Malorean Corp.

Um romance para se identificar

Um panorama para compreender

Desenvolvedores, IA e pequena empresa : uma história (quase) real de transformação

O que vivem de verdade as equipes de um SaaS ERP com a IA hoje



LAURENT DESÉCHALLIERS

Malorean Corp

> *Até pouco tempo atrás, transformar uma ideia em protótipo às vezes levava semanas. Hoje, uma pessoa com a ajuda da inteligência artificial pode conseguir isso em poucos dias. Essa mudança é impressionante. Mas a verdadeira transformação talvez aconteça em outro lugar: na maneira como trabalhamos juntos.*

Laurent Deséchalliers

Nota sobre a tradução

Esta edição é uma tradução automática, produzida por uma inteligência artificial a partir do texto francês.
Ela não foi revisada por um tradutor humano. As ideias, a estrutura e os exemplos são fiéis ao original; algumas construções, por outro lado, podem soar desajeitadas, e certas passagens guardarão marcas da língua de partida.
É o preço a pagar para que esta obra exista em vários idiomas, e não apenas em francês. Em caso de dúvida sobre uma formulação, a edição francesa é a referência.

1

Introdução

O que mais me impressiona na IA não é sua capacidade de escrever código. É a velocidade com que ela torna possível algo que, até pouco tempo atrás, exigiria vários dias ou semanas de trabalho.

Uma ideia pode se transformar em protótipo antes mesmo de terminar a discussão sobre sua viabilidade. Um desenvolvedor pode explorar uma possibilidade sem esperar que uma equipe inteira esteja disponível. Alguém que não sabe programar às vezes consegue criar uma primeira versão daquilo que imaginou. Isso não significa que um protótipo seja um produto confiável, nem que a complexidade tenha desaparecido. Mas a simples possibilidade de experimentar já muda nossa percepção do que parecia possível.

Acontece que nossas empresas não foram construídas para um mundo em que produzir uma primeira solução pode ser tão rápido.

Elas se organizaram em torno das competências das pessoas, do tempo disponível, das passagens de responsabilidade e das decisões necessárias para fazer o

trabalho avançar. Esses mecanismos têm sua razão de existir. Mas, quando as capacidades das pessoas mudam, começam a surgir outras perguntas. Ainda precisamos trabalhar na mesma sequência? Esperar pelas mesmas aprovações? Reservar as mesmas atividades às mesmas profissões?

Essas questões não dizem respeito apenas aos desenvolvedores. Product Managers, designers, profissionais de vendas, líderes de equipe e muitos outros também estão descobrindo novas formas de trabalhar. Alguns já conseguem fazer diretamente parte do que antes pediam aos colegas. Outros veem a própria atividade mudar de natureza. O que parecia ser apenas uma questão de ferramentas começa a afetar a maneira como colaboramos e tomamos decisões.

Seria tentador anunciar uma revolução cujo resultado já conhecêssemos. Não acredito que tenhamos chegado a esse ponto.

Algumas empresas experimentam novas formas de distribuir o trabalho e as responsabilidades. Outras avançam com mais cautela, às vezes por bons motivos. Certas práticas parecem promissoras, mas ainda não temos experiência suficiente para saber quais resistirão às exigências do dia a dia. E uma solução adequada a

uma equipe pequena não se torna automaticamente pertinente para uma empresa maior, mais regulamentada ou mais antiga.

Por isso, não quis escrever um manual sobre a organização ideal a adotar com a IA. Este livro é uma tentativa de compreender o que realmente está mudando: o que as ferramentas tornam possível, como as empresas reagem, o que ganham, o que correm o risco de perder e quais questões precisarão decidir.

Para isso, escolhi dois caminhos que se cruzam.

O primeiro é o da observação. Vamos examinar empresas reais, práticas divulgadas publicamente, estudos e experiências. Procurarei distinguir o que está comprovado do que ainda é hipótese, sem confundir um anúncio atraente com um resultado demonstrado.

O segundo é o de uma história. Vamos acompanhar a Malorean Corp., uma empresa fictícia cujos funcionários descobrem aos poucos como essas novas possibilidades mudam seu trabalho. Eles não têm um plano pronto para reinventar a organização. Têm clientes, hábitos, urgências, convicções e, às vezes, divergências. A ficção não prova nada; ela nos permite olhar para essas transformações do ponto de vista de quem as vive.

Precisamos dos dois olhares. Os exemplos reais ajudam a não confundir nossas intuições com fatos. A história nos lembra que as mudanças organizacionais não acontecem apenas nos organogramas: elas afetam a confiança, as competências, as responsabilidades e o lugar de cada pessoa.

No fundo, uma pergunta atravessa as páginas seguintes: **o que acontece em uma empresa quando as pessoas que fazem parte dela descobrem que podem fazer muito mais do que suas profissões lhes permitiam até então?**

Sobre o autor

Trabalho no setor de software há várias décadas e atualmente sou Product Manager. Minha trajetória passou pelo desenvolvimento de software e pela arquitetura de sistemas, depois pela organização de equipes e, mais tarde, pela gestão de produto, suas operações e seu marketing.

Essa experiência me coloca em uma posição privilegiada para observar a evolução da inteligência artificial e seus efeitos sobre nosso trabalho. Acompanho essas mudanças diariamente em minha atividade profissional, mas também nas conversas com outros Product Managers. Compartilhamos experiências, descobertas e questionamentos. Muitas vezes, essas trocas me levam a olhar além das ferramentas: o que realmente muda em nossas profissões, nas decisões que tomamos e na maneira como trabalhamos juntos?

À medida que observava essas transformações e conversava sobre elas, as perguntas se multiplicavam. Senti vontade de ganhar perspectiva, confrontar nossas impressões com o que acontecia em outras empresas e

compartilhar essas reflexões. Foi assim que este livro nasceu.

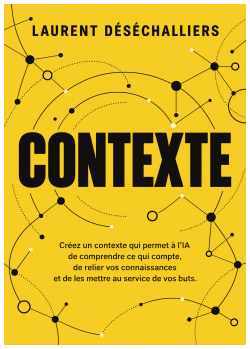
A escrita faz parte da minha trajetória há muito tempo. Contribuí para publicações especializadas e trabalhei como revisor técnico para a M2 Édition e a Eyrolles. Hoje escrevo em **malorean.net** sobre Product Management, IA, formas de trabalhar e sistemas de informação.

Também colaboro com a **Proposition Marketing**, uma revista independente dedicada ao marketing estratégico e operacional. Escrever para leitores das áreas de marketing e gestão empresarial me incentiva a tornar as questões de Produto acessíveis, sem presumir que todos dominem o mesmo vocabulário.

Laurent Deséchalliers

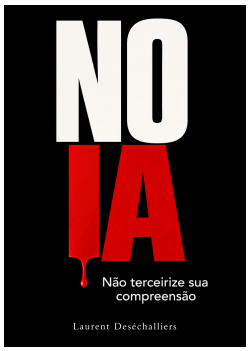
- **laurent@malorean.net**
- **https://malorean.net**

Do mesmo autor



Le Contexte — Como construir um contexto que permita à IA entender o que importa, conectar seus conhecimentos e capitalizar o que você aprende a serviço dos seus objetivos.

Lançamento: outubro de 2026.



NO IA — Não terceirize sua compreensão. Métodos sem IA para cultivar nossa memória, aprofundar nossa reflexão e exercitar nosso senso crítico. Práticas para desenvolver nossas próprias capacidades e aproveitar melhor a IA sem terceirizar a ela nossa compreensão.

Lançamento previsto: final de 2026.

Mais informações em **https://malorean.net**.

Agradecimentos

Quero agradecer, em primeiro lugar, aos Product Managers com quem converso regularmente. Trocamos pontos de vista sobre a evolução do Product Management, das profissões ligadas ao desenvolvimento de software e, de maneira mais ampla, sobre os efeitos da inteligência artificial no trabalho e na organização das empresas. Essas conversas, às vezes animadas, são uma fonte constante de perguntas, questionamentos e ideias novas. Algumas dessas reflexões estão presentes nas páginas deste livro. Obrigado pela curiosidade, pelas experiências compartilhadas e pela disposição para debater.

E quero agradecer também à minha esposa, Christine. Quando terminei meu primeiro livro, **Le Contexte**, prometi a ela uma pausa. Acabei começando este quase imediatamente. Felizmente, esta segunda aventura tomou menos de nossas noites e fins de semana do que a anterior. Obrigado, Christine, pela paciência e pelo apoio, mesmo quando minhas reflexões acabam se transformando em mais um livro.

Nota sobre os exemplos

As reflexões e os exemplos deste livro se baseiam em mais de trinta anos de experiência profissional na área de tecnologia da informação. Eles vêm de situações encontradas em projetos e ambientes de trabalho variados, além de muitas conversas com product managers, CTOs, engineering managers e outros profissionais de diferentes empresas. Portanto, abrangem várias organizações e diferentes períodos da minha trajetória.

Para adequar essas situações ao propósito do livro, eu as modifiquei, generalizei ou recombinei. Os casos de uso, os dados numéricos e as métricas ilustram problemas recorrentes e o que aprendi com eles. São construções: nenhuma descreve um cliente, um produto ou uma organização reais, nem reflete os dados, as práticas ou as posições do meu empregador.

As análises, os métodos e as opiniões expressos neste livro são pessoais e representam apenas o meu ponto de vista.

Este livro foi escrito com IA?

Sim. Inteiramente em colaboração com a inteligência artificial.

Sei que essa pergunta ganhou uma importância especial. Hoje nos perguntamos se um artigo, um romance ou um livro foi escrito com IA. Há até quem passe textos por ferramentas de detecção, como se fosse indispensável separar o que veio de uma pessoa do que veio de uma máquina.

Prefiro ser transparente: a IA participou de todas as etapas deste livro, da pesquisa de fontes à redação, passando pela organização das ideias, pela verificação das referências e pela busca de pontos que eu poderia não ter considerado. Mas essa colaboração merece uma explicação, porque não revela, por si só, quem pensou o conteúdo do livro.

Durante muito tempo, um projeto desta dimensão poderia contar com pessoas encarregadas de pesquisar documentos, confrontar ideias, revisar, estruturar ou preparar o texto. Hoje, a IA me dá acesso a parte dessas capacidades. Ela executa tarefas rapidamente, sugere

formulações, aponta incoerências e às vezes faz perguntas que não me ocorreram. Permite que eu explore muito mais do que conseguiria sozinho, com o tempo de que disponho.

Para escrever este livro, usei um mecanismo de Contexto baseado nas tecnologias do ChatGPT, com o Notion para manter as informações ao longo do tempo. Assim, fontes, ideias, decisões e versões de trabalho podem ser armazenadas e recuperadas de uma conversa para outra. Já apresentei essa abordagem no meu primeiro livro, **Le Contexte**. Nele, explico como um sistema desse tipo permite que a IA se torne uma espécie de supersecretário particular: ela deixa de apenas responder a perguntas, mantém o histórico dos projetos e me ajuda a fazê-los avançar ao longo do tempo.

Existe, porém, uma distinção essencial: **a capacidade de executar tarefas não equivale à responsabilidade de compreender e escolher.** As perguntas que orientam este livro, a experiência profissional que as alimenta, as interpretações, as decisões e o que escolho defender ou deixar de lado são meus. A IA pode discordar de mim, questionar minhas suposições e me ajudar a formular melhor uma ideia. Ela não substitui meu julgamento nem minha responsabilidade como autor.

Também faço uma distinção entre este livro e obras nas quais a própria escrita é parte fundamental da criação. Quando lemos Shakespeare numa tradução de François--Victor Hugo, Cervantes ou Cizia Zykë no original, a voz, o ritmo e o estilo importam tanto quanto o que é contado. Numa obra traduzida, as escolhas do tradutor também fazem parte da experiência. O lugar que desejamos dar à IA nesse tipo de literatura levanta outras questões, que não pretendo resolver aqui.

Este livro contém uma parte ficcional, mas sua ambição principal é compreender uma transformação tecnológica e organizacional. A escrita deve ser agradável e clara; acima de tudo, está a serviço das ideias, dos fatos e das perguntas que desejo compartilhar.

A IA tornou este projeto possível numa escala que provavelmente não teria alcançado apenas com meus próprios recursos, paralelamente à minha atividade profissional. Ela me permite desenvolver projetos pessoais, como este livro, sem abrir mão da profundidade de reflexão que exigem.

E há algo de bastante coerente, afinal, em escrever um livro sobre as transformações provocadas pela IA trabalhando justamente dessa maneira.

Laurent Deséchalliers

Sumário

Nota sobre a tradução 3

Introdução 4

Sobre o autor 8

Do mesmo autor 10

Agradecimentos 11

Nota sobre os exemplos 12

Este livro foi escrito com IA? 13

Malorean Corp. 17

1 Malorean Corp. — Um dia fora do comum 18

2 A IA multiplicou a capacidade de desenvolvimento de software 27

3 Malorean Corp. — O dia seguinte 40

4 Novos modelos organizacionais estão surgindo 56

5 Malorean Corp. — O que podemos fazer e o que devemos fazer 70

6 Os papéis se misturam, as responsabilidades permanecem 88

7 Malorean Corp. — O que Alice sabia sem nunca ter escrito 103

8 Mais autonomia exige limites mais claros, não menos 126

9 Malorean Corp. — Duas decisões razoáveis 147

10 A IA fará desaparecer os níveis intermediários das empresas? 167

11 Malorean Corp. — O tempo recuperado 190

12 Conclusão 214

Malorean Corp.

1 — Malorean Corp. — Um dia fora do comum

A sede da Malorean Corp. ficava em Le Havre. Mas, desde a pandemia de Covid, aquele endereço já não dizia muita coisa sobre onde o trabalho acontecia. Os funcionários estavam espalhados pela França. Encontravam-se diante de câmeras, conversavam com a tela compartilhada e, na maior parte do tempo, ninguém precisava estar na mesma sala para fazer um projeto avançar.

A Malorean Corp. desenvolvia um ERP acessível pela internet: um daqueles sistemas que acabam presentes em quase todas as operações de uma empresa. Um cliente podia ser uma microempresa, com poucos funcionários e necessidades muito práticas. Outro podia ser um grande grupo, com processos complexos, exigências específicas e pouca paciência para imprecisões.

A clientela era principalmente francesa. Havia alguns usuários na Alemanha e na Espanha, mas sobretudo em filiais de grupos que já eram clientes na França. Hubert Félix Duca, o diretor da empresa, tinha ambições

maiores. Gostaria de ver a Malorean Corp. se expandir de verdade pela Europa.

A dificuldade é que um ERP não atravessa fronteiras com a mesma facilidade de um site institucional. As regras mudam de um país para outro. As práticas de folha de pagamento, as obrigações administrativas e a maneira como as empresas trabalham também. Hubert conhecia bem essas restrições. Vinha da gestão, da contabilidade e das finanças, não do desenvolvimento de software. Para ele, uma funcionalidade só tinha valor se os clientes conseguissem usá-la sem se meter em dificuldades.

Dentro da empresa, os papéis eram relativamente claros. Duas equipes de cinco desenvolvedores cuidavam da evolução do software. Alice e Bob, os dois Product Managers, trabalhavam com elas para entender as necessidades, escolher as melhorias úteis e dar uma direção ao produto. Alice tinha sido desenvolvedora: conseguia participar das discussões técnicas sem perder o contexto. Bob vinha de uma escola de negócios e abordava as questões de outro modo. Tinham formações diferentes, mas trabalhavam muito bem juntos.

A equipe comercial procurava novos clientes e cultivava os relacionamentos existentes, com o apoio de uma pessoa de marketing. Cinco especialistas atuavam na

pré-venda e depois nas implantações, presencialmente ou à distância. Não se limitavam a instalar o ERP: ajudavam as empresas a esclarecer suas necessidades, adaptar os processos e muitas vezes continuavam acompanhando-as durante os seis meses seguintes à venda.

A equipe de suporte tinha doze pessoas. Sete atendiam as contas pequenas; cinco, as grandes. Muitas vezes eram elas as primeiras a ouvir que determinada funcionalidade obrigava um cliente a contornar o sistema, em vez de utilizá-lo como deveria. Dois profissionais do jurídico cuidavam dos assuntos da Malorean Corp., mas também contribuíam com seu conhecimento para a concepção do produto, especialmente quando alguma mudança envolvia regras sensíveis, como as de folha de pagamento.

Era assim que a empresa funcionava. Não perfeitamente, claro. Um software usado por tantos clientes acumula pedidos, exceções, hábitos difíceis de mudar e funcionalidades mantidas apenas porque ninguém teve tempo de refazê-las. Mas cada um sabia, mais ou menos, o que precisava fazer.

Então o ChatGPT começou a aparecer nas conversas.

No início, não havia nenhum grande programa de transformação. Alguns meses antes, duas pessoas especialmente curiosas tinham experimentado a ferramenta: Julie Morel, da área comercial, e Lucas Martin, do desenvolvimento. Começaram com usos próprios e depois contaram aos colegas o que estavam conseguindo fazer.

No caso de Julie, a vantagem era evidente. Ela preparava um e-mail delicado, pedia uma formulação mais clara, reorganizava anotações antes de uma reunião ou transformava uma série de ideias em roteiro de apresentação. Às vezes, preparava também o conteúdo de alguns slides. Não era o ChatGPT que decidia o que dizer ao cliente. Ela fornecia o conteúdo, revisava as sugestões, corrigia as frases e só enviava aquilo pelo qual estava disposta a responder.

Julie não dizia que a ferramenta conhecia seus clientes melhor do que ela. Estava simplesmente feliz por não gastar mais meia hora organizando algo que já sabia.

Philippe Garnier, seu colega, observava tudo de longe. Vendedor experiente, acostumado a grandes contas, conhecia as pessoas por trás dos contratos, os desacordos que nunca apareciam nas atas e as frases que mudavam o sentido de uma reunião.

— Você deveria experimentar com seus e-mails — Julie lhe dissera certa manhã. — Nem que seja só para organizar o texto.

— Eu sei escrever meus próprios e-mails.

Não falou por mal. Simplesmente não via por que entregar a uma ferramenta parte de um trabalho cujo valor, para ele, estava justamente no conhecimento dos clientes e na maneira de conversar com eles.

Julie não insistiu. Tinha uma reunião para preparar.

Lucas usava o ChatGPT de um jeito completamente diferente. Desenvolvedor experiente, morando em Lyon, não procurava apenas economizar alguns minutos em tarefas repetitivas. Passava muito tempo experimentando, fazendo perguntas cada vez mais exigentes, recomeçando quando uma resposta não o convencia. Enquanto Julie queria sobretudo um resultado útil, Lucas procurava descobrir até onde podia chegar.

Os relatos dos primeiros usuários acabaram chegando a Hubert. Por cerca de vinte euros por pessoa por mês, a conta parecia simples: se todos recuperassem ao menos um pouco de tempo nas tarefas tediosas, o investimento se pagaria com folga. Por isso, a Malorean Corp. decidiu disponibilizar o ChatGPT para toda a empresa.

Não foi uma revolução anunciada com uma enxurrada de apresentações. Cada pessoa podia usar a ferramenta quando visse vantagem. Alguns a abriam todos os dias; outros, raramente; Philippe, nunca. Para a maioria, a IA continuava sendo uma assistente: a pessoa explicava o que queria, a ferramenta sugeria, alguém conferia e depois decidia.

A vida da empresa continuou.

Na quarta-feira, durante uma conversa entre as equipes de Produto e Desenvolvimento, um assunto antigo voltou à mesa: a gestão de ausências.

Ninguém gostava muito daquele módulo. Era antigo, rígido e oferecia poucas opções. Funcionava suficientemente bem para continuar sendo usado, mas não tão bem a ponto de acabar com as reclamações.

Um funcionário que quisesse marcar suas férias precisava usar o computador de trabalho. Não havia um jeito simples de fazer isso pelo celular, num domingo à noite, enquanto pensava na viagem. E o cálculo das ausências tinha um defeito ainda mais incômodo: o módulo não considerava corretamente os feriados. Num pedido de segunda a sexta-feira, se a quarta fosse feriado, ele podia descontar cinco dias em vez de quatro.

O suporte conhecia o problema. A equipe comercial também. Para alguns clientes, o módulo já era uma fonte recorrente de irritação. Continuavam usando porque ele existia, não porque atendesse de fato às expectativas.

— Precisamos refazer isso direito — alguém resumiu.

Ninguém discordou. Mas “refazer direito” significava rever a interface, corrigir as regras de cálculo, tornar o processo utilizável no celular e substituir parte daquilo que havia envelhecido com o tempo.

Falava-se em algo como dois meses de trabalho. Havia outras prioridades. Mais uma vez, o assunto foi parar na lista do que precisaria ser feito.

Na tarde de sexta-feira, no fim de uma videochamada, Lucas pediu alguns minutos.

— Tenho uma coisa para mostrar.

Não anunciou o assunto. Ninguém sabia que ele vinha trabalhando no módulo de ausências desde a conversa de quarta.

Sua tela apareceu na reunião. Abriu-se uma interface clara, moderna, com as informações essenciais exatamente onde se esperava encontrá-las.

Lucas mostrou como registrar um pedido de ausência. Depois diminuiu a janela até a largura de um celular. A

interface se adaptou. Era possível continuar, escolher as datas e confirmar o pedido sem voltar a um computador.

Ele passou para um segundo exemplo: uma semana de segunda a sexta, com feriado na quarta.

O resultado mostrava quatro dias.

