

softo

O custo
invisível
do backlog
tecnológico

O que a sua empresa perde enquanto

a demanda espera na fila e por que esse

número nunca aparece em nenhum relatório.

SEÇÃO 01

O backlog atrasado tem um custo que ninguém está calculando

Todo time de tecnologia tem um backlog. Tem ticket, tem estimativa, tem sprint, tem uma reunião toda segunda-feira para priorizar o que vai e o que fica. O painel está aberto e os números estão lá.

O que não está no painel é o custo do que ficou esperando.

A integração com o ERP que está há oito meses na fila. Durante esse tempo, alguém consolidou dados na mão, toda semana. Quantas horas? Qual o custo real disso?

A funcionalidade de autoatendimento que ficou três sprints esperando. Durante esse tempo, quantos tickets de suporte foram abertos? Quantas horas do time de atendimento foram consumidas?

A automação de aprovação que está no backlog desde o ano passado. Quantas aprovações foram feitas manualmente desde então? Por quantas pessoas?

Esses números existem. São reais, mensuráveis e quase nunca são calculados.

O backlog atrasado não é só uma lista de coisas a fazer. É uma lista de custos sendo pagos agora, sem ticket, sem ninguém percebendo.

SEÇÃO 02

O que aparece no relatório e o que não aparece

Os custos que aparecem no relatório são os custos de desenvolvimento: horas de time, bugs em produção, retrabalho, sprints que não fecharam. Esses números têm dono, têm ticket, aparecem no board do projeto.

Mas há outro conjunto de custos que ficou órfão, sem área responsável, porque ninguém criou um lugar para registrá-los.

O QUE TEM TICKET NO JIRA

Bug em produção

Sprint que não fechou

Retrabalho de desenvolvimento

Reuniões de alinhamento

O QUE NÃO TEM DONO EM LUGAR NENHUM


Receita que não entrou porque a funcionalidade atrasou

Decisão que esperou 3 meses por um relatório manual

Cliente que foi para o concorrente enquanto a integração não saía

Processo que continuou ineficiente por 18 meses enquanto a automação esperava na fila

A coluna da direita fica distribuída entre áreas, diluída no tempo, invisível até o momento em que some algo grande demais para ignorar. Sem responsável, sem orçamento, sem post-mortem.



Receita adiada, decisões que esperaram meses por um relatório, clientes perdidos para o concorrente, esses custos existem e se somam trimestre a trimestre. Apenas sem ticket.

Uma empresa que calcula o custo real do seu backlog, não só o custo de desenvolvimento, mas o custo de não ter entregado, costuma encontrar um número muito maior do que esperava.

SEÇÃO 03

Por que contratar mais não resolve

Quando o backlog cresce, a resposta padrão é contratar. Faz sentido na superfície: mais pessoas, mais capacidade, fila menor.

O backlog cresce por estrutura. Gente é o que a empresa tem. O que falta é um modelo diferente.

CAUSA ESTRUTURAL

Tudo passa pelo mesmo funil

O modelo de projeto tem data de fim

Falta de sistemas que operam sozinhos


O QUE ISSO PRODUZ

O time técnico é o único ponto de entrada para qualquer demanda. Uma integração crítica espera a mesma fila que um ajuste de CSS.

Projetos terminam. As demandas não. Quando o projeto fecha, as próximas já formaram outra fila e o ciclo recomeça.

Cada processo que deveria ser automático vira uma demanda nova para o time. O backlog cresce com trabalho que não precisava existir.

O resultado é previsível, a empresa contrata, e o backlog encolhe por um trimestre. As demandas continuam chegando no mesmo ritmo. A fila volta ao tamanho anterior, só que o custo fixo cresceu.



Mais pessoas, mesmo backlog, conta maior.

Enquanto o modelo for 'o negócio pede, o time desenvolve', a proporção não muda. Contratar mais aumenta a capacidade, mas não muda a relação entre a velocidade de entrada e a velocidade de saída do backlog.

SEÇÃO 04

A lógica que muda o jogo: métodos que entregam vs métodos que acumulam


Existe uma distinção importante que a maioria das empresas não faz:

UMA FUNCIONALIDADE RESOLVE UM PROBLEMA UMA VEZ.

Um método orientado a outcome resolve aquela categoria de problema de forma contínua.

Quando uma empresa constrói uma funcionalidade de geração de relatório, ela entrega o relatório uma vez. Qualquer mudança no relatório vira uma nova demanda. O backlog cresce.

Quando uma empresa monta um processo responsável por gerar inteligência operacional contínua, o relatório é uma consequência automática. A demanda desaparece, o backlog não volta para aquela função.



A pergunta não é 'quando essa funcionalidade fica pronta?'. É 'qual processo essa empresa nunca mais vai precisar fazer na mão?'

Essa mudança de lógica é o que separa empresas que estão reduzindo backlog de empresas que estão gerenciando backlog. A diferença é de modelo, antes de ser de velocidade.

SEÇÃO 05

Como IA entra no processo de engenharia

IA aplicada de forma fragmentada, cada desenvolvedor com sua ferramenta, sem processo definido, pode até aumentar o backlog. O time fica mais rápido para criar novas demandas do que para resolvê-las.

O ganho real aparece quando a IA é parte do ciclo de engenharia com um processo definido.

Três contextos onde isso acontece com consistência:

ACELERAÇÃO DA CODIFICAÇÃO

Geração de código, scaffolding de features, implementação de padrões repetitivos. Tarefas que consumiam dias de desenvolvimento são executadas em horas. O time sênior para de escrever o que pode ser gerado e passa a focar em arquitetura, decisões críticas e alinhamento ao negócio.

AUTOMAÇÃO DE TESTES E QUALIDADE

Geração de testes unitários e de integração, code review automatizado, identificação de regressões. A qualidade deixa de depender exclusivamente da disponibilidade do time sênior e passa a ser parte do processo, não um gargalo no final do sprint.

REDUÇÃO DE RETRABALHO NO CICLO DE ENTREGA

Requisitos mal interpretados, integrações quebradas, bugs que voltam. Com SDLC agêntico, padrões de erro são identificados antes de virar ticket. O problema é resolvido na origem, não na próxima sprint.

Em todos os casos, o padrão é o mesmo: o que antes limitava o throughput do time passa a ser tratado por um processo mais inteligente. Mais funcionalidades chegam em produção, com mais previsibilidade e menos retrabalho.

PARA FECHAR

Se o backlog cresce todo trimestre, o problema não é o time

Times bons com modelos ruins produzem backlogs inchados. O modelo garante que a fila nunca vai ficar num tamanho saudável, independente de quantas pessoas estiverem nele.

Empresas que saíram dessa situação de backlog inchado, gerando custo de oportunidade, mudaram o método de trabalho e estão apostando no uso intensivo de IA para desenvolvimento.

Outcome Pods é como a Softo estrutura esse trabalho: métodos orientados a um resultado específico, com IA e automação integradas ao desenvolvimento, responsáveis por produzir resultados de forma contínua.

Se o modelo é o problema, mudar o modelo é o próximo passo.

CONHEÇA OS OUTCOME PODS DA SOFTO

DESCUBRA COMO APLICAR IA PARA

ACCELERAR RESULTADOS DENTRO DA SUA OPERAÇÃO.

sof.to

sof.to/pt-BR/services/outcome-pods