

Débito Técnico e Obsolescência: dramas do dia a dia de qualquer IT!

Para ser mais exato, dramas de qualquer organização, afinal de contas, resolver ou mitigar ambos os temas demandam escolhas, foco e recursos (opex e capex) da empresa como um todo, não apenas de IT.

E em um mundo onde a realidade dos “recursos escassos” se mostra cada vez mais forte, desviar a atenção para débitos técnicos e obsolescência muito provavelmente trará a necessidade de escolher entre arrumar a cozinha ou investir em iniciativas transformacionais e com mais apelo de negócios.

Nessa hora a disputa por recursos entre o “run” e o “change” pode ficar bem quente,

com muitos argumentos para ambos os lados!

Aqui um artigo da curto que achei bem bacana sobre essa questão da priorização:

https://www.maxcountryman.com/articles/a-framework-for-prioritizing-tech-debt

De qualquer forma, uma premissa que eu julgo importante é a de que toda solução tem seu ciclo de vida e ter débitos técnicos, especialmente por obsolescência é algo (até onde eu sei) inevitável em algum momento.

O que podemos escolher é como endereçar isso: de forma cotidiana e proativa, ou de forma displicente e encarar uma crise quando a plataforma colapsar.

Mas um ponto adicional que vale refletir e comentar mais uma vez é o aspecto de quanto o modelo Agile pode impactar ainda mais o risco e as dores do débito técnico.

Infelizmente acaba sendo uma tendência do modelo Agile dar menos atenção ao planejamento e desenho de médio e longo prazo, priorizando a visão de curto alcance (uma outra sprint ou no máximo a PI). Isso por si só já aumenta o risco de se criar algo mais difícil de manter e atualizar no futuro.

Depois tem a questão de priorizar as ações de mitigação/resolução de débito técnicos dentro das sprints.

Verdade seja dita, as demandas com impacto no business tendem a ganhar na hora da prioridade, muito pelo próprio entendimento geral de que o PO (primordialmente do business) é quem tem a prerrogativa usual de priorizar o que entra na sprint.

Soma-se a isso o fato de que muitas vezes as “contrapartes” de IT ou possuem menor poder de decisão (por exemplo um scrum master terceiro com pouca autonomia ou senso de ownership sobre a aplicação, ou mesmo arquitetos com foco apenas no desenho da solução, mas não na gestão do lifecycle do sistema ou infra).

Faz parte dos desafios de IT ser capaz de conciliar a agilidade com o zelo e excelência na tanto na criação quanto na manutenção das soluções.