Lucas alterou as datas. O cálculo foi atualizado. Voltou à visualização de celular, percorreu algumas telas e continuou demonstrando. Não era uma sequência de imagens estáticas: os formulários respondiam, os cálculos eram feitos, o fluxo funcionava.

No começo, alguns colegas fizeram perguntas comuns: onde ficava determinada opção, o que acontecia se alguém mudasse uma data? Depois, as perguntas cessaram.

Nas pequenas janelas da videochamada, as expressões mudaram.

Na quarta, tinham falado em dois meses de trabalho. Na sexta, Lucas lhes mostrava uma primeira versão funcional. Não havia pedido ajuda. Não tinha avisado ninguém. E, por enquanto, não explicava como conseguira aquilo.

— Espera aí — alguém disse enfim. — Esse é mesmo o módulo de que falamos anteontem?

Lucas confirmou.

Fez-se silêncio.

Nas telas, todos olhavam para a mesma coisa: um trabalho que imaginavam precisar planejar durante semanas, surgido em dois dias.

Ninguém entendia como aquilo tinha sido possível.

2 — A IA multiplicou a capacidade de desenvolvimento de software

Durante décadas, grande parte da organização do desenvolvimento de software se estruturou em torno de uma restrição simples: **produzir software custa caro**. Transformar uma ideia em produto exigia tempo, competências escassas e muita coordenação. Era preciso definir o escopo de uma funcionalidade, dividi-la em tarefas, estabelecer prioridades, desenvolver, testar, revisar e implantar. O número de desenvolvedores disponíveis, sua experiência e o tempo que podiam dedicar a um assunto limitavam, na prática, o que uma empresa conseguia realizar.

Essa escassez moldou as organizações. Aprendemos a proteger o tempo dos desenvolvedores, preparar o trabalho antes de passá-lo à equipe, reduzir interrupções, organizar backlogs e tomar decisões muito antes de começar a construir. Boa parte da gestão de produto moderna se desenvolveu nesse contexto: era preciso evitar desperdiçar uma capacidade cara de Engenharia com uma ideia ruim.

A chegada dos assistentes de programação e, depois, dos agentes capazes de executar tarefas mais amplas começa a mudar essa equação.

A mudança vai muito além de um autocompletar melhor

As primeiras ferramentas de IA para desenvolvedores podiam ser vistas como uma nova geração de ferramentas de produtividade: completar uma função, explicar um trecho de código, escrever um teste ou sugerir uma refatoração.

A geração seguinte muda drasticamente a natureza do trabalho. Um agente pode receber uma tarefa, explorar uma base de código, modificar vários arquivos, executar testes, corrigir certos erros e apresentar uma alteração pronta para revisão. Um desenvolvedor pode, portanto, delegar grande parte da execução e, principalmente, colocar muito mais trabalho em andamento do que antes.

A consequência não é apenas que “o desenvolvedor programa mais rápido”.

A quantidade de trabalho de software que uma pessoa pode iniciar e fazer avançar aumenta drasticamente.

Algumas empresas começaram a divulgar resultados impressionantes. A AMPECO, ao descrever sua própria

transição para um modelo de Engenharia nativo em IA, relata uma velocidade de desenvolvimento quatro vezes maior e 50% menos bugs por story. Esses números não devem ser tratados como média do mercado: são resultados declarados por uma empresa, num contexto específico. Ainda assim, ilustram a dimensão da mudança buscada. Não se trata apenas de economizar alguns minutos ao escrever uma função, mas de ampliar muito a capacidade de execução de uma equipe. [1]

Os resultados impressionantes divulgados por uma empresa não são o único indício disponível. Eu mesmo vivi essa mudança com o Claude Code. Em quatro dias e mais de 180 commits, construí uma primeira versão de produto com telas, APIs e testes. Com meus métodos anteriores, estimo que teria levado um ou dois meses. Relatei a experiência no **Malorean.net** em julho de 2026. Evidentemente, isso não permite generalizar um ganho de produtividade: mostra o que um PM com experiência prévia em desenvolvimento já consegue realizar, mas também as dificuldades de qualidade, manutenção e controle do código gerado. [7]

Esse ponto muda a dimensão da pergunta.

Se a IA trouxesse apenas alguns pontos percentuais de ganho na programação, as organizações poderiam

continuar iguais e absorver a melhoria. Quando a capacidade de construir aumenta muito mais rapidamente, as outras restrições ganham importância relativa.

Acelerar a escrita do código não basta para acelerar a empresa

Uma equipe de software não é uma linha de produção cuja única etapa seja escrever código.

Antes de construir, é preciso compreender o problema, decidir se vale a pena investir nele, escolher uma direção e esclarecer as restrições. Depois, é preciso revisar, testar, garantir a segurança, implantar, observar e manter. Ao longo de todo o processo, escolhas precisam ser feitas.

Se apenas uma dessas etapas acelera muito, ela simplesmente envia mais trabalho para as outras.

O código pode ser produzido mais rapidamente sem que a compreensão dos clientes avance no mesmo ritmo. Os protótipos podem se multiplicar sem que melhore a capacidade de decidir quais merecem ser desenvolvidos. Mais alterações podem chegar para revisão sem que a atenção humana disponível para avaliá-las aumente na mesma proporção.

Isso não significa que a IA torne as demais atividades inúteis de repente.

Significa que ela **desloca os gargalos**.

O que antes ficava escondido pela lentidão da implementação passa a ser mais visível: a qualidade do problema escolhido, a clareza da intenção, as decisões, a validação, a coerência e a capacidade de aprender depois da entrega.

Produzir mais não significa criar mais valor

Essa distinção se torna central.

Quando desenvolver era difícil, a quantidade de software entregue podia parecer uma medida razoável da capacidade de uma organização de criar valor. Essa simplificação já era imperfeita. Com a IA, torna-se perigosa.

Agora, uma equipe pode gerar mais código, mais pull requests, mais protótipos e mais funcionalidades sem melhorar o produto. Pode até piorar seu desempenho geral se esse volume adicional trouxer mais defeitos, retrabalho, custos ou complexidade de manutenção.

Em outras palavras:

ser capaz de construir não é a mesma coisa que saber o que merece ser construído.

É por isso que as métricas de atividade se tornam ainda menos suficientes. A quantidade de alterações, as linhas de código ou o simples uso de uma ferramenta de IA não mostram se uma equipe criou mais valor.

A Canva oferece um exemplo interessante dessa mudança de foco para o impacto: a empresa desenvolveu um sistema que permite às equipes de Produto, Engenharia e Dados estimar o impacto comercial dos experimentos e relacionar mais diretamente decisões locais às consequências para o negócio. [2]

A pergunta muda.

Antes:

Quanto conseguimos construir?

Agora:

Entre tudo o que conseguimos construir, o que realmente vale a pena fazer?

> ***O problema da produtividade em software tem mais de quarenta anos***
>
> *Em 1986, Fred Brooks apresentou **No Silver Bullet**, publicado na **IEEE Computer** em 1987. Ele distinguiu as dificuldades **acidentais** de desenvolver software — linguagens, ferramentas, tradução de um projeto em código — da complexidade **essencial**: compreender,*

> *conceber, integrar e fazer um sistema evoluir. Na visão de Brooks, nenhum avanço técnico ou de gestão, isoladamente, prometia multiplicar por dez, no prazo de uma década, a produtividade, a confiabilidade e a simplicidade do desenvolvimento. [3]*
>
> *O ensaio foi incluído na edição comemorativa de 1995 de **The Mythical Man-Month**, publicado originalmente em 1975. [4] A IA generativa pode mudar a escala da execução, mas não garante que o problema escolhido seja o correto. Essa é uma interpretação atual das ideias de Brooks, **não uma previsão que ele tenha feito sobre agentes de IA**.*

O novo gargalo é o discernimento

À medida que implementar soluções fica mais barato, algumas capacidades ganham valor relativo: conhecer o cliente, compreender o problema, dominar o contexto estratégico, formular uma intenção clara, avaliar riscos, ter sensibilidade para o que faz um bom produto, saber dizer não e assumir a responsabilidade pelas decisões.

Num artigo publicado no **Malorean.net** em março de 2026, defendi essa ideia: conforme as ferramentas técnicas se tornam mais acessíveis, a vantagem competitiva se desloca para a compreensão do contexto

— áreas de negócio, usuários, clientes, restrições técnicas, concorrentes e regulamentações. É uma análise estratégica, não um resultado experimental. Ela ajuda a entender o que significa exercer bom julgamento: conhecer suficientemente uma situação para escolher a coisa certa a construir, em vez de apenas saber como construí-la. [8]

A IA torna o “como” mais acessível.

Ela não torna o “porquê” automaticamente evidente.

Permite produzir uma solução antes de a organização ter terminado de compreender o problema. Permite atender rapidamente a um pedido de cliente que a empresa deveria ter reconhecido como incompatível com seu mercado. Permite gerar várias versões de uma funcionalidade sem saber qual realmente produzirá um resultado melhor.

Quanto mais fácil fica construir, mais importante se torna a disciplina de escolher o que não construir.

A Ramp ilustra bem esse princípio. Seu CPO ressalta a necessidade de definir os perfis de cliente ideais prioritários (ICPs), os problemas que a empresa pretende resolver especialmente bem e a importância de recusar

certos pedidos — até os lucrativos — quando desviam o produto para outra direção. [5]

Em julho de 2026, Marty Cagan chamou esse descompasso de **paradoxo da produtividade da IA**. Sua análise é que construir mais rápido não resolve o problema das ideias que nunca deveriam ter sido construídas. Ele distingue aquilo que produzimos para aprender daquilo que entregamos para alcançar um resultado. Trata-se da análise de um profissional de Produto, não de uma medição que demonstre que todas as empresas são afetadas da mesma maneira. [6]

O paradoxo da IA, portanto, é este:

ela aumenta muito nossa capacidade de agir e, por consequência, torna a escolha de nossas ações mais importante do que nunca.

Uma transformação do sistema, não apenas da ferramenta

Uma resposta possível seria simplesmente oferecer um assistente de programação a cada desenvolvedor e pedir às equipes que continuassem trabalhando como antes.

Isso gera ganhos locais.

Mas, à medida que esses ganhos aumentam, a organização esbarra em seus outros limites: sua

capacidade de compreender, decidir, coordenar, conferir e absorver o que é produzido.

Nesse ponto, a questão já não é apenas adotar uma ferramenta.

Ela passa a ser organizacional.

Uma organização criada quando a capacidade de Engenharia era seu recurso mais escasso não tem por que continuar sendo a melhor opção quando essa escassez diminui.

Uma pergunta então se torna inevitável:

o que acontece com uma organização criada para proteger um recurso escasso — a capacidade de Engenharia — quando esse recurso se torna muito mais abundante?

Bibliografia

[1] How we built an AI-native engineering system — AMPECO

- URL: **https://www.ampeco.com/blog/how-we-built--an-ai-native-engineering-system/**
- Relato da reorganização de uma equipe de Engenharia em torno de agentes, com aumento expressivo de capacidade declarado pela empresa.

[2] Measuring Commercial Impact at Scale — Canva

- URL: **https://www.canva.dev/blog/engineering/measuring-commerical-impact-at-scale/**
- Descreve como a Canva relaciona experimentos de Produto e Engenharia a estimativas acessíveis de impacto comercial.

[3] No Silver Bullet: Essence and Accidents of Software Engineering — Fred Brooks

- URL: **https://www.cgl.ucsf.edu/Outreach/pc204/NoSilverBullet**
- Ensaio apresentado em 1986 e publicado em 1987, que distingue complexidade essencial e dificuldades acidentais do desenvolvimento de software.

[4] The Mythical Man-Month, edição comemorativa, capítulo 16 — Fred Brooks

- URL: **https://www.oreilly.com/library/view/mythical--man-month-the/0201835959/ch16.xhtml**
- Sumário da edição de 1995 que confirma a inclusão posterior de **No Silver Bullet**; apenas uma prévia do capítulo.

[5] 5 insights from Ramp's CPO on building for hypergrowth — Ramp

- URL: **https://ramp.com/blog/5-insights-from--ramps-cpo-reddit-ama**
- O CPO da Ramp destaca o foco nos ICPs e nos problemas prioritários, além da necessidade de recusar pedidos que dispersam o produto.

[6] The AI Productivity Paradox — Marty Cagan, Silicon Valley Product Group (2026)

- URL: **https://www.svpg.com/the-ai-productivity--paradox/**
- Análise de julho de 2026 que separa velocidade de implementação e resultados; reflexão profissional, não demonstração causal.

[7] Quatre jours de vibe coding dans la peau d'un PM rouillé — Laurent Deséchalliers, **Malorean.net** (2026)

- URL: **https://malorean.net/articles/2026-07-28--quatre-jours-de-vibe-coding-dans-la-peau-d-un--pm-rouille.pt-br.html**
- Relato pessoal com Claude Code: velocidade, dificuldades técnicas, testes e qualidade; não é estudo controlado nem prova generalizável.

[8] Dans le logiciel, l’avantage ne sera plus la technologie. Ce sera la compréhension du contexte — Laurent Deséchalliers, **Malorean.net** (2026)

- URL: **https://malorean.net/articles/2026-03-23--dans-le-logiciel-lavantage-ne-sera-plus-la--technologie-ce-sera-la-comprehension-du--contexte.pt-br.html**
- Ensaio estratégico do autor sobre a importância crescente do contexto de clientes, negócio, regulamentação e tecnologia; não é demonstração quantitativa.

3 — Malorean Corp. — O dia seguinte

— Usei a IA.

Lucas disse aquilo com tanta naturalidade que parecia acreditar que as três palavras explicavam tudo o que os outros tinham acabado de ver.

Ele ainda compartilhava a tela. O novo módulo de gestão de ausências ocupava quase toda a janela. Poucos minutos antes, havia mostrado como pedir uma ausência pelo celular e como o sistema levava os feriados em conta nos cálculos.

— Espera aí — Alice retomou. — Quando você diz que usou a IA, quer dizer que ela ajudou a escrever o código?

— Sim. Quer dizer, não só isso. Usei para conceber o módulo, entender o código que já existia, desenvolver as telas, fazer os cálculos, escrever os testes... Para praticamente tudo, na verdade.

— E você fez isso desde quarta-feira?

— Desde quarta à tarde. Terminei os últimos detalhes hoje de manhã.

Alice observou os rostos na tela. Alguns sorriam. Outros pareciam tentar entender o que tinham deixado passar.

— Mas você não tinha especificações — ela observou.

— Não. A gente tinha conversado sobre isso na quarta. Eu conhecia o módulo e seus problemas. Comecei pedindo à IA que analisasse o que existia. Depois fomos avançando aos poucos.

— E tudo funciona?

Lucas soltou uma risadinha.

— Tudo, eu não sei. O que mostrei a vocês funciona. Com certeza ainda tem coisas para conferir.

Thomas, que trabalhava na mesma equipe, falou pela primeira vez.

— E como ficou o código por trás disso?

— Pode olhar. Escrevi bastante teste.

— Vou olhar — Thomas respondeu.

Ele não parecia especialmente impressionado. Ou talvez estivesse, mas não quisesse demonstrar.

Hubert Félix Duca permanecera calado até então.

Já tinha assistido a demonstrações suficientes para saber que um software que funciona durante dez minutos não está necessariamente pronto para ser usado por centenas de empresas. Conhecia as exceções, as necessidades específicas dos grandes clientes e os problemas que só aparecem quando o sistema chega às mãos de quem vai usá-lo.

Mas também se lembrava do que ouvira na quarta-feira.

Dois meses.

E agora tinha diante dos olhos algo muito parecido com o que haviam imaginado.

— Lucas — perguntou —, pode me lembrar quanto tempo você gastou nisso?

— Uns dois dias.

Hubert se inclinou um pouco em direção à tela.

— Certo.

Não encontrou mais nada a dizer.

A reunião terminou alguns minutos depois. Todos voltaram ao trabalho, mas ninguém falou muito sobre as tarefas previstas para aquela tarde.

No sábado de manhã, Alice voltou a pensar na demonstração enquanto preparava seu café.

O que a incomodava não era apenas a velocidade de Lucas. Era o fato de que ele não tinha precisado de ninguém.

Ela conhecia bem seu trabalho como Product Manager. Antes de iniciar um desenvolvimento, procurava compreender o problema, comparar os pedidos,

conversar com clientes e trabalhar com os desenvolvedores para encontrar uma solução razoável.

Num assunto como o módulo de ausências, provavelmente teria organizado várias conversas. Ajudaria a esclarecer as regras, decidir como tratar os casos especiais e definir o que desenvolver primeiro.

Lucas não tinha feito nada disso.

Ouvira uma conversa, tomara a iniciativa e construíra uma solução.

Parte dela estava admirada. Outra parte se perguntava o que aconteceria se todos os desenvolvedores começassem a trabalhar assim.

No entanto, seu próprio dia a dia também havia mudado muito nos últimos meses.

No início, Alice usava a IA para resumir atas ou reformular documentos. Depois começou a confiar à ferramenta tarefas maiores. Automatizou parte do processamento do feedback dos clientes, a preparação de resumos, o agrupamento de pedidos semelhantes e várias tarefas administrativas que antes consumiam horas.

Agora, às vezes terminava em minutos um trabalho que antes ocupava boa parte da manhã.

Não sentia falta de nenhuma dessas tarefas.

Muito pelo contrário.

Finalmente tinha mais tempo para conversar com clientes, compreender os problemas por trás dos pedidos e refletir sobre o que realmente deveria ser construído.

A IA não havia tornado seu trabalho menos interessante.

Tinha eliminado parte do que a impedia de fazê-lo bem.

Então por que os desenvolvedores deveriam continuar trabalhando exatamente como antes?

Alice pousou a xícara.

A pergunta parecia óbvia.

A resposta, nem tanto.

8

Hubert também passara boa parte do fim de semana pensando.

Sua empresa sempre tivera mais ideias do que capacidade para realizá-las. Funcionalidades esperavam havia meses. Algumas melhorias solicitadas pelos clientes eram adiadas repetidamente. E havia o projeto de expansão pela Europa, que exigiria ainda muito trabalho no software.

Ele conhecia bem o obstáculo: não havia tempo suficiente para fazer tudo.

Na sexta-feira, Lucas talvez tivesse mostrado que esse obstáculo já não era tão intransponível.

Se um projeto estimado em dois meses podia gerar uma primeira versão em dois dias, quantos outros projetos de repente passavam a ser viáveis?

Dez desenvolvedores trabalhavam na Malorean Corp.

Hubert tentou imaginar o que aconteceria se todos avançassem naquela velocidade.

Depois imaginou dez desenvolvedores lançando, cada um por conta própria, novas funcionalidades sem coordenação.

Gostou bem menos dessa segunda imagem.

Chegou a pensar em ligar para Alice no sábado. Desistiu.

No domingo à noite, apenas lhe enviou uma mensagem.

Você tem um tempo amanhã de manhã? Queria conversar sobre sexta-feira.

A resposta veio alguns minutos depois.

Sim. Eu também.

Na segunda-feira, às nove da manhã, Alice e Hubert se encontraram por videochamada.

— Então? — Hubert perguntou.

— Então o quê?

— Você entendeu o que aconteceu na sexta?

Alice sorriu.

— Tecnicamente, mais ou menos. Lucas usou IA para desenvolver muito mais rápido do que teríamos imaginado.

— E o resto?

— Esse é justamente o problema.

Hubert assentiu.

— Passei o fim de semana pensando nisso. Faz anos que nos falta capacidade. Recusamos pedidos, adiamos projetos, passamos a vida escolhendo o que não vamos fazer. E aí Lucas entrega em dois dias algo que estimamos em dois meses.

— Uma primeira versão — Alice ressaltou. — Ainda não sabemos se está pronta para produção.

— Sim, eu sei. Mas, mesmo que leve mais duas semanas para terminar, muda muita coisa.

Alice não podia negar.

— O que me incomoda — Hubert continuou — é que ele fez tudo isso sem passar por você.

Alice demorou um instante a responder.

— Também me incomodou — admitiu.

— Você acha que devemos lembrá-lo de como trabalhamos?

— Pensei nisso.

— E?

— Acho que seria um erro.

Hubert franziu a testa.

— Mas, se ele começar a decidir sozinho o que desenvolver, podemos ter algumas surpresas.

— Claro. Mas olha o que aconteceu. Na quarta dissemos que era preciso refazer o módulo de ausências. Se tivéssemos seguido nosso processo habitual, eu começaria a levantar necessidades, organizaríamos reuniões, prepararíamos o assunto, dividiríamos o trabalho... Provavelmente passaríamos semanas antes de mostrar qualquer coisa.

— Isso não é necessariamente tempo perdido.

— Não. E não estou dizendo que devemos parar de compreender as necessidades. Mas Lucas mostrou que consegue avançar sem esperar por tudo isso.

— Então você está sugerindo que os desenvolvedores decidam sozinhos o que construir?

— Não, justamente o contrário.

Alice pensou por alguns segundos.

— Acho que estamos misturando duas coisas: decidir **o que** devemos fazer e decidir **como** fazer.

— As duas estão ligadas.

— Sim. Mas isso não significa que eu precise participar de cada decisão.

Ela ajeitou a postura na cadeira.

— Sabe, meu trabalho mudou enormemente nos últimos meses. Uso IA todos os dias. Automatizei um monte de coisas que fazia manualmente. Passo muito menos tempo preparando documentos ou organizando informações.

— Eu sei. Você já me contou.

