



# IA aplicada à jornada de talentos

UMA ARQUITETURA DE AGENTES CONSTRUÍDA SOBRE A STACK, AS REGRAS DE NEGÓCIO E OS RISCOS REAIS DA OPERAÇÃO.

Como a Impulso **redesenhou seu fluxo de engenharia** com **agentes integrados à operação** e **decisões técnicas sob controle de profissionais experientes.**

# Antes de acelerar o código era preciso reorganizar o caminho até ele

Antes da nova esteira, as demandas entravam em uma fila, eram priorizadas pela liderança, aguardavam disponibilidade técnica e percorriam desenvolvimento, revisão de código, teste e produção.

Demandas dos diferentes produtos estavam **distribuídas entre múltiplos fluxos e ferramentas**, dificultando a organização e o início do trabalho.

Alguns sistemas foram desenvolvidos ao longo de **mais de 16 anos** e reuniam regras de negócio difíceis de localizar. Por isso, **entender a origem de um problema exigia uma investigação demorada.**

Um bug podia levar de dois a quatro dias para chegar à produção. Novas funcionalidades consumiam semanas. Nos casos críticos, a pressão por velocidade levava a correções diretas no banco de dados, fora do fluxo normal da aplicação.

O gargalo não estava apenas na escrita do código. Boa parte do tempo era consumida para **localizar a demanda, reconstruir o contexto, preparar ambientes, reproduzir o erro e coordenar a passagem do trabalho entre pessoas e ferramentas.**



# O desafio era acelerar sem trocar um gargalo por outro



## CORREÇÕES EMERGENCIAIS COM EFEITOS COLATERAIS

**Alterações manuais** no banco aceleravam respostas imediatas, mas **podiam ignorar processos executados pelo código e gerar inconsistências posteriores.**

## CONTEXTO EXCESSIVO PARA AGENTES E REVISORES

**Entregar funcionalidades inteiras a uma única execução** ampliava a janela de contexto, **aumentava o risco de perda de informações** e gerava conjuntos extensos de mudanças difíceis de revisar.

## MODELOS GENÉRICOS DISTANTES DA OPERAÇÃO

Abordagens e kits de desenvolvimento assistido por IA disponíveis no mercado ofereciam **fluxos e templates genéricos** ou **pressupunham estruturas diferentes da nossa.** Nos testes, adotá-los integralmente **aumentaria a complexidade sem contemplar as regras de negócio, a stack e a infraestrutura da Impulso.**

# A arquitetura nasceu de testes e correções no fluxo real

A **Impulso** começou estudando e testando abordagens disponíveis no mercado para desenvolvimento assistido por IA. A partir desses experimentos, **a equipe selecionou os componentes aderentes à operação, descartou o excesso e construiu uma estrutura própria**, combinando agentes, skills, regras de negócio, padrões de código e informações da infraestrutura.

**O diferencial da Impulso está em conhecer a engenharia por dentro.** Entendemos como o time e os sistemas operam, identificamos o que já funciona e adaptamos cada abordagem à realidade da operação. **Agentes e profissionais propõem e testam melhorias, respondem rapidamente às falhas e evoluem a arquitetura continuamente**, de acordo com o ambiente, o time e o risco de cada entrega.



# Handoffs menores preservam contexto e tornam o trabalho verificável

## PLANEJAMENTO GRANULAR

A demanda é quebrada em **microtarefas**, agrupadas por milestones. **Cada execução recebe uma responsabilidade delimitada**, como criar uma tela, integrar o backend ou implementar uma etapa de autenticação.

## UMA JANELA DE CONTEXTO POR ETAPA

**Ao concluir a tarefa, o agente encerra sua sessão e transfere ao próximo somente o estado necessário para continuar.** A nova execução não precisa reler todo o raciocínio anterior nem carregar informações que não serão usadas naquela etapa.

## INVESTIGAÇÃO APOIADA POR CÓDIGO E DADOS

Os agentes consultam o código, a estrutura dos serviços e os dados necessários à análise para localizar possíveis causas sem depender da reprodução manual de todo o ambiente. **O diagnóstico retorna ao time antes de qualquer correção seguir no fluxo.**

## PADRÕES E COMPONENTES REUTILIZÁVEIS

**Agentes criam e mantêm padrões de formulários e listagens, além de componentes reutilizáveis para outras partes do sistema.** A cada demanda, adaptam o que já existe. O time valida as entregas e concentra seu esforço nas regras de negócio e nas exceções.

