Il team non aveva alcuna familiarità con gli obiettivi di business, la soluzione legacy proprietaria, l'architettura o la logica sottostante.



Documentazione minima

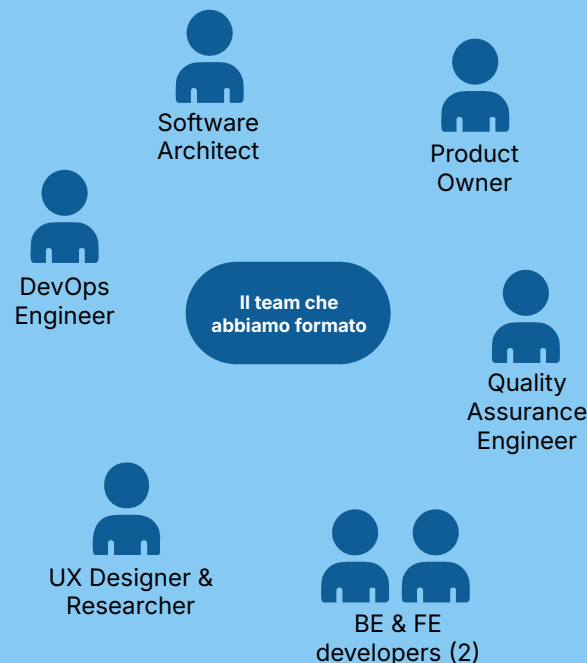
L'onboarding tecnico è strettamente limitato a un repository di codice grezzo e a un documento di briefing ad alto livello.



Rigorosa conformità UX/UI

I deliverable richiedevano un'aderenza del 100% al design system aziendale e alle linee guida del brand del cliente.

Per affrontare questa sfida, abbiamo riunito un team interfunzionale e multidisciplinare, in grado di combinare l'intuito di prodotto con l'esecuzione ingegneristica e di rispecchiare il contesto di business attraverso l'orchestrazione di diverse soluzioni di IA agentic.



Il Playbook: Estrarre il segnale dal rumore

La formula vincente si è basata su un modello di governance ben strutturato, capace di combinare efficacemente il giudizio, la supervisione e la collaborazione umana con flussi di lavoro e strumenti multi-agente.

Legenda:

 Agente

 Umano



Raccolta di requisiti

Un flusso di lavoro multi-agente ha trasformato il codice legacy non documentato e la conoscenza frammentata del dominio in un Product Requirements Document (PRD) strutturato e validato dal team.



Analisi del codice ed estrazione della logica: Eseguito il parsing del codebase legacy per effettuare reverse engineering della logica funzionale, delle regole di business sottostanti e delle dipendenze tecniche.



Audit Tecnico: Le specifiche estratte sono state sottoposte a stress test per valutarne la tenuta rispetto ai casi limite e ai vincoli di sistema, garantendo rigore ingegneristico.



Sintesi del PRD: I risultati sono stati consolidati in 40 Epic coerenti, successivamente revisionati e approvati dai principali stakeholder tecnici e di business.

User stories verificate



UX/UI Design

Passaggio dalle User Story approvate a mockup pronti per la produzione in meno di un'ora, attraverso un flusso di lavoro accelerato in tre fasi che ha garantito una rigorosa aderenza al design system del cliente.



AI Wireframing: Utilizzo di un co-pilota IA per analizzare le User Story e generare automaticamente prototipi interattivi a bassa fedeltà.



Allineamento al Design System: Eseguiti perfezionamenti human-in-the-loop (con intervento umano) per allineare i wireframe all'UI kit aziendale del cliente.



Validazione Visiva: Condotte rapide revisioni del design per l'approvazione immediata sia da parte del nostro team che di quello del cliente.

Mockup HTML interattivi



Cura del codice

Sono stati implementati agenti specializzati nella generazione di codice per sviluppare i componenti target direttamente dal PRD validato, dalle User Story e dai wireframe della UI.



Forniti prompt con rigide regole architetture, standard di codifica, logica di business e parametri di sicurezza per garantire output di alta qualità.



Gli sviluppatori sono passati dallo scrivere codice da zero al curare, perfezionare e ottimizzare i moduli generati dall'IA.

Codebase Pronta per la Produzione



Quality Assurance automatizzata

La precedente strategia di test è stata replicata e modernizzata generando suite di test funzionali, di integrazione ed end-to-end (E2E).



Analizzati i pattern di test legacy e le regole di business per generare automaticamente script di test moderni.



Il QA Engineer ha validato la copertura garantita dall'IA ed eseguito i test di accettazione per assicurarsi che le funzionalità modernizzate operassero esattamente come previsto.



Infrastruttura come Codice (IaC)

L'ambiente di hosting legacy è stato valutato per effettuare il provisioning di una pipeline di deployment moderna e cloud-native in pochi minuti, utilizzando un flusso di lavoro DevOps multi-agente.



Analizzata la documentazione dell'infrastruttura del cliente per estrarre le convenzioni di nomenclatura, le policy di sicurezza, le regole di networking e i vincoli di conformità (compliance).



Progettati e generati template di Infrastructure as Code (IaC) pronti per la produzione utilizzando Terraform su AWS.



Configurati flussi di lavoro di build containerizzati tramite Docker e automatizzate le pipeline di rilascio.

Presentazione dei risultati. Abbiamo vinto!

Grazie a un mix perfetto tra IA, ingegneria del software e competenze umane, abbiamo consegnato un framework e un PoC in grado di rendere il cliente autonomo nella modernizzazione delle restanti applicazioni obsolete.

1. Progettazione di un Playbook

Un framework SDLC adattabile che bilancia IA e intelletto umano. Utilizza agenti IA per mappare il codice legacy non documentato ed estrarne la logica di business tramite reverse engineering prima dello sviluppo, soddisfacendo rigorosamente i requisiti di validazione, l'equivalenza e le normative di settore.

2. Una soluzione ricostruita

Rilasciata in 3 giorni, la soluzione riduce in modo esponenziale sia le tempistiche di modernizzazione che i rischi associati. Il risultato è uno stack tecnico completo, moderno e scalabile in termini di architettura software, interfaccia utente, codebase, infrastruttura containerizzata, nel pieno rispetto degli standard di sicurezza.

Caso studio

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



BUILDING GREAT TECHNOLOGY

www.sorint.com