— Mas talvez não tenha percebido o quanto isso mudou meu trabalho. Hoje tenho muito mais tempo para o que realmente importa: compreender os clientes, identificar problemas e decidir onde investir nossa energia.

Hubert escutava.

— Antes, boa parte do meu trabalho era fazer os assuntos avançarem dentro da empresa. Preparar tickets, esclarecer pedidos, responder perguntas, coordenar conversas. E se amanhã os desenvolvedores precisassem muito menos de mim para isso?

— Você teria mais tempo.

— Exatamente. E eles também.

Ela fez uma pausa.

— Aceitamos que a IA transforme meu trabalho. Por que iríamos querer que o trabalho dos desenvolvedores permanecesse igual?

Hubert recostou-se na cadeira.

— Está bem. Mas alguém ainda precisa decidir o que vamos construir.

— Claro. E, acima de tudo, por que vamos construir.

— Então o que você propõe?

Alice hesitou.

Não tinha preparado uma apresentação. Não tinha um modelo organizacional para propor, nem um processo novo pronto para implantar.

Tinha apenas uma intuição.

— Estou pensando se não deveríamos experimentar uma coisa.

Hubert esperou que ela continuasse.

— Poderíamos dar um problema para Lucas resolver. Um problema real de cliente. E deixar muito mais liberdade para ele descobrir como.

— Sem especificações?

— Sem entregar uma solução pronta. Explicamos o problema, damos acesso às pessoas que o conhecem e deixamos que ele investigue.

— E você?

— Continuo por perto. Ajudo a compreender as necessidades, questiono as ideias dele e evito que ele desenvolva algo só porque consegue. Mas não determino cada passo.

Hubert refletiu.

— Então você quer que ele faça parte do trabalho de Product Manager?

— Talvez. Ou talvez a gente simplesmente devesse parar de dividir sistematicamente cada pedaço de trabalho entre nós.

— E se não der certo?

— Vamos descobrir.

Hubert sorriu.

— Não parece um método muito elaborado.

— Eu sei.

Alice também sorriu.

— Mas prefiro tentar com um problema limitado a passar três meses discutindo uma nova organização.

Hubert não respondeu de imediato.

Depois pareceu se lembrar de algo.

— Por acaso temos um problema que alguns clientes da indústria alimentícia vêm mencionando há semanas.

Alice assentiu. Sabia exatamente qual.

O ERP da Malorean Corp. gerenciava estoques e operações de fabricação, mas não permitia acompanhar corretamente quais lotes de matérias-primas tinham sido usados para fabricar determinados produtos acabados.

Quando um fornecedor avisava a um fabricante de alimentos que um lote de matéria-prima estava defeituoso, a empresa precisava identificar rapidamente os produtos feitos com aquele lote.

Na situação atual, os clientes da Malorean Corp. precisavam reconstruir essas informações por conta própria.

— Recebi mais um retorno sobre isso na semana passada — Hubert explicou. — Um cliente disse que passou um dia inteiro com o responsável pela qualidade tentando encontrar os produtos afetados por um alerta do fornecedor.

— Sim. O suporte nos trouxe vários casos parecidos.

— Você acha que Lucas poderia trabalhar nisso?

Alice refletiu.

O assunto era mais delicado do que o módulo de ausências. Não bastava criar algumas telas. Era preciso entender o que acontecia de verdade nas fábricas, como as matérias-primas eram utilizadas e como os clientes identificavam seus produtos.

Mas era justamente isso que a interessava.

— Acho que podemos confiar o problema a ele, sim.

— Você acha que ele conhece suficientemente esse setor?

— Provavelmente não.

Hubert ergueu as sobrancelhas.

— Que tranquilizador.

— Nem eu conheço todos os detalhes. Mas sei com quem conversar. Temos clientes que conhecem suas restrições muito bem. Temos pessoas no suporte e na implantação que veem as dificuldades todos os dias. Lucas pode falar com elas.

— Você quer que ele fale diretamente com os clientes?

— Por que não?

— Normalmente, esse é o seu trabalho.

— Justamente.

Hubert ainda não parecia totalmente convencido.

— E se ele construir algo que não atenda à necessidade?

— Meu papel é ajudá-lo a evitar isso. Mas não escrevendo uma especificação de cinquenta páginas.

— E se ele quiser colocar a solução em produção?

— Ela terá de ser revisada e aprovada, como qualquer outra mudança no software. Dar mais liberdade não significa permitir que ele assuma qualquer risco.

Hubert ficou alguns instantes em silêncio.

— Sabe o que me preocupa? — perguntou, enfim. — Não é apenas a possibilidade de ele cometer um erro. É que a gente deixe de saber quem faz o quê.

— Isso também me preocupa.

— E mesmo assim você quer tentar?

— Quero.

Ela pensou um pouco antes de acrescentar:

— Porque a alternativa é continuar exatamente como antes, apenas pedindo aos desenvolvedores que programem mais rápido. Não tenho certeza de que foi isso que Lucas nos mostrou na sexta.

Conversaram por mais uns vinte minutos.

Hubert queria evitar interromper os projetos em andamento. Alice não queria que o experimento virasse

uma grande iniciativa anunciada para a empresa inteira antes mesmo de começar.

Então decidiram manter as coisas simples.

Lucas trabalharia no problema de rastreabilidade dos lotes. Teria liberdade para explorar soluções e conversar diretamente com as pessoas capazes de ajudá-lo a compreender a necessidade.

Alice seria sua referência em Produto. Não prepararia um backlog detalhado, mas continuaria envolvida nas decisões importantes.

Começariam por um escopo limitado, sem prometer data de entrega a nenhum cliente.

E, principalmente, reservariam tempo para observar o que aconteceria.

— Ainda precisamos definir como saberemos se o experimento deu certo — Hubert observou.

— Sim. E não quero medir apenas quantas funcionalidades ele construiu.

— Então o que você quer medir?

— Primeiro, se ele consegue entender o problema e propor algo útil. Depois, até onde consegue avançar sem precisarmos organizar tudo por ele.

— E quanto tempo isso exige de nós.

— Isso também.

Hubert olhou as horas.

— Certo. Vamos tentar.

Alice sorriu, embora não se sentisse totalmente tranquila.

— Vamos tentar.

— Você fala com ele hoje?

— Sim.

A videochamada terminou.

Alice ficou alguns instantes olhando para a tela. Abriu o aplicativo de mensagens e procurou Lucas.

Começou a escrever:

Oi, Lucas. Você tem um tempinho hoje? Gostaria de propor um assunto um pouco diferente da maneira como costumamos trabalhar.

Leu a mensagem novamente antes de enviá-la.

Pela primeira vez em muito tempo, estava prestes a confiar um problema a um desenvolvedor sem ter imaginado antes, em detalhes, a solução.

Ainda não sabia se era uma boa ideia.

E já não tinha tanta certeza de que seu papel fosse saber disso antes dele.

4 — Novos modelos organizacionais estão surgindo

A IA permite executar mais trabalho. Mas como as empresas estão reorganizando equipes e decisões diante dessa mudança? As respostas são diferentes.

Algumas reforçam a Gestão de Produto ao redor dos desenvolvedores. Outras ampliam o papel dos profissionais de Engenharia, dão mais recursos às áreas não técnicas ou reorganizam o trabalho em torno de equipes pequenas apoiadas por agentes.

A divisão tradicional do trabalho deixa de ser algo óbvio.

O modelo tradicional tinha sua lógica: Produto concentrava grande parte do contexto dos clientes e do negócio, Design materializava a experiência e Engenharia transformava essas intenções em software. Tickets, especificações, passagens de responsabilidade e reuniões permitiam sincronizar essas áreas diante de uma capacidade limitada de desenvolvimento.

Essa organização não se tornou ruim de uma hora para outra.

Mas, quando a capacidade de executar aumenta, várias empresas começam a distribuir de outro modo as pessoas, o contexto e o trabalho.

Portanto, ainda não se trata de buscar “o modelo certo”. Trata-se de observar as principais respostas que estão surgindo.

Reforçar Produto ao redor dos desenvolvedores

A resposta mais intuitiva é manter uma separação relativamente tradicional entre Produto e Engenharia, aumentando a capacidade de Gestão de Produto disponível para os desenvolvedores.

A Gamma oferece um exemplo interessante. A empresa relata uma proporção de aproximadamente um Product Manager para quatro engenheiros, enquanto a proporção histórica frequentemente citada é de um para dez. [1]

Nesse modelo, o crescimento da capacidade de Engenharia vem acompanhado de uma presença maior de Product Managers.

A ideia é simples: se mais coisas podem ser construídas, dedica-se mais capacidade a compreender, priorizar e orientar esse trabalho.

Dar aos desenvolvedores uma participação maior no problema

Outra estratégia consiste em ampliar o papel do profissional de Engenharia.

O desenvolvedor já não espera necessariamente que um problema seja totalmente transformado em solução e depois em tickets. Pode participar mais da compreensão da necessidade, da prototipação, das decisões locais e do acompanhamento dos resultados.

Numa vaga de Product Engineer, a Linear descreve as responsabilidades esperadas: conduzir uma funcionalidade da ideia ao produto entregue, participando da definição do problema sem depender continuamente de Gestão de Produto. [2]

Esse modelo busca reduzir algumas passagens entre áreas ao reunir mais contexto e capacidade de agir na mesma pessoa.

Não significa que todo desenvolvedor vire PM.

Descreve, antes, uma ampliação do papel de Engenharia em determinados assuntos.

Permitir que as áreas ultrapassem antigas fronteiras

O movimento não diz respeito apenas aos desenvolvedores.

Num estudo sobre a evolução das funções, a Figma relata que 64% dos participantes se identificam com pelo menos dois papéis e mais de um terço afirma ter responsabilidades que atravessam diferentes profissões. Esse resultado não pode ser atribuído exclusivamente à IA, mas retrata fronteiras entre papéis menos rígidas. [3]

A Atlassian também documenta fluxos em que designers trabalham diretamente no código real do produto e abrem uma pull request antes da revisão de Engenharia. [4]

A Cursor, por sua vez, descreve uma organização em que Produto, Design e Engenharia participam mais diretamente da construção. [5]

Portanto, a conclusão não é tanto que “todo mundo faz tudo”, mas que **mais pessoas atuam fora dos limites tradicionais de suas profissões**.

Essa sobreposição traz outra questão: se várias áreas conseguem agir no mesmo terreno, como manter responsabilidades claras?

Voltaremos a isso.

Confiar mais trabalho a equipes pequenas e responsáveis

Outra família de respostas concentra um escopo maior de atuação em menos pessoas.

A AMPECO descreve reuniões de alinhamento em que um engenheiro e um Product Owner definem a intenção antes que os agentes assumam grande parte da análise, do desenvolvimento e dos testes. [6]

A Taito descreve um funcionamento em que a equipe aprova um plano e depois uma ou duas pessoas conduzem a construção com agentes. [7]

Nesses modelos, um pequeno grupo humano consegue colocar muito mais trabalho em movimento do que antes.

A mudança organizacional observável, portanto, é a possível redução de passagens entre áreas e a concentração de um assunto numa pessoa ou num grupo pequeno.

Resta definir o que essas pessoas podem decidir sozinhas e quando precisam consultar alguém responsável por um domínio.

Automatizar parte das verificações

Uma produção mais rápida também pressiona os mecanismos de validação.

Algumas empresas respondem automatizando mais controles ou tratando mudanças de maneiras diferentes conforme o risco.

A Meta utiliza o Diff Risk Score para estimar o risco de uma alteração provocar um incidente. [8]

A Cloudflare formaliza padrões de Engenharia que podem ser verificados automaticamente antes do merge. [9]

A Taito também combina regras determinísticas, agentes de revisão e gatilhos associados ao risco. [7]

O ponto importante, por enquanto, é apenas que **a validação não precisa mais ocorrer da mesma forma para todas as mudanças**.

Mais adiante, examinaremos como combinar automação, responsabilidade humana e limites para escalonamento.

Oferecer contexto em vez de prescrever cada etapa

Outras empresas vão além: questionam o ticket detalhado como principal ferramenta de coordenação do trabalho.

A Linear critica explicitamente o modelo em que o PM delimita um problema, escreve tudo num sistema e espera que Engenharia execute. A empresa insiste na necessidade de tornar feedback dos clientes, ideias,

direção estratégica, decisões e código acessíveis tanto às pessoas quanto aos agentes. [10]

A Cursor também ressalta a importância de objetivos suficientemente claros para permitir mais iniciativa local. [5]

A mudança observada, portanto, não é o desaparecimento de toda coordenação.

É o deslocamento de parte dessa coordenação de instruções detalhadas para **um contexto mais rico e diretamente acessível**.

Um capítulo específico tratará do que esse contexto deve conter, como mantê-lo e como definir os limites da autonomia.

Permitir que quem conhece a necessidade construa

Algumas empresas chegam a permitir que as equipes que conhecem a necessidade criem pequenas ferramentas por conta própria, sem precisar entrar sempre na fila dos desenvolvedores.

A Shopify criou o Quick, uma plataforma interna que permite a mais funcionários publicar sites e ferramentas pequenas rapidamente. [11]

A Vercel descreve um modelo no qual funcionários não técnicos constroem ferramentas internas num ambiente

seguro, sem desviar os desenvolvedores das prioridades do produto principal. [12]

A Atlassian também mostra designers atuando no código antes da revisão técnica. [4]

Nesses casos, a IA não torna apenas os desenvolvedores mais rápidos.

Às vezes, ela permite deslocar a execução para **a pessoa que já conhece o contexto da necessidade**.

Um princípio anterior à IA

Em 2018, Marty Cagan já defendia o que chamava de **empowered product teams**: equipes multidisciplinares às quais a liderança confia problemas a resolver, em vez de listas de funcionalidades a entregar. Elas devem buscar soluções reunindo competências de Produto, Design e Engenharia, respondendo pelos resultados, não pelo volume produzido. [13]

Com Chris Jones, ele aprofundaria essa abordagem no livro **EMPOWERED**, de 2020. O modelo não anunciava profissões intercambiáveis nem o fim dos gestores: ao contrário, pressupunha visão compartilhada, competências sólidas e líderes que apoiam o desenvolvimento das equipes. [14]

Essa ideia, portanto, não surgiu com os agentes. O que a IA potencialmente muda é a velocidade e a facilidade com que uma equipe autônoma pode executar certas soluções que decidiu investigar. **A autonomia de Produto é uma ambição mais antiga; a IA muda seus meios e suas restrições.**

Escolhas que ainda não convergiram

Essas respostas não são etapas de uma transformação única. Reforçar Produto e dar aos desenvolvedores mais responsabilidades de Produto podem até parecer contraditórios. Ainda assim, podem funcionar em empresas diferentes ou coexistir na mesma organização.

O ponto comum está em outra parte: **quando executar se torna mais acessível, a distribuição de contexto, decisões e responsabilidades volta a ser uma escolha.**

Ainda precisamos ver o que essa mudança provoca no cotidiano das pessoas envolvidas, para além dos anúncios e dos organogramas.

Voltaremos em breve à Malorean Corp. para observar como esses modelos mudam a vida de uma pequena empresa francesa. Depois examinaremos as

responsabilidades e regras necessárias a essa autonomia.

Bibliografia

[1] The bottleneck keeps shifting: What AI is changing about how we build — Atlassian / Gamma

- URL: **https://www.atlassian.com/blog/how-we-build/the-bottleneck-keeps-shifting-what-ai-is-changing--about-how-we-build**
- Descreve a ampliação da presença de Gestão de Produto na Gamma à medida que a capacidade de Engenharia aumenta.

[2] Product Engineer — Linear Careers

- URL: **https://linear.app/careers/0c7c2e26-0a98-42cf-a47c-9a3999fb513b**
- Vaga que apresenta as responsabilidades de um Product Engineer, da definição do problema à entrega.

[3] Are Roles and Responsibilities a Thing of the Past? — Figma

- URL: **https://www.figma.com/blog/2025-shifting--roles-report/**
- Documenta a crescente sobreposição de funções e a ampliação das responsabilidades para além das fronteiras tradicionais.

[4] Designers’ workflow for shipping code — Atlassian

- URL: **https://www.atlassian.com/blog/ai-at-work/designers-workflow-for-shipping-code**
- Descreve designers alterando diretamente código de produto antes da revisão de Engenharia.

[5] How Cursor Is Navigating the New Shape of Product Craft — Atlassian / Cursor

- URL: **https://www.atlassian.com/blog/how-we-build/how-cursor-is-navigating-the-new-shape-of--product-craft**
- Mostra a aproximação entre Produto, Design e Engenharia na construção de soluções.

[6] How we built an AI-native engineering system — AMPECO

- URL: **https://www.ampeco.com/blog/how-we-built--an-ai-native-engineering-system/**
- Apresenta um funcionamento no qual poucas pessoas definem a intenção antes de delegar parte expressiva da execução aos agentes.

[7] Inside an AI-native engineering organization — Swarmia / **Taito.ai**

- URL: **https://www.swarmia.com/blog/inside-an-ai--native-engineering-organization/**

- Descreve uma equipe pequena conduzindo agentes, com boa parte da execução coordenada por uma ou duas pessoas.

[8] Diff Risk Score: AI-driven risk-aware software development — Meta

- URL: **https://engineering.fb.com/2025/08/06/developer-tools/diff-risk-score-drs-ai-risk-aware--software-development-meta/**
- Apresenta uma pontuação de risco que permite adaptar o fluxo de validação à criticidade provável das alterações.

[9] How Cloudflare enforces engineering standards using AI — Cloudflare

- URL: **https://blog.cloudflare.com/engineering--standards-enforcement/**
- Mostra padrões explícitos de Engenharia transformados em verificações automáticas.

[10] Issue tracking is dead — Linear

- URL: **https://linear.app/next**
- Defende compartilhar mais contexto entre feedback, estratégia, decisões e código, para além dos tickets isolados.

[11] Quick: An internal hosting platform for the AI era — Shopify

- URL: **https://shopify.engineering/quick**
- Apresenta uma plataforma interna para que mais funcionários construam e publiquem ferramentas diretamente.

[12] Stopping the slow death of internal tools — Vercel

- URL: **https://vercel.com/blog/stopping-the-slow--death-of-internal-tools**
- Mostra usuários de negócio criando ferramentas internas sem tirar a equipe de Engenharia das prioridades centrais.

[13] Empowered Product Teams — Marty Cagan, Silicon Valley Product Group (2018)

- URL: **https://www.svpg.com/empowered-product--teams/**
- Texto de referência que distingue equipes responsáveis por resolver problemas de equipes limitadas à entrega de funcionalidades.

[14] EMPOWERED: Ordinary People, Extraordinary Products — Marty Cagan e Chris Jones (2020)

- URL: **https://www.svpg.com/books/empowered--ordinary-people-extraordinary-products/**

- Apresentação do livro sobre equipes autônomas, objetivos, contexto estratégico e liderança.

5 — Malorean Corp. — O que podemos fazer e o que devemos fazer

Na terça-feira de manhã, Lucas começou fazendo uma pergunta que surpreendeu Alice.

— Posso falar com o cliente?

No dia anterior, ele havia recebido os poucos relatórios que o suporte guardava sobre o problema de rastreabilidade dos lotes. As mensagens falavam de números de fabricação, matérias-primas, planilhas do Excel e alertas de fornecedores. Lucas passara a noite organizando tudo com a ajuda da IA.

— Claro — Alice respondeu. — Por quê?

— Porque ainda não entendi exatamente do que eles precisam.

Alice sorriu. Uma semana antes, provavelmente teria conduzido essa conversa antes de transmitir suas conclusões ao desenvolvimento. Agora tinha prometido deixar Lucas assumir a iniciativa.

— Vou apresentar você à pessoa do suporte que acompanha essa conta. Ela pode organizar a conversa.

Na quarta-feira, Lucas descobriu uma situação muito mais complexa do que a expressão “gerenciar lotes” sugeria.

O responsável pela qualidade do cliente mostrou uma carta recebida de um fornecedor. Era preciso separar uma matéria-prima. O fornecedor sabia o número do lote. O fabricante sabia quais pedidos tinham sido recebidos. Mas, para identificar com precisão os produtos acabados afetados, as equipes precisavam cruzar vários documentos.

— Da última vez — explicou o responsável pela qualidade —, éramos quatro pessoas conferindo arquivos durante boa parte do dia. E, mesmo depois disso, tivemos de voltar à fábrica para ter certeza.

Lucas pediu para ver como as informações eram registradas no momento da fabricação. O cliente mostrou uma tela do ERP e, em seguida, uma planilha.

— Mas por que vocês também anotam esse número no Excel?

— Porque o sistema de vocês não permite encontrar o que precisamos.

A resposta não foi agressiva. Isso só a tornou mais difícil de ouvir.

Ao fim da chamada, Lucas tinha três páginas de anotações e menos certezas do que no começo.

— E então? — Alice perguntou quando ele ligou de volta.

— Eu estava pensando em uma tela para registrar os números dos lotes.

— E agora?

— Agora acho que primeiro precisamos entender como eles registram o que realmente consumiram. Senão vamos ter uma tela bonita e continuar sem saber responder à pergunta deles.

Alice assentiu. Poderia ter dito aquilo a ele na segunda-feira. Mas não estava descontente por Lucas ter descoberto sozinho.

Na sexta-feira seguinte, Lucas lhe mostrou uma primeira versão numa base de demonstração.