# Mais velocidade exige pontos claros de decisão humana

O fluxo foi desenhado para que **profissionais consigam verificar o trabalho em etapas menores e interromper, corrigir ou redirecionar a execução antes que um erro se espalhe.**

## VALIDAÇÃO HUMANA NOS PONTOS CRÍTICOS

**O agente analisa, propõe, implementa e executa as verificações que podem ser automatizadas.** Com base em critérios definidos pelo time, avalia o impacto de cada PR e identifica o que pode seguir pela esteira. Quando uma mudança afeta regras de negócio críticas, **destaca os pontos que exigem validação humana antes do merge.**

## REVISÃO PROPORCIONAL AO RISCO

Em componentes críticos, **a IA não substitui a avaliação de segurança e impacto feita por profissionais experientes.** Quanto maior a consequência da mudança, maior deve ser a profundidade da revisão.

## CORREÇÃO DA PRÓPRIA ARQUITETURA

**Quando uma entrega foge do padrão esperado, o time não se limita a corrigir o código.** Investiga por que o agente se desviou e ajusta regras, skills, instruções ou a divisão das tarefas. **Foi assim que o excesso de contexto deu origem ao modelo atual de handoffs.**

## AUTONOMIA DELIMITADA PELO TIPO DE DECISÃO

Nem toda etapa migra para a IA. **O time define onde o agente pode acelerar análise e execução e onde a consequência exige julgamento técnico.**



# Mais velocidade nas etapas que antes consumiam tempo técnico

DE 2 A 4 DIAS  
PARA CERCA DE  
30 A 40 MINUTOS

Em diferentes casos, **agentes corrigiram em menos de uma hora bugs que antes levavam dias para chegar à produção.** Em uma dessas ocorrências, o suporte abriu um card enquanto o time estava na reunião diária. O agente identificou o bug, implementou a correção, abriu o PR e avisou um profissional do time. **Antes de a reunião terminar, o problema estava resolvido e a operação já havia sido informada.**

DE ATÉ 1 DIA E MEIO  
PARA 5 A 10 MINUTOS

Com apoio dos agentes, **profissionais em diferentes níveis de experiência passaram a investigar código, bancos de dados e infraestrutura com profundidade,** chegando a hipóteses fundamentadas em minutos. O time valida as conclusões antes de seguir para a correção.

DE 2 A 4 HORAS  
PARA 2 A 10 MINUTOS

**Demandas comuns de desenvolvimento** — como criar uma tela de cadastro, adicionar filtros ou implementar buscas — **passaram a ganhar uma primeira versão funcional em minutos,** com apoio de componentes reutilizáveis e padrões já definidos.

## UM SISTEMA PRONTO PARA VALIDAÇÃO EM DUAS SEMANAS

**Em cerca de duas semanas, desenvolvemos do zero um sistema interno e colocamos uma versão funcional nas mãos do time responsável.** A equipe pôde testar o funcionamento, validar os fluxos e indicar os ajustes necessários para sua evolução.

## DO ERRO AO PR DRAFT

Em diferentes ocorrências registradas pelo monitoramento, **um agente identifica o projeto e o repositório afetados e aciona automaticamente a esteira.** O agente de correção analisa o erro, o stack trace e o código envolvido, implementa a mudança e abre um pull request em rascunho. **Os pontos de maior risco são destacados para validação humana.**

# O que a operação real fala sobre a AI Development Pipeline?



“

Não buscamos criar uma solução universal. Times, projetos e fluxos têm suas particularidades. **Acompanhamos o que o mercado oferece, testamos abordagens na nossa realidade e incorporamos o que funciona. Quando necessário, adaptamos ou desenvolvemos componentes próprios.**

Assim, a esteira continua evoluindo com as necessidades da operação e o aprendizado de cada entrega.

**Hoje, análises e correções que antes levavam dias podem acontecer em minutos.** Os agentes realizam as verificações possíveis e **sinalizam mudanças que exigem julgamento humano**, especialmente quando afetam regras de negócio críticas. **O valor está em combinar velocidade, adaptação e controle proporcional ao risco.**

”



Renato Ferraz  
**Staff Software Engineer**  
@ Impulso



# Da demo ao workflow em uso

Com **qualidade, governança e evidência para decidir escala**, a **Impulso** combina **produto, engenharia e IA aplicada** para transformar pilotos, agentes, automações e gargalos de tecnologia em **operação controlada**.