Era possível associar um lote de matéria-prima a uma fabricação e depois encontrar os produtos acabados resultantes dela. Ele também criara um pequeno exemplo para acompanhar um lote até os produtos envolvidos num alerta.

Alice clicou algumas vezes.

— Você fez isso nesta semana?

— Fiz. Ainda tem muita coisa errada ou faltando, mas já temos algo para mostrar ao cliente.

— E se também pudéssemos fazer a busca no sentido contrário?

— Já dá.

Ele abriu outra tela.

Alice caiu na gargalhada.

— Claro.

Desde a demonstração do módulo de ausências, sua percepção da dimensão dos projetos havia mudado. O que antes deixaria para o trimestre seguinte agora parecia estar ao alcance.

— Também poderíamos permitir uma busca por fornecedor. E colocar um alerta diretamente nos pedidos afetados. E por que não preparar os documentos de recolhimento?

Lucas não respondeu imediatamente.

— Podemos anotar essas ideias — disse, por fim. — Mas primeiro eu queria voltar ao cliente com o que temos.

— Não vamos esperar quinze dias para mostrar uma versão mais completa, vamos?

— Não é a demonstração que me preocupa.

— O que preocupa você?

Desta vez, Lucas compartilhou uma tela menos agradável de olhar. Tabelas antigas, campos com nomes difíceis de entender, várias ligações entre fabricação e estoque.

— Podemos criar quantas telas quisermos. O problema é que nosso ERP nunca foi pensado para guardar todo esse histórico dos lotes. Existem movimentações de estoque de vários anos atrás, às vezes corrigidas manualmente. Algumas informações simplesmente não existem.

— A IA não pode ajudar a colocar tudo isso em ordem?

— Pode me ajudar a escrever uma migração. Não pode inventar o que nunca foi registrado.

Alice ficou em silêncio.

— Se quisermos fazer isso direito — Lucas continuou —, teremos de modificar parte do modelo de dados e verificar as consequências nos outros módulos. Não podemos estragar o estoque de um cliente só para economizar duas semanas.

— Não estou pedindo para você quebrar o sistema.

— Eu sei. Mas desde sexta-feira, cada vez que mostro alguma coisa, você pede mais três.

A observação não tinha sido agressiva. Mesmo assim, atingiu Alice.

— É verdade — ela reconheceu.

— Eu também quero avançar rápido, você sabe. Mas não é porque consigo gerar código em uma hora que quinze anos de funcionamento do ERP se tornam simples.

Depois de hesitar, acrescentou:

— Thomas olhou parte do código do módulo de ausências. Encontrou dois casos que tínhamos tratado mal. Ele estava certo. Corrigimos, mas isso me lembrou de uma coisa: podemos construir muito mais rápido e ainda precisar de alguém que conheça as armadilhas.

Alice se lembrou de Thomas, na sexta da demonstração, pedindo simplesmente para ver o código.

— Então o que você propõe?

— Continuamos com a demonstração usando dados fictícios. Conferimos se entendemos a necessidade. Mas, antes de mexer nos estoques reais, quero dedicar tempo a compreender o sistema existente e fazer as verificações necessárias.

Alice olhou para a primeira versão, ainda aberta na outra tela.

Na semana anterior, defendera a autonomia de Lucas diante de Hubert. De repente, era menos agradável descobrir que essa autonomia incluía o direito de dizer não a ela.

— Está bem — disse finalmente. — Por enquanto, não mexemos nos dados reais.

Na segunda-feira seguinte, Hubert abriu a reunião semanal com uma pergunta que já se tornara habitual.

— Então, como vai nosso experimento?

— Está avançando — Alice respondeu. — Mais rápido do que esperávamos em alguns pontos. Mais devagar em outros.

— Como assim?

Alice mostrou a demonstração de Lucas e resumiu a conversa de sexta-feira.

Hubert prestou especial atenção às ressalvas técnicas.

— Ele faz bem em ter cuidado — disse. — Não quero resolver um problema de rastreabilidade criando problemas de estoque para todos os nossos clientes.

— Sim. E acho que pressionei um pouco.

— Você?

— Há duas semanas, tenho a impressão de que tudo passou a ser possível. Sempre que surge uma ideia, penso que podemos testá-la imediatamente.

Hubert sorriu.

— Foi mais ou menos assim que me senti depois da primeira demonstração.

— O problema é que fica fácil confundir a velocidade com que colocamos algo na tela com a velocidade com que entregamos algo confiável.

— Pelo menos Lucas sabe quando é hora de diminuir o ritmo.

— Sim. E preciso confiar nele quanto a isso.

Dois dias depois, Julie Morel perguntou a Alice se ela tinha cinco minutos.

Alice conhecia aquela frase. Com Julie, cinco minutos facilmente viravam meia hora.

— Encontrei um cliente potencial importante na semana passada — Julie começou. — Eles têm várias fábricas e procuram um ERP que se adapte à maneira como trabalham. Olha o que preparei.

Ela compartilhou uma série de protótipos de telas muito bem acabados. Havia diferentes etapas de aprovação conforme a categoria de pedido, regras específicas para cada unidade e telas que mudavam de acordo com o usuário.

— Foi você que fez isso? — Alice perguntou.

— Com o ChatGPT. Peguei minhas anotações da reunião, pedi que sugerisse um fluxo e depois fomos ajustando. Gastei uma noite.

Alice percebeu imediatamente a utilidade daquelas imagens. Em poucas horas, Julie havia dado uma forma concreta a uma necessidade que até então só existia num relatório comercial.

— O cliente já viu as telas?

— Ainda não. Queria mostrar primeiro a você. Mas ele deixou bem claro que, se conseguíssemos fazer algo assim, a conversa poderia mudar bastante.

Alice percorreu as telas.

— A ideia é interessante. Você deveria falar diretamente com Lucas.

Julie a olhou, surpresa.

— Diretamente? Achei que você fosse preparar o assunto.

— Por quê? Você conhece o cliente, conversou com ele e até preparou as telas. Pode explicar o que ele procura melhor do que eu.

— E Lucas pode avaliar se é viável?

— Sim. Mas cuidado, Julie: avaliar a viabilidade não significa assumir o compromisso de desenvolver.

— Claro.

Alice conhecia aquele “claro”. Não era mentira. Era o entusiasmo de uma vendedora diante de uma possível venda.

No dia seguinte, as duas se reuniram com Lucas. Julie apresentou as telas, explicou os pedidos do cliente potencial e respondeu às perguntas.

Lucas logo se interessou pelas diferenças entre as regras de cada unidade.

— Já temos uma parte desse mecanismo no ERP — comentou. — Não para fazer exatamente isso, mas ele existe. Provavelmente daria para preparar alguma coisa e testar.

— Em quanto tempo? — Julie perguntou.

— Para uma demonstração? Talvez não demore muito.

Julie olhou para Alice. Seu sorriso era quase o de alguém que já tinha fechado o negócio.

— Se mostrarmos isso na semana que vem, pode fazer toda a diferença.

Alice sentiu a própria vontade de avançar tentando responder antes que ela pensasse direito. Lembrou-se do projeto de rastreabilidade, dos pedidos em espera, das regras particulares daquele cliente e de todos os outros clientes que talvez não precisassem de nada daquilo.

— Não — disse.

Os dois se viraram para ela.

— Não o quê? — Julie perguntou.

— Não vamos começar a desenvolver essa funcionalidade no nosso produto só para fechar esse contrato.

— Mas você acabou de dizer que era interessante.

— É interessante. Só que não sabemos se esse cliente precisa de uma funcionalidade que deveria existir para todos ou de uma maneira de trabalhar muito específica das fábricas dele.

— Podemos perder a venda.

— Talvez.

Julie claramente não esperava essa resposta.

— Finalmente temos condições de responder rápido a pedidos assim — insistiu. — Seria uma pena não aproveitar.

— Não quero deixar de aproveitar. Quero compreender o que estamos oferecendo. Se acrescentarmos cada pedido específico ao coração do ERP, acabaremos com um software impossível de evoluir.

— Mas Lucas disse que consegue fazer.

— Lucas disse que consegue preparar uma demonstração. Não é a mesma coisa.

Seguiu-se um silêncio. Julie olhou mais uma vez para suas telas.

— Então o que eu digo a ele?

— Que entendemos o problema e queremos mostrar uma possibilidade. Não que decidimos acrescentar a funcionalidade ao produto.

Julie assentiu. Parecia decepcionada, mas não desanimada.

— Certo. Posso marcar uma reunião para ele explicar melhor as diferenças entre as fábricas.

— Ótimo. E guarde os protótipos. Eles vão nos ajudar a fazer as perguntas certas.

Quando Julie saiu da chamada, Alice ficou com Lucas por alguns segundos.

— Você realmente conseguiria construir isso rápido?

— Uma primeira versão, sim.

— E teria vontade?

— Sinceramente? Sim. Parecia divertido.

Alice riu.

— Eu também queria dizer sim.

— O que fez você parar?

— Percebi que nós dois estávamos falando do que **podíamos** fazer e quase não falávamos do que **devíamos** fazer.

Na semana seguinte, Lucas enviou a Alice um link acompanhado de uma mensagem.

Montei uma coisinha ontem. Pode dar uma olhada?

Alice abriu a página. Reconheceu a interface do ERP, mas agora havia um pequeno painel lateral que permitia alterar determinadas regras. Era possível escolher uma unidade, mudar um limite de aprovação, mostrar ou ocultar uma etapa e executar a demonstração novamente.

Ela ligou para Lucas.

— O que é isso?

— Uma ferramenta pequena para preparar demonstrações com regras diferentes. Reaproveitei nosso mecanismo antigo de configuração. Está só numa instância de teste.

— Quanto tempo você gastou?

— Duas horas, talvez. Tive a ideia pensando nos protótipos da Julie.

— Mas nem é o projeto em que você está trabalhando!

— Não. Queria ver se dava para evitar mexer no código toda vez que quiséssemos testar uma variação.

Alice começou a mudar os parâmetros. A tela respondia. Tentou outra configuração, depois uma terceira.

— Espera. Você está dizendo que posso preparar sozinha várias versões de uma demonstração?

— Para as regras que disponibilizamos, sim. Não é qualquer coisa. E principalmente não nos ambientes dos clientes.

— Mas posso mostrar dois fluxos diferentes à Julie e perguntar qual corresponde de verdade ao que o cliente dela precisa.

— Exatamente.

Alice ficou alguns instantes olhando para a página. O primeiro módulo de ausências havia tornado Lucas mais rápido. Aquela ferramenta dava a **ela** uma possibilidade que não tinha na véspera.

Podia testar uma ideia sem pedir que um desenvolvedor a implementasse especialmente para ela.

— Você imagina o que podemos fazer com isso?

Lucas sorriu.

— Acho que sim. Mas, antes que você peça para transformar isso num produto completo, lembre-se de que foi uma gambiarra de duas horas.

— Prometo.

Ela desligou e voltou a brincar com os parâmetros.

Na segunda-feira seguinte, Hubert percebeu que Alice não havia aberto seu documento habitual de acompanhamento.

— Você não preparou nada hoje?

— Preparei. Mas queria mostrar uma coisa.

Ela apresentou a ferramenta de Lucas e criou diante dele duas variações da mesma demonstração.

Hubert entendeu rapidamente.

— Você não pediu aos desenvolvedores que fizessem essas mudanças?

— Não. Lucas criou uma ferramenta. Posso testar algumas regras sozinha.

— Poderíamos deixar a equipe comercial usar?

— Talvez mais adiante. Por enquanto, não é robusta o suficiente. Mas Julie já pode trabalhar comigo nos casos do cliente potencial.

Hubert examinou a tela com interesse.

— E o projeto dos lotes?

— Lucas está avançando. Voltou ao cliente com a primeira versão. Entendemos melhor o que precisa ser feito. Mas ele se recusa a mexer nos dados históricos até garantir que o tratamento seja seguro.

— Parece razoável.

— Sim. Na sexta isso me irritou. Hoje estou feliz por ele ter insistido.

Alice girou a caneta entre os dedos.

— Algo estranho está acontecendo, Hubert. Lucas conversa com os clientes e toma decisões sobre a solução. Julie aparece com protótipos quase utilizáveis. E eu agora posso testar certas variações do nosso ERP sem precisar de desenvolvimento.

— Isso preocupa você?

— Às vezes. Fico pensando no que acontece com nossas profissões quando todos começam a fazer uma parte do trabalho dos outros.

— E encontrou uma resposta?

— Ainda não.

Ela refletiu.

— Só sei que não quero ser a pessoa por quem toda ideia tem de passar. Julie não tinha motivo para esperar

que eu transformasse suas telas em tickets. Lucas não precisa que eu lhe diga como construir cada tela.

— Mas você recusou desenvolver o pedido do cliente potencial.

— Sim. Porque aquela era uma decisão sobre nosso produto. Posso ter me enganado, ainda posso mudar de ideia, mas cabia a mim assumir essa escolha.

— E Lucas se recusou a tomar um atalho com os estoques.

— Porque ele sabe o que corremos o risco de quebrar. Nesse assunto, preciso ouvir o julgamento dele. Mesmo quando estou com pressa.

Hubert observou por um instante a pequena janela com o rosto de Alice.

— Então podemos deixar as pessoas fazerem mais coisas, mas não podemos deixar todas as decisões sem um responsável claro.

— É o que estou começando a pensar.

Ela hesitou e acrescentou:

— Antes, às vezes eu sentia que meu trabalho era importante porque muitas coisas passavam por mim. Agora me pergunto se o mais importante não é o que acontece no final. Estamos resolvendo o problema certo?

Estamos construindo algo confiável? Vale a pena fazer isso?

Hubert assentiu.

— E, se ninguém souber quem deve responder a essas perguntas, teremos um problema.

— Sim.

Ficaram em silêncio por alguns segundos. Nenhum dos dois queria transformar a conversa no anúncio de uma nova organização.

O experimento mal havia começado. Trouxera mais possibilidades, mas também mais decisões a tomar.

Alice recebeu uma notificação. Era Julie: o cliente potencial aceitara ver as diferentes versões da demonstração. Também queria saber quando poderiam ficar disponíveis para suas equipes.

Alice não respondeu imediatamente.

O software agora permitia criar respostas muito mais rapidamente.

Mas alguém ainda precisava decidir quais a Malorean Corp. estava disposta a prometer — e a assumir.

6 — Os papéis se misturam, as responsabilidades permanecem

Durante muito tempo, as fronteiras entre Produto, Design e Engenharia serviram como pontos de referência simples.

Produto compreendia o problema e decidia o que construir. Design concebia a experiência. Engenharia transformava essa intenção em software. Na prática, essa divisão nunca foi tão nítida, mas ainda assim estruturava as equipes, as contratações, os processos e as responsabilidades.

Essa distribuição já vinha sendo questionada antes da IA: as equipes autônomas de Produto descritas no capítulo 1.4 reuniam Produto, Design e Engenharia em torno de um problema comum. Agora, a IA acelera a travessia de outra fronteira: a da execução, quando cada pessoa consegue realizar mais tarefas antes reservadas a outra especialidade.

O panorama das organizações mostrou essa evolução, especialmente por meio do estudo da Figma. [1] No Malorean, também descrevi o movimento cruzado dos desenvolvedores em direção à compreensão dos

problemas e dos PMs em direção à construção de soluções. [2]

Seria tentador concluir que os papéis vão desaparecer. Acredito que essa mudança exige, antes, reformular a responsabilidade de cada um: **quanto mais flexíveis as fronteiras entre papéis, mais explícitas devem ser as responsabilidades.**

A capacidade de agir já não pertence a uma única profissão

O que a IA redistribui primeiro não é necessariamente o poder de decisão. É a capacidade de agir.

Pense nos designers que modificam o código por conta própria, uma prática documentada pela Atlassian: uma equipe técnica continua avaliando a qualidade e as condições para colocar a mudança em produção. [3] Essa etapa não representa um retorno à antiga divisão. Mostra que poder intervir no código não é a mesma coisa que responder por sua confiabilidade.

A distinção também aparece na vaga de Product Engineer da Linear: a empresa espera que essa pessoa consiga conduzir um assunto, do problema inicial até o resultado entregue. [4] Isso não lhe confere

automaticamente autoridade sobre todas as decisões de estratégia de Produto.

Podemos, portanto, reduzir passagens de responsabilidade sem eliminar a especialização que justificava algumas delas. A pergunta já não é quantas profissões participam, mas **quem responde pelo quê quando várias áreas atuam no mesmo terreno**.

Poder contribuir não significa responder pelo domínio

Precisamos separar duas coisas que as organizações frequentemente confundem: a permissão de agir e a responsabilidade pelo resultado.

Um profissional de vendas pode construir uma ferramenta interna. Isso não o torna responsável pela segurança da plataforma.

Um PM pode modificar uma interface. Isso não significa que passe a responder pela coerência do Design System.

Um desenvolvedor pode decidir um comportamento local do produto. Isso não lhe dá automaticamente a responsabilidade pela estratégia, pelo posicionamento ou pelo roadmap geral.

Essa distinção já existe em muitas equipes, mas a IA obriga as organizações a formalizá-la.

Num funcionamento muito especializado, a profissão e a responsabilidade costumavam coincidir. Quem podia agir era também a pessoa à qual o domínio pertencia.
Quando a capacidade de agir se espalha, essa coincidência desaparece.
O risco, então, é confundir autonomia com diluição da responsabilidade.
Já defendi um princípio parecido sobre qualidade: os desenvolvedores não podem entregar software e depois considerar que os defeitos pertencem à equipe de QA, ao plantão de correções ou ao próximo desenvolvedor disponível. A qualidade técnica do que foi entregue continua sendo uma responsabilidade identificável. [5]
A sobreposição dos papéis não muda esse princípio. Torna-o ainda mais importante.

O conceito de Guardião
É para enfrentar esse problema que utilizo o conceito de **Guardião**.
Em cada domínio importante, uma pessoa ou equipe continua responsável por sua coerência e qualidade.
O Guardião não é necessariamente quem executa todo o trabalho daquele domínio.

Também não é quem deve aprovar cada pequena decisão.

É a pessoa a quem a organização recorre quando uma escolha importante afeta o domínio, quando um limite é ultrapassado ou quando algo dá errado.

Numa organização simples, podemos imaginar pelo menos três responsabilidades explícitas:

- **Guardião de Produto**: coerência do Produto, escolhas estruturantes, público-alvo, proposta de valor, grandes decisões e relação com a estratégia.
- **Guardião de Design**: Design System, coerência da experiência, acessibilidade, novos padrões e exigência de qualidade em UX.
- **Guardião de Engenharia**: arquitetura, segurança, desempenho, facilidade de manutenção, qualidade técnica e condições para colocar em produção.

O princípio é simples:

qualquer pessoa pode contribuir em várias áreas, mas cada domínio continua tendo alguém que responde por ele.

Esse modelo permite a sobreposição dos papéis sem fazer as responsabilidades desaparecerem.

Vendas podem construir sem se tornarem responsáveis pela qualidade técnica

Vamos a um exemplo deliberadamente simples.

Uma pessoa da equipe comercial identifica uma necessidade, usa uma ferramenta apoiada por IA e cria um pequeno aplicativo ou uma alteração útil. Desde que a empresa tenha um ambiente suficientemente seguro, não há razão para considerar que só Engenharia tenha o direito de construir.

Mas essa liberdade não significa que a qualidade técnica passe a ser responsabilidade de Vendas.

Se a mudança envolve segurança, permissões, dados sensíveis ou arquitetura, o Guardião de Engenharia precisa poder solicitar uma revisão, exigir uma correção ou bloquear a alteração.

A pessoa de Vendas continua responsável pela necessidade.

Engenharia continua respondendo pelo que a empresa aceita manter e operar tecnicamente.

A mesma lógica vale para Design.

Um PM pode montar uma tela a partir de componentes conhecidos. Desde que permaneça dentro do Design System e de padrões já dominados, exigir uma aprovação

de Design para tudo reduziria a autonomia sem necessidade.

Por outro lado, se esse PM sair do Design System, inventar um novo padrão ou conceber uma interação complexa e estruturante, o Guardião de Design deve poder intervir.

A fronteira deixa de ser: “quem tem direito de mexer em quê?”

Passa a ser: **“a partir de que nível de risco ou novidade o responsável por um domínio deve intervir?”**

O responsável pela iniciativa e o Guardião do domínio não são a mesma pessoa

Essa talvez seja uma das distinções mais importantes das novas organizações.

O **responsável pela iniciativa** conduz um assunto.

Precisa compreender o problema, fazer o trabalho avançar, tomar decisões locais, coordenar o necessário, verificar os resultados e assumir a responsabilidade de levá-lo até o fim.

O **Guardião**, por sua vez, tem uma responsabilidade mais transversal.

Não é necessariamente dono da iniciativa. Responde por um nível de coerência, qualidade ou risco dentro de um domínio.

Um Product Engineer pode ser responsável por uma funcionalidade sem passar a responder por toda a arquitetura da empresa.

Um PM pode conduzir um lançamento sem se tornar responsável por todo o Design.

Uma pessoa de Vendas pode conduzir o desenvolvimento de uma ferramenta de negócio sem responder pela segurança de todos os aplicativos internos.

Essa distinção esclarece a responsabilidade pelo resultado sem exigir que uma única pessoa domine todas as especialidades.

A AMPECO ilustra uma versão desse movimento: um Product Owner e um engenheiro se alinham sobre “o quê” e “por quê”; depois, agentes assumem boa parte da análise, do desenvolvimento e dos testes. A execução se distribui, mas um ser humano permanece responsável pela validação final. [6]

A IA pode, portanto, reduzir o número de pessoas necessárias para executar uma iniciativa e, ao mesmo

tempo, aumentar a necessidade de saber com clareza **quem responde por ela**.

O cargo talvez já não seja a melhor unidade para compreender uma organização

Essa evolução também levanta uma questão mais ampla: ainda faz sentido organizar a empresa principalmente por títulos de cargo?

O debate sobre o Product Owner é revelador. No Malorean, defendi a ideia de que PO é, antes de tudo, uma função — um mandato de decisão e responsabilidade — e não uma profissão cuja razão de existir seja manter um backlog. [7]

Essa distinção ganha ainda mais utilidade com a IA.

Se definimos um papel pela lista de tarefas que executa, a IA torna essa definição instável. Muitas tarefas passam a estar ao alcance de perfis diferentes.

Se definimos uma responsabilidade pelo que uma pessoa ou equipe deve assumir, ela resiste muito melhor.

O PM pode escrever menos especificações sem deixar de responder pela qualidade das escolhas de Produto.

O desenvolvedor pode escrever menos código manualmente sem deixar de responder pela qualidade técnica.

O designer pode deixar de ser a única pessoa a posicionar elementos na tela sem perder a responsabilidade pela coerência da experiência.

Na Cursor, onde as fronteiras entre Produto, Design e Desenvolvimento também estão menos rígidas, a liderança de Produto lembra que o essencial continua sendo escolher o que merece ser construído e dar direção suficiente para evitar desvios. [8]

A profissão muda menos porque perde todas as responsabilidades do que porque deixa de ter exclusividade sobre certas formas de execução.

O Guardião não pode virar uma nova burocracia

Existe, porém, uma armadilha evidente.

Se toda alteração de código feita por um PM precisar passar pela aprovação de Engenharia, se toda decisão local de um desenvolvedor tiver de voltar para o PM e se toda tela criada por alguém que não seja designer precisar esperar por um comitê de Design, não teremos resolvido nada.

Teremos apenas recriado as antigas passagens obrigatórias com outro nome.

O Guardião não pode se tornar um aprovador universal.

Sua existência serve justamente para permitir maior autonomia: cada pessoa pode agir dentro de regras conhecidas, sem precisar pedir autorização continuamente, porque a organização sabe onde estão as responsabilidades e em quais casos elas devem voltar a ser acionadas.

Na vida real, esse modelo nunca funciona perfeitamente. Mudanças chegam à produção que o Guardião não teria aprovado exatamente daquela forma. Decisões locais passam despercebidas. Às vezes as equipes tomam mais liberdade do que o previsto. Em outros momentos, algumas pessoas escalam decisões cedo demais, por cautela.

Isso não significa necessariamente que o modelo fracassou.

Uma organização realmente autônoma aceita que parte das decisões seja tomada sem inspeção prévia. Caso contrário, não existe autonomia.

O problema, portanto, não é garantir que jamais aconteça um desvio.

É definir **quais desvios são aceitáveis, quais precisam ser detectados pelo sistema e quais exigem a**

intervenção de um Guardião antes de se tornarem caros.

É aí que a pergunta muda.

Distinguimos duas responsabilidades: a de quem conduz uma iniciativa e a do Guardião de um domínio.

Falta resolver outra questão:

quando o responsável pela iniciativa deve decidir sozinho e quando precisa envolver o Guardião?

Na Malorean Corp., essa fronteira entre iniciativa e responsabilidade começa a provocar atritos. Voltaremos a eles no próximo episódio, antes de examinar salvaguardas e regras de escalonamento.

Bibliografia

[1] Are Roles and Responsibilities a Thing of the Past? — Figma

- URL: **https://www.figma.com/blog/2025-shifting--roles-report/**
- Analisa a ampliação das responsabilidades e a crescente sobreposição de papéis nas equipes modernas de produto.

[2] PM, développeurs et IA : les rôles se brouillent, les responsabilités restent — Malorean

- URL: **https://malorean.net/articles/2026-08-01-pm--developpeurs-et-ia-les-roles-se-brouillent-les-responsabilites-restent.pt-br.html**
- Reflexão do autor: a IA amplia quem pode agir, sem eliminar as responsabilidades de Produto, Design e Engenharia.

[3] Designers' workflow for shipping code — Atlassian

- URL: **https://www.atlassian.com/blog/ai-at-work/designers-workflow-for-shipping-code**
- Descreve designers modificando diretamente o código, com revisão de Engenharia para qualidade, desempenho e produção.

[4] Product Engineer — Linear Careers

- URL: **https://linear.app/careers/0c7c2e26-0a98-42cf-a47c-9a3999fb513b**
- Vaga que apresenta as responsabilidades de um Product Engineer, da definição do problema à entrega.

[5] La qualité appartient à ceux qui livrent — Malorean

- URL: **https://malorean.net/articles/2026-06-03-la--qualite-appartient-a-ceux-qui-livrent.pt-br.html**

- Defende uma responsabilidade técnica explícita: a equipe não pode transferir a qualidade a outro grupo depois da entrega.

[6] How we built an AI-native engineering system — AMPECO

- URL: **https://www.ampeco.com/blog/how-we-built--an-ai-native-engineering-system/**
- Mostra uma execução amplamente delegada a agentes, com intenção, arquitetura e validação sob responsabilidade humana.

[7] Le PO n’est pas un métier, c’est une fonction — et l’IA n’y change rien, elle accélère — Malorean

- URL: **https://malorean.net/articles/2026-08-03-le--po-nest-pas-un-metier-cest-une-fonction.pt-br.html**
- Diferencia um mandato de decisão e responsabilidade de uma profissão definida sobretudo pela administração de backlog.

[8] How Cursor Is Navigating the New Shape of Product Craft — Atlassian / Cursor

- URL: **https://www.atlassian.com/blog/how-we-build/how-cursor-is-navigating-the-new-shape-of--product-craft**

- Mostra fronteiras de papel mais flexíveis sem eliminar julgamento, direção e salvaguardas de Produto.

7 — Malorean Corp. — O que Alice sabia sem nunca ter escrito

Na quinta-feira de manhã, Lucas ligou para Alice pela terceira vez naquela semana.

— Prometo que são só dois minutos.

Alice aprendera a desconfiar daquela frase. Lucas nunca pedia reuniões longas. Chegava com uma pergunta específica, os dois conversavam e ele voltava a trabalhar. Era justamente isso que tornava difícil recusar aquelas interrupções.

— Pode falar.

— Na rastreabilidade dos lotes, o cliente quer poder pesquisar por fornecedor. Isso eu consigo acrescentar à demonstração. Mas, se o fornecedor entregou vários lotes com o mesmo número interno, mostramos tudo junto ou pedimos para o usuário escolher?

Alice pensou por alguns segundos.

— Não podemos dar a entender que identificamos um lote se vários correspondem à busca. Mostre os resultados separadamente. Depois podemos ver com o cliente como ele diferencia cada um.

— Foi o que pensei. Mas não tinha certeza.

— Então por que você está me ligando?

Lucas deu uma risada sem graça.

— Depois da história dos protótipos da Julie, prefiro conferir antes de decidir alguma coisa que pareça uma regra de Produto.

Na semana anterior, Alice havia pedido que ele não transformasse o pedido de um cliente potencial numa funcionalidade padrão. Agora lamentava que Lucas tivesse entendido que seria melhor perguntar a ela sempre que surgisse alguma dúvida.

— Certo. Mas, nesse caso, você não precisava de mim.

— Fico mais tranquilo ouvindo isso de você.

Desligaram. A chamada havia durado sete minutos. Alice voltou ao documento que estava escrevendo, releu a última frase e não conseguiu lembrar o que pretendia acrescentar.

Dez minutos depois, Lucas lhe enviou uma mensagem.

Outra pergunta: nas fabricações antigas, quando não há informação sobre os lotes, mostramos "desconhecido" ou escondemos a coluna?

Ela começou a digitar uma resposta. Então parou.

A pergunta era diferente, mas o raciocínio era o mesmo: nunca apresentar como certa uma informação que o ERP não possuía. Poderia ter explicado esse princípio em vez de decidir dois casos separadamente.

Mesmo assim, respondeu.

Deixe claro que a informação não existe. Não esconda o problema.

Lucas reagiu com um sinal de positivo.

Alice olhou a agenda. Uma conversa com o suporte tinha sido adiada. À tarde, tinha uma reunião com cliente. Entre uma e outra, precisava revisar as propostas de Julie para o cliente potencial com várias unidades.

Gostava de que Lucas falasse diretamente com ela. Era precisamente o que desejara ao lhe confiar o projeto. Mas ele avançava tão rápido que cada nova possibilidade trazia uma nova pergunta. E, quanto mais a consultava, menos tempo Alice tinha para fazer aquilo que estava em melhores condições de fazer: compreender os clientes e pensar na direção do produto.

Lucas também perdia tempo. Às vezes esperava sua resposta para continuar.

Uma ideia desagradável lhe passou pela cabeça: se aquilo continuasse, talvez acabassem colocando todas as

decisões de volta nos tickets. Seria mais fácil organizar. Ela especificaria, Lucas executaria e ninguém mais precisaria se perguntar até onde ia a autonomia dele. Exatamente aquilo que tinham tentado superar.

Ao meio-dia, Alice se encontrou com Bob por videochamada. Os dois haviam adquirido o hábito de conversar sem pauta, apenas para comparar o que estavam vivendo.

— Como vão as coisas com Lucas? — Bob perguntou.

— Até bem demais, de certa forma.

— Isso não parece um elogio.

— Ele avança muito rápido, enxerga um monte de coisas, conversa com os clientes. Mas vive me ligando para decidir coisas que poderia decidir sozinho.

— E você responde?

— Claro.

— Então ele faz bem em ligar.

Alice revirou os olhos.

— Obrigada, Bob.

— Só estou brincando pela metade. Se posso receber sua resposta em cinco minutos, em vez de correr o risco de decidir errado, por que não faria isso?

— Porque nunca vamos conseguir trabalhar assim.

Bob pediu um exemplo. Alice contou as duas perguntas da manhã, depois a conversa com Julie e uma dúvida recente de Lucas: deveriam acrescentar uma configuração pedida por um cliente se ela provavelmente não serviria a mais ninguém?

— Você percebe? — Alice disse. — São sempre variações da mesma pergunta.

— E qual é sua resposta?

— Depende.

Bob sorriu.

— Aí está. É exatamente esse tipo de resposta que faz as pessoas ligarem de novo.

Alice ia protestar. Mas ele tinha razão.

— Não é um “depende” sem critério — continuou. — Eu avalio se estamos resolvendo um problema recorrente, se atende nossos clientes prioritários, se vamos complicar o produto para todo mundo, se conseguiremos manter a solução…

Ela parou.

— Viu? — Bob disse. — Você acabou de me dar parte da regra.

— Não é uma regra. É mais um jeito de pensar.

— Melhor ainda. Talvez seja justamente isso que você deva transmitir a Lucas.

Alice ficou em silêncio.

Havia anos escrevia tickets. Descrevia comportamentos esperados, casos especiais, critérios de aceitação. Mas quase nunca registrava o raciocínio que lhe permitia escolher entre diferentes comportamentos possíveis. Essa lógica ficava em sua cabeça ou desaparecia numa reunião depois que o trabalho era entregue.

— Devo ter centenas de coisas assim — disse.

— Provavelmente. Eu também.

— E nunca tivemos motivo para escrevê-las.

— Tínhamos motivos. Não tínhamos tempo. E ninguém abria um ticket chamado “explicar por que às vezes recusamos uma boa ideia”.

Os dois riram.

Depois da chamada, Alice abriu um novo documento. Escreveu: **Princípios para decisões de Produto**.

Ela acrescentou uma primeira frase, apagou, tentou outra. Quinze minutos depois, tinha um parágrafo que parecia uma política interna. Era correto. Mas não tinha nada a ver com a maneira como explicara o assunto a Bob.

Fechou o documento.

Depois abriu o ChatGPT.

— Ajude-me a tirar da cabeça tudo o que sei sobre nosso produto — ditou. — Não quero um procedimento teórico. Faça perguntas, ajude-me a lembrar exemplos e a separar o que é realmente uma regra daquilo que é só um hábito meu.

A primeira resposta sugeriu começar pelas decisões recentes.

Alice falou.

Falou do cliente potencial de Julie, dos protótipos atraentes e do risco de transformar uma necessidade específica numa promessa feita a todos os clientes. Explicou que um grande contrato podia valer muito dinheiro sem representar automaticamente a direção certa para o produto.

— Espera, preciso explicar por quê — acrescentou em voz alta. — Nosso ERP atende empresas muito diferentes. Se incluirmos cada regra particular no coração do software, acabaremos complicando a vida de todos. Mas às vezes um pedido específico revela um problema muito mais geral. Também não podemos recusar tudo.

O ChatGPT perguntou como ela fazia essa distinção.

Alice descreveu as conversas com o suporte, os casos observados em vários clientes, os pedidos da equipe comercial e as restrições das implantações. Lembrou-se de discussões antigas das quais não restara nenhuma ata. Explicou por que algumas opções haviam sido abandonadas, por que uma configuração fora preferida a uma nova funcionalidade e por que uma exceção de folha de pagamento não podia ser tratada como uma preferência de interface.

Às vezes se surpreendia respondendo a uma pergunta que nunca pensara em formular explicitamente.

A IA reformulava, agrupava ideias, propunha um princípio. Alice corrigia.

— Não, não é “nunca desenvolvemos nada específico para um cliente”. Podemos fazer isso em determinadas condições, mas precisamos decidir conscientemente qual é o custo. E, sobretudo, não podemos tratar uma hipótese comercial como se fosse uma decisão de Produto.

Pouco depois, o ChatGPT escreveu que uma funcionalidade de rastreabilidade poderia garantir a reconstrução de todos os lotes históricos.

— De jeito nenhum — Alice disse à tela. — Se os dados nunca foram registrados, não podemos reconstruí-los por mágica. Isso é um limite que precisa ser mostrado, não uma capacidade que podemos prometer.

Pediu que o trecho fosse corrigido. Quis que o documento distinguisse o que já tinha sido decidido, o que ela considerava provável e o que ainda precisaria ser confirmado com o suporte, Lucas ou os clientes. Evitou incluir informações confidenciais das contas reais; os exemplos foram reformulados.

Passou uma hora. Depois duas.

O documento se enchia de coisas que Alice sabia havia muito tempo: os clientes que a Malorean queria atender prioritariamente, os problemas recorrentes, as concessões aceitas, os compromissos que não deveriam assumir, os limites dos dados antigos, os motivos pelos quais um protótipo não era uma funcionalidade pronta para vender.

Ela acrescentou links para as informações já disponíveis: feedback do suporte, decisões que ainda podiam ser encontradas, anotações de reuniões e documentação do produto. Sempre que faltasse comprovação, o texto deveria deixar isso claro.

Perto do fim da terceira hora, o ChatGPT sugeriu uma síntese.

Alice releu. Algumas páginas eram longas demais. Outras misturavam uma regra geral com um exemplo. Corrigiu mais uma vez as formulações categóricas demais, acrescentou uma data e delimitou o alcance do documento.

Então exportou tudo para PDF.

Percorreu as páginas, surpresa com a quantidade.

Três horas antes, pensava apenas em explicar a Lucas quando ele podia tomar uma decisão sem consultá-la. Agora tinha nas mãos um retrato imperfeito, mas surpreendentemente rico, daquilo que a equipe sabia sobre o produto sem nunca ter compartilhado sistematicamente.

Não era uma verdade definitiva. Algumas passagens ainda precisariam ser confirmadas ou atualizadas. O PDF não substituiria os clientes, os dados nem o julgamento das pessoas responsáveis.

Mas, pelo menos, a próxima pessoa não precisaria começar do zero.

Lucas abriu o arquivo na manhã seguinte.

— Você escreveu tudo isso para mim?

— Não sozinha — Alice respondeu. — Passei três horas conversando com o ChatGPT. Ele me ajudou a formular perguntas, organizar as respostas e perceber o que faltava. Eu revisei e corrigi. Mas já sabia a maior parte dessas coisas.

— Então por que eu nunca tinha visto nada disso?

Alice sorriu, um pouco sem graça.

— Porque essas coisas não cabiam em nenhum ticket.

Lucas percorreu o sumário. Encontrou princípios, exemplos de decisões, restrições do negócio, informações sobre clientes, links para as fontes e perguntas ainda abertas.

— Posso passar isso para minha IA quando trabalho na rastreabilidade?

— Sim, no nosso ambiente autorizado. Mas diga para usar como contexto, não como prova de que tudo é verdade. E, se uma regra for importante, voltamos à fonte ou à pessoa que responde por ela.

— Isso me pouparia um tempo enorme. Às vezes peço à IA que me ajude a escolher entre duas soluções, mas ela só conhece o que contei. Pode me dar uma resposta

muito convincente sem considerar uma decisão tomada três anos atrás.

Alice conhecia aquela sensação. Uma resposta razoável isoladamente podia ser absurda para o ERP deles.

Lucas parou num trecho.

— Aqui você diz que nunca devemos esconder a ausência de uma informação de negócio se isso puder dar a impressão de que o resultado está completo.

— Foi o que eu disse ontem.

— Sim. E, se tivesse lido isso ontem, não teria ligado.

Ele continuou a leitura. Depois levantou a cabeça.

— Certo, mas ainda tenho um problema.

Alice sentiu um breve desânimo.

— Qual?

— Agora entendo melhor como você raciocina. Continuo sem saber a partir de que ponto tenho o direito de decidir sozinho.

Ela pousou o café.

Ele acabara de encontrar a maior lacuna do documento.

Voltaram ao assunto juntos. Alice compartilhou uma página em branco.

— Vamos tentar uma coisa simples. Antes de me ligar, faça três perguntas: entendo o resultado que buscamos? Minha decisão está dentro dos limites que conhecemos? E o que pode dar errado se eu me enganar?

— Vamos precisar definir “dentro dos limites”.

— Exatamente.

Escreveram três níveis.

Verde: Lucas decidia sozinho quando a escolha era local, reversível, coerente com as regras conhecidas e não envolvia uma nova promessa ao cliente. Precisava conseguir explicar o raciocínio e verificar o resultado.

Laranja: ele consultava Alice quando encontrasse uma situação nova, uma contradição entre objetivos ou uma dúvida significativa sobre o comportamento do Produto. Se o risco fosse técnico, procurava também as pessoas competentes, como Thomas quando o assunto envolvia o histórico do ERP.

Vermelho: ninguém decidia sozinho se fosse necessário alterar uma regra estruturante, assumir um compromisso comercial em nome da Malorean, mexer com dados sensíveis ou aceitar um risco grande e difícil de reverter. Era preciso tomar uma decisão explícita com os responsáveis envolvidos e, às vezes, com Hubert.

Lucas releu a tabela.

— Então, se eu mudar a apresentação dos resultados na demonstração para evitar confusão, estou no verde.

— Sim.

— Se quiser acrescentar um comportamento que possa afetar todos os clientes, falo com você.

— Conversamos, sim. Principalmente se ainda não entendemos a necessidade.

— E se eu quiser modificar dados históricos para registrar os lotes?

— Não cabe só a mim autorizar. Precisamos da análise técnica e dos controles adequados. Você já me lembrou disso mais de uma vez.

Lucas sorriu.

— Só queria ter certeza de que você lembrava.

Acrescentaram uma regra: a falta de resposta de Alice jamais transformaria uma decisão laranja ou vermelha numa decisão verde. Por outro lado, ele também não deveria pedir autorização para cada decisão verde.

— E se eu errar a cor? — Lucas perguntou.

— Você me conta. Ajustamos os limites. Prefiro melhorar a regra a passar o tempo reclamando porque você não adivinhou o que eu estava pensando.

Lucas assentiu.

— Eu poderia até pedir à minha IA que confrontasse minhas escolhas com essa tabela, incluindo as razões e informações faltantes.

— Sim. Desde que ela não decida por nós se um risco é aceitável.

Alice acrescentou a tabela ao documento de referência. Começava a compreender a diferença entre transmitir informações e realmente dar a alguém condições de decidir.

Poderiam ter parado ali. Mas Lucas voltou à demonstração de rastreabilidade.

— Estamos falando da minha autonomia há uma hora. Vou fazer uma pergunta sobre o projeto em si: como saberemos se deu certo?

Alice respondeu quase automaticamente:

— Se o cliente conseguir encontrar os produtos fabricados com determinado lote.

— Sim, mas como vamos verificar? E qual é o nível de confiabilidade necessário?

Desta vez, ela não procurou responder de imediato.

Haviam passado dias explorando telas, regras e estruturas de dados. Tinham uma demonstração promissora. Mesmo assim, ninguém havia escrito com precisão como avaliar o resultado.

Alice se lembrou do responsável pela qualidade que passara boa parte de um dia cruzando arquivos depois de um alerta de fornecedor.

— Devemos começar por aí — disse. — Com o número de um lote sinalizado, ele consegue encontrar as fabricações afetadas sem refazer toda a pesquisa manualmente? Quanto tempo leva? O resultado está completo? E, quando não temos a informação, isso fica claro?

— Podemos testar com ele. Não apenas perguntar se gostou da demonstração.

— Exatamente.

Alice abriu outra página.

— Na verdade, deveríamos ter começado por isso antes de sair construindo.

Lucas caiu na gargalhada.

— Se você tivesse me pedido isso duas semanas atrás, eu diria que estava tentando transformar o projeto num ticket de novo.

— Talvez. Mas olha o que estamos escrevendo. Registraram o problema observado, os usuários envolvidos, o resultado esperado, como medi-lo, os limites dos dados, o que ficava fora do escopo e as decisões que ainda precisavam de aprovação. Uma página, nada mais.

Lucas sugeriu chamar aquilo de **Mini PRD**.

— Ninguém está me dizendo qual tela construir — ele observou. — Estamos entrando em acordo sobre o que tornará o trabalho útil.

— E sobre os limites que não podemos ultrapassar.

Alice pensou em voz alta:

— Para os próximos projetos, isso deveria ser nosso ponto de partida. Antes de perguntar o que vamos desenvolver, perguntamos qual resultado queremos e como saberemos se o alcançamos. Só depois delimitamos o escopo e decidimos o que vale a pena construir.

— E, se eu conseguir preencher esse documento com as informações certas, posso começar sem esperar que você prepare um backlog.

— Você pode propor a definição do escopo. Isso não significa que aprovaremos qualquer compromisso sem analisar. Mas sim, esse é o objetivo.

Completaram o Mini PRD de rastreabilidade deixando abertas as perguntas que ainda não sabiam responder. Precisavam conferir alguns casos com o cliente e confirmar restrições técnicas.

O documento não criava nenhuma falsa certeza. Dizia o que estavam tentando aprender.

Alice o considerou mais útil do que uma lista de funcionalidades para marcar como concluídas.

8

Na segunda-feira seguinte, a reunião semanal de Alice com Hubert foi diferente.

Ela não compartilhou nem o painel habitual de acompanhamento nem a lista de funcionalidades produzidas por Lucas.

Mostrou a tabela de decisões e depois o Mini PRD.

Hubert leu as primeiras linhas.

— Vocês passaram tanto tempo fazendo documentos, agora que Lucas trabalha muito mais rápido?

Alice conhecia a objeção. Tinha feito a mesma pergunta poucos dias antes.

— Acho justamente que não documentávamos o suficiente. Escrevíamos o que precisava ser construído, mas raramente as informações que permitiam escolher. Por isso Lucas me pedia decisões o tempo todo. Com isso, consegue tomar parte delas sem mim.

— E como você sabe que ele respeita os limites?

— Definimos quando pode decidir sozinho e quando deve nos consultar. No resto, não vamos fiscalizar tudo. Vamos verificar os resultados e corrigir os limites se for necessário.

Hubert parou na linha sobre os dados antigos.

— Vocês não podem garantir a rastreabilidade do que nunca foi registrado.

— Não — Alice respondeu. — E o documento diz isso. Precisamos deixar claro para o cliente.

Hubert aprovou.

— Prefiro saber disso antes que alguém da equipe comercial prometa o contrário.

— Justamente — Alice disse. — O contexto não vai servir só para Lucas. Julie também poderia encontrar nele o que sabemos e o que não decidimos. Mas teremos de manter as informações atualizadas e deixar claro quais partes são referências oficiais.

Concordaram em manter uma versão de trabalho editável, sendo o PDF apenas uma fotografia com data. Decisões importantes seriam registradas com seus motivos; princípios recorrentes poderiam, quando apropriado, virar instruções para agentes ou verificações incorporadas às ferramentas. Nada disso substituiria os testes técnicos ou as aprovações necessárias antes de colocar algo em produção.

— Não quero descobrir daqui a seis meses que a IA está aplicando uma regra antiga que ninguém mais defende — disse Hubert.

— Nem eu — Alice respondeu.

Pela primeira vez, Hubert pareceu especialmente interessado no Mini PRD.

— E vocês vão verificar se o cliente realmente consegue rastrear os lotes?

— Sim — Alice respondeu. — Vamos apresentar casos para ele testar conosco. Se o resultado não for confiável, não importa quantas telas terminamos.

Hubert fechou o documento.

— Muito bem. Quero saber o que vocês estão aprendendo, não apenas o que estão entregando.

Na quarta-feira, Alice percebeu algo estranho no aplicativo de mensagens.

Nenhuma mensagem de Lucas desde a manhã.

Conferiu o canal do projeto. Havia atividade: Lucas testara diferentes maneiras de apresentar os resultados na instância de demonstração. Numa delas, explicava que escolhera separar lotes com referências idênticas para não sugerir uma ligação inexistente. Citava o princípio registrado no contexto de Produto, indicava os dados usados e propunha verificar com o cliente se a tela era compreensível.

Ele não tinha pedido autorização.

Alice leu as explicações e voltou ao trabalho. Não tinha nada a acrescentar.

No fim da tarde, Lucas ligou.

— Tenho um laranja que talvez esteja ficando vermelho — disse.

— Pode falar.

— O cliente quer pesquisar também as fabricações antigas. Em parte do histórico, não temos uma ligação confiável entre as matérias-primas e os produtos acabados. Posso mostrar o que sabemos, mas, se apresentarmos isso como uma busca completa, eles

podem entender que não encontrar resultados significa que nenhum produto foi afetado.

Alice endireitou a postura.

— Você tem razão. Não podemos deixar essa ambiguidade.

— Posso acrescentar um aviso e diferenciar os períodos em que os dados são utilizáveis. Mas queria decidir com você o que poderemos prometer ao cliente.

— Aí, sim. E também precisamos consultar o suporte sobre como o cliente entenderia esse limite.

Lucas compartilhou a tela. Começaram a trabalhar no problema. Não era uma aprovação de botão nem uma resposta que Alice já conhecia. Tinham diante deles uma decisão capaz de mudar a confiança de um cliente no software.

Durante a conversa, Alice percebeu que não havia pensado uma única vez no tempo que estava gastando.

Antes de desligar, Lucas hesitou.

— Sabe, seu PDF não respondeu a todas as minhas perguntas.

— Ainda bem — Alice disse.

— Ainda bem?

— Se eu tivesse conseguido colocar todas as respostas ali, significaria que você não teria mais nada para descobrir.

Lucas sorriu.

— Certo. Mas acho que agora entendo melhor quando devo procurar você.

Depois da chamada, Alice abriu a agenda novamente. Tinha uma hora livre antes da próxima reunião com cliente.

Uma hora inteira.

Ainda não sabia se o experimento funcionaria em maior escala, nem quantos erros cometeriam pelo caminho. Só sabia que Lucas avançara sem ela em várias decisões e voltara no momento em que seu julgamento realmente fazia diferença.

Acrescentou ao documento compartilhado uma nova questão a esclarecer sobre a apresentação dos dados históricos. Não era um ticket dizendo a Lucas o que programar. Era uma decisão a investigar com as pessoas envolvidas.

A maneira de trabalhar não se tornara menos exigente. Eles apenas começavam a concentrar a exigência no lugar certo.

8 — Mais autonomia exige limites mais claros, não menos

Uma organização pode decidir permitir que mais pessoas ajam sem esperar por um especialista.

A sobreposição dos papéis e das responsabilidades examinada anteriormente cria um novo problema de coordenação: várias pessoas agora conseguem atuar numa mesma iniciativa sem necessariamente compartilhar os mesmos critérios para decidir.

Mas, se cada nova capacidade de agir precisar ser seguida por uma aprovação humana sistemática, a autonomia continuará sendo teórica.

Teremos apenas deslocado as passagens obrigatórias de uma profissão para outra.

O verdadeiro desafio é construir um sistema no qual mais decisões possam ser tomadas localmente sem perder coerência, segurança, qualidade ou direção de Produto.

É aqui que surge um paradoxo importante:

quanto mais queremos distribuir autonomia, mais explícitos precisam ser os limites.

Autonomia não nasce da ausência de regras.

Ela surge da possibilidade de agir sem pedir autorização constantemente porque os objetivos, os limites, os riscos e as responsabilidades são suficientemente claros.

Se tudo precisa de aprovação, não existe autonomia

Imagine um Product Engineer responsável por uma iniciativa.

Ele conversa com usuários, explora uma solução, cria protótipos, desenvolve e prepara o lançamento.

No papel, é autônomo.

Mas, se precisa perguntar ao PM sobre cada pequena decisão de Produto, ao designer sobre cada detalhe da interface e ao Tech Lead sobre cada escolha técnica, a organização não redistribuiu realmente a responsabilidade. Apenas pediu que uma pessoa passasse por mais aprovações.

O mesmo obstáculo aparece nas outras profissões: permitir que alguém construa algo adianta pouco se cada ação tiver de seguir os processos e aprovações de antes.

A questão, portanto, não é apenas saber **quem pode agir**.

Precisamos saber **dentro de quais limites essa pessoa pode decidir sozinha**.

Na Cursor, essa lógica aparece na definição de objetivos e salvaguardas suficientemente claros para que quem constrói possa agir sem se afastar da visão ou do roadmap. [1]

A organização substitui, assim, parte das instruções detalhadas por um conjunto de critérios para decidir.

Substituir parte dos tickets por critérios de decisão

Vimos, com a Linear, por que um ticket inteiramente prescritivo pode limitar a iniciativa. [2] Agora a questão é prática: de quais informações uma equipe precisa para tomar suas próprias decisões sem esperar por uma nova especificação?

Isso não exige eliminar os tickets. No Malorean, defendi o backlog como ferramenta de ação e coordenação, não como memória integral do Produto. [3]

Um conjunto de critérios para autonomia poderia, portanto, explicitar as informações que orientam decisões locais sem fixar antecipadamente cada solução.

Ele pode conter:

- o cliente ou perfil de cliente ideal (ICP) prioritário;
- o problema a resolver;
- o resultado esperado;
- os limites do escopo;

- as restrições importantes;
- o nível de qualidade esperado;
- o que a organização aceita explicitamente deixar de fazer;
- os limites a partir dos quais uma decisão deve ser escalada.

A Ramp oferece um exemplo concreto dessa lógica: definir claramente os ICPs centrais, os problemas prioritários e aquilo que a empresa quer resolver especialmente bem também permite recusar oportunidades lucrativas quando desviam o produto de sua direção. [4]

A Figma, por sua vez, lembra que passar rapidamente do zero a um protótipo não basta. Casos reais, exceções e padrões de qualidade continuam sendo decisivos para transformar um protótipo rápido num bom produto. [5]

Essa lista ilustra uma escolha de organização: não apenas dizer **o que fazer**, mas tornar visíveis os limites que precisam ser respeitados. Não é apresentada como método universal.

O contexto se torna uma infraestrutura

Para tomar boas decisões localmente, precisamos ter acesso às informações que ajudam a compreender o problema.

Esse talvez seja um dos maiores movimentos organizacionais da era dos agentes de IA.

Uma empresa não pode pedir a mais pessoas que decidam enquanto mantém a maior parte do contexto na cabeça de alguns especialistas. A Linear destaca que feedback de clientes, direção estratégica, decisões e código devem estar acessíveis tanto a pessoas quanto a agentes. [2]

No Malorean, descrevi o papel do PM como **arquiteto do Contexto**: transformar conhecimento disperso — entrevistas, objeções comerciais, suporte, decisões, dados, fontes e sinais fracos — num patrimônio organizacional reutilizável. [6]

Essa ideia muda profundamente a função da documentação.

O objetivo deixa de ser produzir mais páginas.

Passa a ser disponibilizar o que alguém precisa para decidir sem reconstruir todo o raciocínio desde o início.

Um bom contexto permite compreender:

- o que os clientes realmente disseram;
- quais hipóteses já foram testadas;
- quais restrições importam;
- quais decisões estruturantes já foram tomadas;
- quais concessões já foram aceitas;
- quais métricas permitem avaliar o resultado.

Quanto maior a autonomia, mais confiável e compartilhado precisa ser esse contexto.

Caso contrário, a empresa distribui a capacidade de agir sem distribuir a capacidade de compreender.

Preservar a memória do porquê

Essa redistribuição cria outro problema.

Mais decisões são tomadas localmente, às vezes por pessoas diferentes, às vezes com agentes que não participaram das discussões anteriores.

E uma decisão sem memória logo volta a ser discutida.

Lembramos da conclusão, mas já não do contexto.

Sabemos que existe uma regra, mas esquecemos por que foi escolhida.

Chega um novo cliente, entra um novo gestor, uma equipe muda, e uma discussão já encerrada começa de novo com menos informação do que da primeira vez.

É justamente essa a função que atribuo aos **Product Decision Records**. [7]

Um PDR não documenta cada pequena decisão.

Registra decisões de Produto suficientemente importantes para orientar várias equipes, produtos ou situações futuras.

Ele conserva, especialmente:

- o contexto;
- as alternativas consideradas;
- a decisão escolhida;
- os motivos da escolha;
- as consequências positivas e negativas;
- os riscos aceitos;
- as condições que justificariam reabrir a decisão;
- as pessoas que decidiram ou foram consultadas.

A finalidade não é criar mais burocracia.

Pelo contrário, um bom PDR deve evitar que a mesma discussão seja reaberta cinquenta vezes.

Surge então uma distinção útil:

o código registra o que foi construído; a memória das decisões precisa registrar por que certas regras estruturantes foram escolhidas.

Uma autonomia duradoura precisa das duas coisas.

Levar as salvaguardas para dentro do sistema

Mesmo com um bom contexto, seria ineficiente pedir a cada funcionário que memorizasse todas as regras da organização.

Por isso, algumas empresas deslocam regras repetitivas da validação humana para as ferramentas. É um mecanismo observado, não uma garantia de que a automação, sozinha, assegure a qualidade.

A Meta usa o Diff Risk Score para estimar a probabilidade de uma mudança provocar um incidente e orientar fluxos de validação conforme o risco. [8]

A Cloudflare transforma padrões explícitos de Engenharia em verificações automáticas. Algumas regras obrigatórias podem impedir a aprovação de uma mudança até que o problema seja corrigido ou uma exceção explícita seja concedida. [9]

Esse movimento importa porque muda o papel dos especialistas.

Um Guardião de Engenharia não precisa repetir manualmente a mesma regra em cinquenta pull requests se ela puder ser verificada automaticamente.

Um Guardião de Design não precisa aprovar cada botão se o Design System torna a escolha correta simples e os erros visíveis.

Um Guardião de Produto não precisa reexplicar as mesmas restrições todas as semanas se as decisões estruturantes, o público-alvo e os objetivos estiverem acessíveis.

Os limites se tornam infraestrutura.

Testes, CI, Design System, padrões, regras de segurança, templates, instruções para agentes, métricas e contexto compartilhado assumem parte do controle antes exercido pelas pessoas.

A especialização humana pode então se concentrar nas exceções.

Nem todas as decisões merecem o mesmo controle

É aqui que a noção de risco se torna central.

Traduzir um texto e modificar um sistema de permissões não deveriam passar pelo mesmo processo de validação.

Uma mudança local e reversível não precisa do mesmo nível de revisão que uma migração de dados ou uma decisão que altera o posicionamento da empresa.

Uma forma de compreender essas organizações é trocar a pergunta binária — “precisa aprovar?” — por uma pergunta gradual: **qual nível de controle esse risco justifica?**

Os três níveis seguintes são **uma proposta deste livro**, não uma prática comprovada em todas as empresas mencionadas.

Autonomia local

A decisão permanece dentro dos objetivos e padrões conhecidos.

É reversível, suas possíveis consequências são limitadas e não altera nenhuma regra estruturante.

O responsável pela iniciativa decide.

Aprovação direcionada

A decisão introduz uma novidade, uma exceção ou um risco significativo num domínio.

O Guardião correspondente participa: Engenharia, Design, Produto ou outra área.

Ele não assume necessariamente a iniciativa. Oferece o julgamento especializado que está faltando.

Escalonamento

A decisão altera os limites do próprio sistema.

No campo técnico: arquitetura estruturante, segurança, dados sensíveis, migração crítica ou risco regulatório.

Em Design: uma interação nova e importante, ruptura com o Design System ou grandes consequências para outros fluxos de usuário.

Em Produto: mudança de público-alvo, proposta de valor, precificação, investimento relevante, roadmap ou escolhas estratégicas.

Nesse caso, o responsável pela iniciativa já não está apenas decidindo **dentro** dos limites existentes.

Está propondo alterá-los.

No modelo apresentado aqui, esse é o tipo de situação que exige o envolvimento do Guardião ou de um nível superior de decisão.

Sem limites explícitos, aparecem dois riscos opostos

A ausência de limites claros para as decisões expõe a organização a dois perigos contrários.

O primeiro é a burocracia.

Tudo é escalado. Especialistas viram filas de espera. Gestores precisam aprovar decisões banais e a autonomia desaparece.

O segundo é a diluição da responsabilidade.

Cada um decide localmente sem perceber quando uma decisão deixa de ser local. Escolhas contraditórias se acumulam até criar um problema visível para o cliente ou a empresa.

No Malorean, descrevi várias ferramentas de coerência organizacional destinadas justamente a esclarecer essas

fronteiras: linguagem compartilhada, documento de uma página, hierarquia de métricas, registro de decisões e regras de escalonamento. [10]

Seu valor não está em produzir mais documentação. Está em obrigar algumas perguntas a serem feitas explicitamente:

- quem toma a decisão final?
- sobre quais assuntos?
- em que momento?
- o que pode continuar local?
- o que precisa ser escalado?
- em quais condições uma decisão pode ser reaberta?

Esses dois riscos nos levam a uma hipótese organizacional: **posicionar melhor as decisões** pode ser mais importante do que multiplicar ou eliminar aprovações.

Decidir também significa aceitar perdas

As salvaguardas não podem ser apenas técnicas.

Uma parte essencial dos limites está na capacidade de fazer escolhas reais entre alternativas.

As organizações muitas vezes produzem decisões tão vagas que todos conseguem manter a aparência de consenso. Cada pessoa projeta nelas a própria

prioridade, e a execução revela depois que as restrições eram incompatíveis.

Chamo isso de **decisões frouxas**. [11]

Elas são especialmente perigosas num sistema com muita autonomia.

Se o objetivo é ambíguo, cada equipe pode tomar uma decisão localmente razoável, mas incoerente para a organização como um todo.

Dar mais autonomia também exige, portanto, tornar visíveis as perdas aceitas.

Qual cliente vem primeiro?

Qual segmento de mercado não será atendido agora?

Que nível de qualidade é inegociável?

Qual restrição pode ser temporariamente flexibilizada?

Qual objetivo prevalece quando entra em conflito com outro?

Autonomia não exige regras que digam tudo.

Exige limites que tornem as escolhas importantes suficientemente claras para que as decisões locais apontem na mesma direção.

Medir o resultado é a última salvaguarda

A autonomia não termina quando algo é entregue.

Se uma pessoa realmente assume uma iniciativa, também precisa observar o que acontece depois.

O primeiro capítulo já distinguia quantidade produzida e valor criado. Aqui a pergunta é outra: **como alguém que tomou uma decisão descobre, depois, se ela foi acertada?**

A Canva oferece um mecanismo concreto: suas equipes conseguem relacionar experimentos a uma estimativa de impacto comercial. [12]

Para cada iniciativa, isso leva a esclarecer:

- qual transformação era esperada;
- qual indicador permite observá-la;
- qual padrão de qualidade precisava ser preservado;
- o que os usuários realmente fizeram;
- se é melhor continuar, corrigir, abandonar ou investigar mais.

A medição fecha o ciclo da responsabilidade.

Sem ela, a autonomia pode virar apenas uma autorização para produzir.

Com ela, transforma-se em responsabilidade pelo resultado.

Autonomia não significa retirada da gestão

O modelo de equipes autônomas de Marty Cagan, apresentado no capítulo 1.4, ajuda a esclarecer o papel da gestão. Confiar um problema a uma equipe sem entregar uma solução pronta exige objetivos claros e contexto estratégico. [13] Cagan também insiste no desenvolvimento de competências, no acompanhamento e na liderança: deixar a equipe decidir não é abandoná-la. [14]

A comparação tem, porém, um limite importante. Cagan descreve a responsabilidade de equipes de Produto com competências complementares. O conceito de Guardião, proposto neste livro, responde a outro problema: como preservar a responsabilidade por um domínio quando a execução passa de uma profissão a outra?

O Guardião cuida dos limites, não aprova cada gesto

O capítulo 1.6 apresentou o Guardião como responsável pela coerência de um domínio, mesmo sem executar pessoalmente cada tarefa. Aqui, sua função se torna concreta: definir regras compreensíveis, automatizar controles repetitivos e intervir quando o risco ou a novidade ultrapassa o que a equipe pode decidir sozinha.

Se precisar verificar tudo, voltará a ser um gargalo. Se ninguém responder por nada, a responsabilidade se diluirá.

Autonomia, portanto, não é aprovar cada ação nem eliminar todo controle: é permitir mais decisões locais preservando responsabilidades claras.

Mas essa transformação levanta uma questão que vai além de Produto e Engenharia.

Se o contexto circula diretamente, as decisões podem ser tomadas mais perto de onde o trabalho acontece e os agentes absorvem parte da coordenação, **o que acontece com os níveis de gestão que antes exerciam essas funções?**

A Malorean Corp. primeiro nos mostrará como essa pergunta aparece na vida das pessoas. Depois examinaremos suas consequências para a empresa como um todo.

Bibliografia

[1] How Cursor Is Navigating the New Shape of Product Craft — Atlassian / Cursor

- URL: **https://www.atlassian.com/blog/how-we-build/how-cursor-is-navigating-the-new-shape-of--product-craft**

- Mostra como objetivos, salvaguardas e julgamento de Produto permitem agir sem se afastar da direção escolhida.

[2] Issue tracking is dead — Linear

- URL: **https://linear.app/next**
- Defende compartilhar mais contexto entre feedback dos clientes, estratégia, decisões e código, em vez de depender apenas de tickets.

[3] Le backlog n’est pas un dépotoir : c’est un outil d’action — Malorean

- URL: **https://malorean.net/articles/2026-06-03-le-backlog-nest-pas-un-depotoir.pt-br.html**
- Diferencia o backlog como ferramenta de coordenação da memória duradoura do Produto e de suas decisões.

[4] 5 insights from Ramp’s CPO on building for hypergrowth — Ramp

- URL: **https://ramp.com/blog/5-insights-from-ramps-cpo-reddit-ama**
- Destaca ICPs centrais, problemas prioritários e a necessidade de recusar oportunidades que desviem o produto.

[5] What It Takes to Build Great Products Now — Figma

- URL: **https://www.figma.com/blog/what-it-takes-to-build-great-products-now/**
- Lembra que protótipos rápidos precisam considerar casos reais, exceções e qualidade para se tornarem produtos duradouros.

[6] Le PM comme architecte du Contexte — Malorean

- URL: **https://malorean.net/articles/2026-06-08-le-pm-comme-architecte-du-contexte.pt-br.html**
- Propõe transformar sinais, fontes e decisões dispersas em contexto reutilizável para melhores escolhas.

[7] Product Decision Record : tracer les choix produit qui structurent l'entreprise — Malorean

- URL: **https://malorean.net/articles/2026-06-03-product-decision-record-tracer-les-choix-produit.pt-br.html**
- Propõe um registro breve das decisões estruturantes, alternativas, consequências e condições de revisão.

[8] Diff Risk Score: AI-driven risk-aware software development — Meta

- URL: **https://engineering.fb.com/2025/08/06/developer-tools/diff-risk-score-drs-ai-risk-aware--software-development-meta/**
- Apresenta uma pontuação de risco para adaptar os fluxos de revisão conforme a probabilidade de incidentes.

[9] How Cloudflare enforces engineering standards using AI — Cloudflare

- URL: **https://blog.cloudflare.com/engineering--standards-enforcement/**
- Descreve padrões de Engenharia convertidos em verificações automáticas que podem impedir mudanças inadequadas.

[10] Les outils de cohérence organisationnelle — Malorean

- URL: **https://malorean.net/articles/2026-06-02-les--outils-de-coherence-organisationnelle.pt-br.html**
- Relaciona linguagem comum, objetivos, métricas, memória das decisões e regras de escalonamento.

[11] Pourquoi les organisations préfèrent les décisions molles — Malorean

- URL: **https://malorean.net/articles/2026-06-02--pourquoi-les-organisations-preferent-les--decisions-molles.pt-br.html**
- Analisa decisões vagas que escondem perdas, diluem responsabilidades e adiam conflitos até a execução.

[12] Measuring Commercial Impact at Scale — Canva

- URL: **https://www.canva.dev/blog/engineering/measuring-commerical-impact-at-scale/**
- Descreve a medição do impacto comercial que conecta experimentos locais e criação de valor.

[13] Team Objectives – Empowerment — Marty Cagan, Silicon Valley Product Group (2020)

- URL: **https://www.svpg.com/team-objectives--empowerment/**
- Defende confiar problemas às equipes, junto com o contexto estratégico necessário para decidir.

[14] Product Leadership Is Hard — Marty Cagan, Silicon Valley Product Group (2020)

- URL: **https://www.svpg.com/product-leadership-is--hard/**

- Explica por que a autonomia de equipes de Produto exige liderança, acompanhamento e desenvolvimento de competências.

9 — Malorean Corp. — Duas decisões razoáveis

Na segunda-feira de manhã, Vivien, responsável pela coordenação das duas equipes de desenvolvimento, abriu a reunião com a mesma pergunta que fazia havia anos.

— Alguém pode me dizer como estão as tarefas desta semana?

Ele já tinha seu painel aberto. Tickets, responsáveis, prazos e impedimentos estavam organizados nas colunas que aperfeiçoara pacientemente. Conhecia as duas equipes de cinco desenvolvedores, seus pontos fortes, seus hábitos, os assuntos em que era preciso insistir e aqueles que era melhor deixar amadurecer.

Ninguém respondeu imediatamente.

Então Lucas falou.

— Na rastreabilidade, atualizei o Mini PRD na sexta. Testamos a busca por lote com o cliente. O próximo passo é validar os casos em que os dados estão incompletos. Registrei as perguntas abertas no contexto.

— E sua data de entrega? — Vivien perguntou.

— Ainda depende das verificações do estoque histórico.

Thomas precisa examinar uma questão comigo.

Thomas assentiu. Por sua vez, trabalhava numa melhoria do fluxo de preparação dos envios. Também havia atualizado o andamento na ferramenta.

Vivien passou aos outros desenvolvedores. As respostas eram parecidas. Cada um entendia melhor seu problema, havia tomado certas decisões e conseguia explicar o próximo passo. Alguns ainda pediam ajuda para dividir o trabalho. Os demais não tinham esperado pela reunião para avançar.

No fim, Vivien olhou para o painel.

Tudo o que acabara de ouvir já estava ali, às vezes com mais detalhes.

— Bom — disse. — Passamos por tudo.

A reunião durara doze minutos.

Antes, teria levado quarenta e cinco.

Vivien não sabia ao certo se deveria ficar feliz.

Trabalhava na Malorean havia tempo suficiente para ter conhecido todos os desenvolvedores reunidos no escritório de Le Havre. Organizara entregas, administrara atrasos, negociara prioridades com Produto e explicara

ao diretor por que uma demanda prometida para sexta não ficaria pronta antes da terça seguinte.

Quando a empresa adotou o trabalho remoto, depois da Covid, precisou adaptar seu método. As conversas de corredor já não bastavam. Pediu que as decisões fossem registradas, que os impedimentos ficassem visíveis e que todos anotassem o que haviam combinado com outros departamentos.

Aquilo se tornara seu trabalho cotidiano: fazer circular informações que ninguém enxergava por completo.

Ele gostava. Gostava especialmente dos momentos em que um projeto que tinha saído dos trilhos retomava o rumo porque alguém finalmente aproximara as pessoas certas.

Mas, desde a chegada das ferramentas de IA ao trabalho das equipes, algo havia mudado. Os desenvolvedores produziam mais rápido, mas, sobretudo, alguns haviam começado a assumir os assuntos de outra maneira. Lucas conversava diretamente com clientes. Alice compartilhava um contexto de Produto que explicava as razões das decisões, não apenas a lista de funcionalidades a desenvolver. Bob começava a adotar a mesma prática nos próprios projetos.

E agora um agente preparava para Vivien parte dos resumos que antes ele montava manualmente.
O trabalho não tinha desaparecido. Escapava aos poucos de suas mãos, ou estava sendo feito em outro lugar.
Ainda não havia conversado com ninguém sobre isso.

Na quarta-feira seguinte, Lucas mostrou a Alice uma evolução da demonstração de rastreabilidade.
— Testamos o que acontece quando um fornecedor sinaliza um lote potencialmente defeituoso. Conseguimos encontrar as fabricações conhecidas. E acrescentei um estado “em análise” para não dar a impressão de que o lote está liberado.
Alice examinou a tela.
Um aviso informava que a situação precisava ser verificada. Para as fabricações antigas, uma segunda mensagem explicava que faltavam algumas informações.
— É mais honesto do que a primeira versão — ela disse.
— Sim. O responsável pela qualidade do cliente pediu justamente que não confundíssemos “nenhum problema identificado” com “não temos as informações necessárias”.

Alice concordou. Já haviam tratado dessa distinção no documento de contexto.

— Você registrou no Mini PRD?

— Sim. E acrescentei casos de teste.

— Então continue.

Ela não precisou decidir onde colocar o aviso nem qual seria o texto exato. Lucas sabia o resultado que queriam alcançar. Escolhera uma apresentação e planejara confrontá-la com o cliente.

Pouco depois, Thomas apresentou a Vivien uma melhoria do módulo de expedição.

O problema era diferente. Em alguns clientes, um pedido podia ficar parado por causa de estados intermediários ou mal preenchidos, mesmo quando não havia nenhuma proibição explícita. As equipes de logística perdiam tempo corrigindo registros para os quais não havia risco identificado.

Thomas esclarecera as regras de preparação: um lote marcado como “bloqueado” impedia a saída; nas demais situações, os dados continuavam visíveis e seguiam o fluxo habitual, com os controles existentes.

— Tem certeza de que isso não altera o comportamento de outros módulos? — Vivien perguntou.

— Nos casos existentes, sim. Acrescentei testes de regressão. Não mexi nas regras de bloqueio.

Vivien não tinha motivo para desconfiar do cuidado de Thomas. Ele conhecia exceções que nem a documentação histórica mencionava mais.

— Muito bem. Peça para revisarem a alteração antes de irmos adiante.

Thomas concordou.

As duas mudanças ainda não tinham sido reunidas no mesmo cenário.

O problema apareceu na sexta-feira, num ambiente de testes integrados.

Uma pessoa do suporte havia preparado um fluxo completo: recebimento de uma matéria-prima, fabricação, alerta do fornecedor, identificação dos produtos afetados e preparação de uma expedição.

Lucas acompanhava a demonstração por videochamada. Thomas também estava presente. Alice pedira para participar, a fim de observar como alguém que não havia criado o fluxo entenderia o funcionamento.

Tudo correu bem nos primeiros minutos.

O número do lote foi encontrado. As fabricações apareceram. O aviso “em análise” foi exibido.

Então a pessoa do suporte passou para o módulo de expedição.

O documento de entrega permanecia em preparação.

Nenhuma mensagem impedia o avanço.

— Espera — Alice disse. — Estamos falando do mesmo lote?

Lucas se inclinou na direção da tela.

— Sim. Por quê?

— Lá em cima ele está em análise, mas aqui podemos continuar como se não houvesse nada de especial.

Thomas abriu os detalhes do pedido.

— Não está bloqueado. Está em análise. Não é o mesmo estado.

— Para mim — Lucas respondeu —, “em análise” significava justamente que não deveríamos tratá-lo como liberado enquanto alguém não tivesse decidido.

Thomas virou o rosto para a câmera.

— Então por que você não marcou como “bloqueado”?

— Porque não queria inventar uma proibição geral. Temos clientes que precisam investigar sem interromper

todas as operações. Cabe ao responsável pela qualidade decidir o que realmente foi afetado.

— E no meu módulo — Thomas retrucou — temos lotes antigos com informações incompletas. Se eu bloquear tudo o que não estiver explicitamente liberado, alguns pedidos nunca mais vão andar.

Alice olhou para as duas telas.

Entendia os motivos de Lucas. Entendia os de Thomas. O problema era que um usuário podia ler “em análise” numa parte do ERP e não encontrar nenhum aviso que o impedisse de continuar na outra.

— Vamos parar a demonstração aqui — ela disse.

Ninguém protestou. Nenhum dado real de cliente havia sido alterado, e as mudanças não tinham ido para produção. Era exatamente para isso que servia o teste.

Mas o silêncio que veio depois era menos confortável do que após uma demonstração bem-sucedida.

Vivien entrou na chamada quando explicaram o que acabara de acontecer.

Deixou Lucas descrever sua decisão, depois Thomas explicar a dele. Fez algumas perguntas sobre os testes e o escopo de cada mudança.

— Cada um tem seu Mini PRD? — perguntou.

— Sim — Lucas respondeu.

— E suas regras de decisão?

— Sim — disse Thomas. — Não alterei o bloqueio existente. Era uma melhoria local do fluxo de preparação.

— Eu também não — Lucas disse. — Acrescentei informações para não esconder uma dúvida. Tínhamos escrito justamente que isso era importante.

Vivien abriu os dois documentos lado a lado.

Nenhum estava errado. Simplesmente tratavam de coisas diferentes.

O primeiro descrevia como sinalizar uma incerteza na rastreabilidade. O segundo descrevia quando uma expedição deveria parar. Nenhum havia sido escrito para responder à pergunta que os conectava: o que o sistema deveria fazer quando um lote era suspeito, mas ainda não estava explicitamente bloqueado?

— Temos uma lacuna entre os dois projetos — Vivien disse.

Thomas assentiu.

— Tecnicamente, podemos escolher diferentes comportamentos. Mas precisamos saber qual queremos.

— E precisamos perguntar ao cliente o que o responsável pela qualidade realmente faz nesse caso — Lucas acrescentou. — Tenho hipóteses, não uma regra válida para todo mundo.

Alice falou.

— Vou esclarecer qual deve ser o comportamento do Produto. Mas não sozinha. Precisamos do suporte, de Lucas, de Thomas e provavelmente de um cliente para verificar as consequências.

Vivien levantou o olhar.

— Então organizo uma reunião para amanhã.

Alice hesitou.

— Você pode nos ajudar a reunir as pessoas certas, sim. Mas não é seu papel carregar a decisão por nós.

Não disse aquilo para excluí-lo. Mesmo assim, Vivien sentiu que algo havia mudado.

Passara anos sendo aquele que encaminhava os problemas para o processo certo. Desta vez, as pessoas envolvidas já sabiam que precisavam conversar. Não precisavam que ele transformasse a divergência numa sequência de tickets.

Precisavam de outra coisa: uma decisão que nenhum dos dois desenvolvedores podia tomar sozinho.

No dia seguinte, Alice começou perguntando a cada um o que procurava evitar.

Lucas não queria que o responsável pela qualidade acreditasse ter uma rastreabilidade completa quando faltavam dados.

Thomas não queria que uma regra imprecisa paralisasse sem necessidade as operações de clientes cujos históricos estavam incompletos.

O suporte lembrou que um alerta de fornecedor nem sempre significava recolhimento imediato. Alguns clientes precisavam primeiro determinar a extensão do problema, consultar as equipes de qualidade e decidir o que fazer.

— Certo — Alice disse. — Talvez nosso erro tenha sido tentar dar dois significados diferentes ao mesmo estado.

Compartilhou uma proposta.

Um lote poderia ser identificado como **a verificar**, sem que o ERP fingisse tomar, no lugar da empresa cliente, uma decisão regulatória ou de qualidade. Por outro lado, os fluxos que pudessem usar ou expedir esse lote teriam de tornar o alerta visível e exigir o procedimento previsto de confirmação, em vez de deixá-lo desaparecer.

As informações históricas incertas deveriam continuar sinalizadas como tal. E os casos em que o software não tinha dados suficientes não poderiam ser apresentados como ausência de risco.

— Isso podemos verificar no módulo de expedição — Thomas disse. — Mas também quero testar os pedidos antigos.

— E quero conferir os termos com o cliente — Lucas acrescentou. — “A verificar” talvez não tenha o mesmo sentido para o responsável pela qualidade.

Alice concordou.

Ainda não haviam terminado de conceber a solução. Mas finalmente haviam identificado quem precisava decidir o quê.

Ela registrou a decisão provisória e as perguntas em aberto no contexto compartilhado, com os motivos e os casos a verificar. Acrescentou uma condição à tabela: quando uma decisão local alterava o significado de uma informação usada em vários módulos, deixava de ser automaticamente verde, mesmo que o código parecesse reversível.

— Deveríamos ter percebido isso antes — Lucas disse.

— Sim — Alice respondeu. — Mas é justamente por isso que fazemos testes. Agora sabemos o que observar na próxima vez.

Thomas examinou os dois documentos.

— Também precisamos de um teste que atravesse os módulos. Temos muitos testes que verificam cada comportamento separadamente. Bem menos que testam o que acontece quando eles se encontram.

Vivien, que ouvia havia alguns minutos, anotou.

Pela primeira vez desde o início da chamada, reencontrava um problema que conhecia bem: ninguém tinha uma visão completa de todas as dependências.

Mas o surpreendia a maneira como a solução tomava forma. Alice não havia ditado as telas. Lucas e Thomas não esperavam uma divisão de tarefas. O suporte trazia a experiência do cliente. Seu próprio papel fora tornar visível a lacuna e facilitar a conversa.

Isso tinha sido útil.

E tinha muito pouco a ver com a reunião de entregas que organizava toda segunda-feira.

À tarde, Thomas recebeu uma mensagem de um desenvolvedor menos experiente da outra equipe.

O colega não entendia por que um processamento antigo de estoque fazia duas verificações aparentemente idênticas. O agente de programação sugeria eliminar uma delas.

Thomas abriu o arquivo.

— Não mexa nisso antes de olharmos juntos — respondeu.

Passaram meia hora compartilhando a tela. A segunda verificação cobria um caso herdado de uma alteração antiga do produto, pouco explicada no código. Thomas não lembrava exatamente de qual entrega ela vinha, mas sabia onde procurar.

— Sua ferramenta não inventou uma resposta ruim — explicou. — Encontrou duas verificações parecidas e sugeriu simplificar. Sem conhecer a história do software, entendo por quê.

— Você realmente nunca usa IA para programar? — o desenvolvedor perguntou.

— Não para propor alterações às cegas em assuntos como esse.

— Mas você acabou de passar trinta minutos numa coisa que Lucas talvez verificasse em cinco.

Thomas pensou.

— Talvez. E talvez Lucas encontrasse a mesma coisa. O importante é você entender por que a verificação existe. Na próxima vez, saberá fazer uma pergunta melhor tanto à ferramenta quanto a mim.

O desenvolvedor perguntou se podia documentar a explicação.

— Por favor. Estou cansado de ser o único que se lembra.

Thomas desligou, surpreendido com a própria frase. Durante anos, seu conhecimento das exceções o tornara indispensável. Às vezes temia que a IA tornasse essa experiência menos valiosa. A conversa mostrou outra possibilidade: podia ajudar os colegas a desenvolver o discernimento que ainda lhes faltava, em vez de guardar cada resposta para si.

Abriu o documento compartilhado e escreveu algumas linhas.

Na reunião de segunda-feira, Hubert perguntou a Alice como avançava o experimento.

Ela contou sobre as duas decisões incompatíveis, o teste que as revelara e a correção dos limites.

— Então vocês deixaram dois desenvolvedores decidirem e quase criaram uma incoerência? — Hubert resumiu.

— Sim.

— Isso não me tranquiliza muito.

— Detectamos antes de qualquer entrada em produção — Alice respondeu. — O problema não é que duas pessoas tenham decidido. É que nenhuma de nossas regras dizia como as escolhas delas deveriam se encontrar.

Hubert olhou as anotações do incidente.

— Antes, provavelmente teríamos organizado mais uma reunião de aprovação.

— Talvez. Mais outra reunião não necessariamente revelaria a questão. Precisávamos de alguém testando o fluxo completo e de responsabilidades claras quando uma escolha afeta vários domínios.

— E agora quem decide?

— Lucas conduz seu projeto. Thomas conduz o dele e nos ajuda com os riscos técnicos. Eu sou a Guardiã de Produto e respondo pela coerência do conjunto. Nos casos delicados, consultamos os especialistas envolvidos. Não podemos transferir tudo isso para uma ferramenta.

Hubert assentiu lentamente.

— E Vivien?

Alice demorou um pouco a responder.

— Ajudou a aproximar as equipes e entender onde estava o problema. Mas não precisava distribuir as tarefas nem decidir o comportamento do Produto.

Hubert pareceu pensar em outra coisa.

— Eu o vi na sexta — disse. — Mostrou-me o novo acompanhamento dos projetos. O relatório é muito bom. Mas acho que uma máquina poderia preparar oitenta por cento do que ele contou.

— Ele também sabe — Alice respondeu.

— O que não sei é o que devemos concluir disso.

Alice não tinha uma resposta pronta.

— Não que ele tenha deixado de ser útil — disse, por fim.

— Acabamos de ver que ele sabe fazer as pessoas trabalharem juntas. Só que talvez esse trabalho não precise mais ser um cargo pelo qual tudo passa.

Hubert continuou olhando para a tela.

— Vamos precisar conversar com ele.

Vivien recebeu o convite para o dia seguinte.

Seu primeiro pensamento foi uma reorganização. Nas últimas semanas, sua agenda passara a ter mais espaços vazios. Os desenvolvedores conduziam seus próprios projetos. As ferramentas coletavam automaticamente

parte das informações que antes ele passava as noites reunindo. Continuava lidando com dificuldades humanas, integrando os recém-chegados e destravando situações complexas, mas essas intervenções já não enchiam sua agenda como antes.

Gostava de ser indispensável às entregas da Malorean. Não tinha certeza de gostar de descobrir que era menos indispensável.

No entanto, a reunião começou com uma pergunta que o surpreendeu.

— O que você gostaria de continuar fazendo, Vivien? — Hubert perguntou.

— Meu trabalho — ele respondeu.

Hubert sorriu.

— É exatamente isso que estamos tentando entender.

Vivien olhou para Alice, convidada para falar do funcionamento das equipes, e voltou a olhar para Hubert.

— Não quero que a gente finja. As equipes já não precisam que eu pergunte toda semana como estão os projetos. E eu não quero passar o tempo lendo resumos que um agente já preparou.

— Do que elas ainda precisam? — Alice perguntou.

— Dos conflitos que ninguém registra. Das pessoas que hesitam em dizer que estão perdidas. Dos novos funcionários que não sabem a quem perguntar. Das decisões que atravessam várias equipes. E, às vezes, de alguém que simplesmente tenha tempo para ouvir.

Ele fez uma pausa.

— Mas isso não significa que tudo precise continuar passando por mim.

Hubert endireitou-se na cadeira.

— Não. E vamos precisar dizer claramente quem responderá por essas coisas se seu papel mudar. Não quero eliminar um cargo e descobrir depois que ninguém mais cuida das pessoas.

Vivien concordou.

Conversaram sobre o acompanhamento dos desenvolvedores, as decisões entre equipes e o que precisava continuar sob responsabilidade da direção.

Alice falou do papel de Thomas em ajudar os colegas a compreender o código antigo.

Nenhuma solução definitiva saiu da chamada.

Quando terminou, Vivien ficou alguns instantes olhando para a agenda.

Sentia-se menos ameaçado do que na véspera, mas ainda não estava tranquilo. Sua profissão não havia desaparecido. Estava se decompondo em várias responsabilidades, algumas das quais poderiam ser assumidas de outra forma.

Naquela noite, fechou o computador mais cedo.

Pela primeira vez em muito tempo, não tinha nenhuma entrega atrasada para recuperar.

E ainda não sabia o que faria com o tempo que havia recuperado.

10 — A IA fará desaparecer os níveis intermediários das empresas?

Hoje uma equipe pode consultar diretamente suas decisões, agir dentro de regras conhecidas e confiar parte do acompanhamento a agentes. Precisa menos de um gestor para transmitir informações ou organizar cada etapa do trabalho.

Diante disso, **o que acontece com os níveis hierárquicos que cumpriam essa função?**

A resposta pode parecer óbvia: se a IA assume a coordenação, os gestores intermediários perdem a utilidade. Só que um gestor não se limita a repassar instruções. Ele toma decisões difíceis, acompanha pessoas, conecta equipes e cuida de problemas que não cabem em nenhum procedimento.

A pergunta verdadeira, portanto, não é tanto quantos cargos vão desaparecer, mas **para onde irão suas funções e seu poder de decisão**.

Uma hierarquia construída para coordenar decisões

Um cliente importante relata um problema. O suporte o comunica ao gestor, que procura a equipe de Produto.

Produto pede uma estimativa a Engenharia. Vários responsáveis reúnem informações, comparam urgências e talvez a decisão suba ainda mais um nível.

Esse caminho não é necessariamente absurdo. Cada gestor possui parte do contexto e pode comprometer parte da empresa com uma decisão. O problema aparece quando a cadeia se torna a única maneira de compreender o problema e decidir.

Historicamente, a hierarquia permitiu às empresas crescerem apesar da fragmentação das informações. Os níveis intermediários traduzem objetivos da direção em prioridades, coordenam equipes, acompanham dependências e resolvem disputas por recursos. Também participam da avaliação dos funcionários, de seu desenvolvimento e da solução de dificuldades humanas.

Mas nem todas essas atividades são da mesma natureza. Preparar um relatório de incidente é diferente de decidir abandonar uma oportunidade comercial. Acompanhar um prazo não exige o mesmo discernimento que resolver um conflito ou ajudar alguém a se desenvolver.

No Malorean, já defendi uma distinção semelhante a respeito do Product Owner: um cargo pode perder sua justificativa como simples intermediário sem que a função que deveria cumprir deixe de ser necessária.

Precisamos questionar o trabalho realizado, não apenas o título de quem o executa. [1]

Se os níveis intermediários diminuírem, quem assumirá as funções que exerciam?

O trabalho remoto já havia mudado as regras da coordenação

A IA não inventou essa pergunta. Empresas que trabalham remotamente, às vezes em vários países e fusos horários, tiveram de aprender a decidir sem depender de conversas de corredor ou reuniões improvisadas.

Na GitLab, o princípio **handbook-first** consiste em deixar regras e contexto acessíveis por escrito. Qualquer pessoa pode propor uma alteração; os responsáveis pelo domínio decidem se ela será incorporada. Assim, um funcionário pode encontrar uma informação sem esperar que um gestor a transmita. [2]

A Alan descreve uma prática diferente, mas complementar. Decisões importantes são discutidas em conversas no GitHub: uma pessoa toma a decisão e a implementa, outra permanece responsável pelo sucesso da iniciativa, enquanto os demais envolvidos são

consultados ou informados. A discussão pode ser aberta sem exigir unanimidade. [3]

Na 37signals, criadora do Basecamp, grande parte da coordenação acontece por trocas escritas, balanços e informações mantidas junto aos projetos, em vez de dispersas em conversas passageiras. [4]

Não são três empresas sem gestão. São três maneiras de **reduzir a dependência de alguém que concentra as informações ou precisa lembrar repetidamente o que foi decidido**.

Esse precedente importa para os agentes de IA. Um sistema consegue consultar uma regra, recuperar uma decisão e identificar o responsável desde que a organização tenha explicitado essas informações. Uma decisão guardada apenas na cabeça de um gestor continua inacessível tanto aos colegas quanto aos agentes.

Mas é preciso cuidado para não idealizar o trabalho remoto. Um estudo conduzido na Microsoft com mais de 61 mil funcionários observou, entre outras coisas, redes de colaboração mais isoladas depois da adoção do trabalho remoto. Escrever mais não garante nem o compartilhamento do contexto nem a cooperação entre equipes. [5]

O verdadeiro legado do trabalho remoto não é o desaparecimento do escritório: é a possibilidade de organizar parte do trabalho em torno de informações e decisões acessíveis sem um intermediário permanente. A IA pode se apoiar nessa conquista. Não a criou sozinha.

A IA começa a assumir parte desse trabalho

Um documento guarda uma regra; um painel mostra um atraso. Um agente pode ir além: cruzar informações, preparar um relatório de andamento, detectar uma contradição, sugerir uma resposta ou executar uma ação autorizada.

São justamente atividades que consumiam parte do tempo dos gestores: consolidação, relatórios, acompanhamento e preparação de decisões. Seu custo pode diminuir. Seu desaparecimento não está demonstrado.

Uma análise exploratória da HEC Paris, baseada entre outras fontes em 45 entrevistas, sugere que tarefas gerenciais estruturadas e ricas em dados se prestam melhor à automação do que aquelas que dependem da confiança, das relações humanas e do julgamento. Ela

não mede nem as horas de gestão realmente eliminadas nem os cargos que se tornaram desnecessários. [6]

Um experimento realizado na Procter & Gamble também mostra que, em tarefas de inovação de Produto, uma pessoa trabalhando com IA pode obter resultados comparáveis aos de uma dupla sem IA. A ferramenta ajuda, em particular, a combinar perspectivas de áreas diferentes. Mas facilitar uma colaboração pontual não significa substituir um gestor de forma permanente. [7]

Em outro experimento, com 7.137 funcionários de 66 empresas, usuários frequentes de um assistente de IA economizaram cerca de duas horas semanais com e-mails durante a segunda metade do estudo. Mesmo assim, os pesquisadores não observaram uma transformação significativa da composição das tarefas ao longo de seis meses. O estudo tratava de um assistente de escritório, não de agentes que executassem processos inteiros por conta própria. [8]

Essa diferença é decisiva: **um ganho individual de produtividade ainda não é uma transformação organizacional**.

Para que um nível hierárquico seja menos necessário, as informações precisam realmente chegar às equipes, os direitos de decisão devem mudar e as situações

incomuns precisam ter um lugar para serem resolvidas. Não é apenas uma questão de ferramentas, mas de organização.

Três empresas, três trajetórias muito diferentes

As organizações não estão reagindo da mesma maneira. Três casos permitem distinguir uma reorganização já iniciada, uma ambição concebida em torno da IA e as dificuldades de uma transformação acelerada.

Bayer: uma hierarquia efetivamente reduzida

Com o modelo **Dynamic Shared Ownership** (DSO), a Bayer redistribuiu parte das decisões para equipes multidisciplinares. Na carta aos acionistas de 2025, o dirigente do grupo declarou ter eliminado até seis níveis organizacionais e reduzido em aproximadamente dois terços os cargos de gestão. Relatou decisões mais rápidas e prazos de desenvolvimento mais curtos. [9]

A dimensão dessa mudança merece atenção, mesmo que os resultados sejam relatados pela própria Bayer. A transformação pressupõe alterar responsabilidades cotidianas, não apenas cargos num organograma: equipes decidem mais localmente, gestores dedicam mais tempo ao acompanhamento das pessoas e algumas funções de coordenação são reorganizadas.

Uma observação, porém, muda a interpretação do caso: **o DSO começou antes da expansão dos agentes de IA generativa**. A Bayer demonstra que um grande grupo pode reduzir fortemente sua hierarquia; não prova que a IA tenha causado essa redução.

Block: fazer o contexto circular sem passar por toda a hierarquia

A Block vai mais longe no papel que pretende dar à IA. Num texto assinado conjuntamente por Jack Dorsey e pelo investidor Roelof Botha, seus dirigentes descrevem uma organização apoiada por um sistema compartilhado de informações.

A ambição é que um funcionário compreenda uma situação e consiga agir sem esperar que um gestor lhe transmita o contexto. Os autores distinguem os profissionais que contribuem diretamente, os responsáveis por resultados transversais (DRIs) e os gestores que continuam produzindo enquanto orientam outras pessoas, chamados de **player-coaches**. [10]

O gestor cuja principal função era repassar informações perderia importância. Em contrapartida, o desenvolvimento das pessoas, a responsabilidade pelos resultados e as decisões da direção continuariam existindo.

É uma ambição descrita pela empresa e por seu investidor, não uma avaliação independente que comprove o sucesso duradouro do modelo.

Meta: quando o volume produzido supera a capacidade de controle

O contraponto vem de uma investigação da Reuters sobre os projetos de reorganização da Meta em 2026: equipes menores, grande uso de IA e redução planejada ou iniciada de alguns níveis. Uma segunda onda de reestruturação acabou sendo abandonada.

Segundo documentos internos citados pela Reuters, as mudanças em sistemas e infraestruturas internos cresceram muito mais rapidamente do que as novas funcionalidades entregues aos usuários. Ao mesmo tempo, aumentaram os incidentes técnicos e o tempo necessário para resolvê-los. A Meta contestou ou apresentou ressalvas a certos aspectos da reportagem. [11]

Esses números não permitem atribuir os incidentes à IA nem à redução dos gestores: várias transformações aconteciam simultaneamente. Mas levantam uma questão fundamental: de que adianta produzir mais se a empresa já não consegue verificar, estabilizar e absorver o que produz?

A Bayer mostra a possibilidade de uma estrutura mais plana, a Block desenha uma versão pensada para a IA e a Meta lembra os riscos de uma reorganização na qual a produção cresce mais rápido que a coordenação. Nenhuma dessas trajetórias anuncia, sozinha, o futuro da gestão.

As funções de gestão não desaparecem com os cargos

Pense num exemplo: um cliente relata uma falha no software. Enquanto o problema é conhecido e as regras de solução são claras, a equipe pode agir sem esperar por uma sequência de autorizações. Com dados confiáveis, consegue investigar, aplicar uma correção e informar os demais.

Então surgem dois diagnósticos contraditórios. O agente recomenda uma ação, um desenvolvedor recomenda outra. A correção pode afetar um serviço vizinho.

Nesse momento, alguém precisa tornar o desacordo visível, reunir as competências necessárias e decidir qual risco aceitar. A autonomia não eliminou a coordenação; mudou a forma de exercê-la.

Um experimento de Martin Krzywdzinski e colegas oferece um indício interessante. Numa linha de produção simulada, 87 duplas enfrentaram falhas com ajuda de um

sistema de IA que às vezes errava. As equipes auto-organizadas se comunicaram mais e resolveram melhor os problemas do que aquelas com um líder designado. A análise destaca o papel decisivo da comunicação. [12]

Duas pessoas num laboratório não representam uma grande empresa. Mas o estudo lembra uma distinção importante: **uma equipe pode trabalhar sem chefe formal; não consegue dispensar o trabalho de coordenação.**

Esse trabalho pode ser feito pelos colegas, por um especialista, por alguém com mandato para decidir ou por um gestor hierárquico. As escolhas variam conforme competências, riscos e divergências.

Também permanece o que nenhum relatório automático basta para realizar: receber uma pessoa nova, ajudá-la a evoluir, resolver tensões, acompanhar uma decisão difícil ou reconhecer um bom trabalho. Nem todos os gestores fazem bem essas tarefas, e elas podem ser distribuídas de outra forma. Mas alguém ainda precisa ter essa missão e tempo para cumpri-la.

A Gallup observa que o uso de IA não elimina a preocupação dos trabalhadores com o emprego e que a percepção de apoio no trabalho está associada a menor preocupação. Isso não demonstra a necessidade de um

gestor tradicional, mas lembra que a organização também é uma comunidade humana. [13]

Por fim, quem formará aqueles que amanhã precisarão tomar as decisões difíceis?

Já levantei no Malorean uma preocupação com os desenvolvedores juniores: quando uma IA escreve código e corrige erros imediatamente, ela pode eliminar oportunidades de aprender a depurar, entender uma falha e desenvolver discernimento. É uma preocupação a ser estudada, não prova de perda generalizada de competências. [14]

O mesmo problema se aplica aos futuros gestores. Também aprendemos observando decisões difíceis, recebendo feedback e vendo como uma equipe sai de um conflito. Uma empresa que elimina níveis hierárquicos precisa prever outras formas de transmitir essa experiência.

No fim, cada função pode ter vários destinos: automação, incorporação pela equipe, transferência a um especialista ou escalonamento à direção. **Uma tarefa pode desaparecer, mudar de mãos ou se acrescentar ao trabalho de outra pessoa.**

Por isso, para avaliar a eficiência de uma organização mais plana, não basta contar os cargos eliminados. É preciso comparar o acompanhamento evitado com as novas cargas de supervisão dos agentes, resolução de incidentes, apoio e formação. Ainda não dispomos de um balanço líquido robusto dessa redistribuição.

Uma organização mais plana é realmente mais autônoma?

Resta uma última pergunta, menos visível nos organogramas: **quem decide quando os intermediários desaparecem?**

Podemos imaginar equipes mais bem informadas, livres para resolver mais problemas sem escalar cada decisão. Mas também podemos imaginar o contrário: menos chefes no cotidiano e mais decisões controladas diretamente pela direção por meio dos sistemas.

Essa tensão existia antes da IA generativa. Numa pesquisa sobre empresas industriais, Nicholas Bloom e colegas mostram que ferramentas que dão mais informações aos funcionários estão associadas a maior autonomia local. As que facilitam a comunicação com superiores podem, ao contrário, favorecer a centralização: torna-se mais fácil pedir instruções ao topo. [15]

Esse precedente nada demonstra sobre os agentes atuais. Mas ajuda a esclarecer um mecanismo: **distribuir informações não é a mesma coisa que distribuir poder**.

Os sistemas de gestão algorítmica tornam o risco mais concreto. Podem distribuir tarefas, sugerir prioridades, medir desempenho e emitir alertas. Pesquisas mostram que certas funções de direção ou avaliação podem ser exercidas por essas ferramentas, mesmo quando gestores humanos continuam presentes para aplicar as decisões ou tratar exceções. [16]

Um sistema não aparece como superior no organograma. No entanto, se decide quais tarefas cada pessoa recebe, com quais critérios o resultado será avaliado e em que condições uma ação é permitida, organiza parte da autoridade.

Precisamos, então, examinar as decisões por trás das telas: quem define os objetivos dos agentes? Quem escolhe as regras e pode mudá-las? Quem pode questioná-las? E, quando um algoritmo erra, quem tem o poder real de desfazer o que ele desencadeou?

Uma equipe pode ter enorme capacidade de execução e não conseguir mudar suas prioridades. Outra pode

manter um gestor e, ainda assim, decidir de verdade seus métodos e suas escolhas locais.

No Malorean, já descrevi essas “decisões frouxas” que criam aparência de acordo sem explicitar a escolha, a perda aceita ou quem tem a palavra final. Eliminar um nível hierárquico não resolve o problema se o poder continuar tão difícil de identificar. [17]

Podemos, portanto, imaginar um paradoxo: **menos hierarquia visível, mas mais controle centralizado nas ferramentas**.

Isso não é inevitável. Os mesmos sistemas podem ajudar as equipes a decidir localmente e tornar as regras de poder mais transparentes. Tudo depende do que a empresa realmente escolhe delegar, não do número de caixas eliminadas do organograma.

O que poderia realmente desaparecer

A IA fará desaparecer os níveis intermediários das empresas?

Alguns níveis podem se tornar menos necessários. Mas seu desaparecimento não será automático nem significará o fim da gestão.

Funções que consistem principalmente em transmitir informações, consolidar relatórios de andamento ou

organizar aprovações repetitivas podem perder relevância. Outras, ligadas a decisões difíceis, conflitos, transmissão de experiência e responsabilidade, continuarão precisando ser exercidas de alguma forma.

A Bayer mostra que uma redução expressiva dos níveis é possível sem que a IA seja a causa identificada. A Block propõe confiar mais coordenação a um sistema compartilhado. A Meta lembra que acelerar a execução sem medir confiabilidade e carga humana pode fragilizar a organização. Nada demonstra ainda um desaparecimento generalizado da gestão por causa dos agentes.

Mesmo assim, a mudança pode ser profunda. Um gestor seria menos frequentemente uma passagem obrigatória para obter informação ou aprovação. Poderia dedicar mais atenção à coerência, às situações excepcionais e ao desenvolvimento das pessoas. Em outros casos, algumas funções voltariam diretamente às equipes; outras seriam incorporadas ao software ou concentradas nas mãos de poucos decisores.

Talvez não seja a gestão que esteja se tornando inútil. É a hierarquia como mecanismo obrigatório de coordenação que começa a ser questionada.

Para um desenvolvedor, a consequência é concreta. Quando o trabalho passa por menos intermediários, ele não recebe apenas mais recursos para produzir. Encontra mais diretamente as necessidades, as escolhas e os resultados pelos quais será preciso responder.

É o movimento descrito ao longo deste primeiro livro: a capacidade de agir aumenta e, com ela, a necessidade de compreender, escolher e assumir as consequências.

Essa mudança não garante decisões melhores nem organizações mais humanas. Mas nos obriga a concebê-las de outra maneira: compartilhar o contexto, deixar claro quem detém o poder, preservar a transmissão da experiência e assumir a responsabilidade pelos resultados.

A questão não é apenas saber o que a IA agora nos permite produzir, mas que tipo de empresa queremos construir em torno dessa nova capacidade.

No último episódio, voltaremos à Malorean Corp. para ver como seus dirigentes e suas equipes vivem, para além do organograma, as escolhas que fizeram.

Bibliografia

[1] Le PO n’est pas un métier, c’est une fonction — et l’IA n’y change rien, elle accélère — Malorean

- URL: **https://malorean.net/articles/2026-08-03-le-po-nest-pas-un-metier-cest-une-fonction.pt-br.html**
- Ensaio do autor que separa a função útil de um cargo cuja atividade principal é intermediar decisões e execução.

[2] Handbook First — GitLab

- URL: **https://handbook.gitlab.com/handbook/company/culture/all-remote/handbook-first/**
- Apresenta a documentação como referência compartilhada, com discussão aberta de mudanças e responsáveis humanos pelas decisões.

[3] Comment prendre des décisions dans un environnement de travail asynchrone — Alan

- URL: **https://alan.com/fr-fr/blog/healthy-business/a/decisions-asynchrones-methode-alan**
- Explica como decisões são discutidas por escrito, quem decide, quem responde e quem deve ser consultado ou informado.

[4] The 37signals Guide to Internal Communication — 37signals

- URL: **https://37signals.com/how-we-communicate**
- Descreve comunicação majoritariamente assíncrona, trocas escritas e preservação do contexto dos projetos.

[5] The effects of remote work on collaboration among information workers — Microsoft Research

- URL: **https://www.microsoft.com/en-us/research/publication/the-effects-of-remote-work-on--collaboration-among-information-workers/**
- Analisa redes de colaboração após a adoção do trabalho remoto e o risco de maior isolamento entre grupos.

[6] AI Automates Routine Management and Raises the Value of Leadership — HEC Paris

- URL: **https://www.hec.edu/en/dare/tech-ai/ai--automates-routine-management-and-raises-value--leadership**
- Análise exploratória das tarefas gerenciais baseada em entrevistas e avaliações, sem medida causal de perda de cargos.

[7] The Cybernetic Teammate: A Field Experiment on Generative AI Reshaping Teamwork and Expertise — NBER

- URL: **https://www.nber.org/papers/w33641**

- Experimento na P&G sobre IA em tarefas de inovação de Produto e combinação de competências de áreas diferentes.

[8] Shifting Work Patterns with Generative AI — NBER

- URL: **https://www.nber.org/papers/w33795**
- Estudo randomizado com 66 empresas: economia de tempo individual sem mudança significativa na composição das tarefas em seis meses.

[9] Chairman's Letter — Bayer Annual Report 2025

- URL: **https://reports.bayer.com/annual-report-2025/en/to-our-stockholders/chairmans-letter.html**
- A Bayer relata forte redução de níveis e cargos de gestão no DSO, iniciado antes dos agentes de IA.

[10] From Hierarchy to Intelligence — Sequoia Capital / Block

- URL: **https://sequoiacap.com/article/from--hierarchy-to-intelligence**
- Jack Dorsey e Roelof Botha descrevem a visão de coordenação por sistema compartilhado, com DRIs e player-coaches.

[11] Mark Zuckerberg had a bold plan to replace Meta staff with AI. Here's how it imploded — Reuters

- URL: **https://www.timeslive.co.za/news/sci-tech/2026-08-26-special-report-mark-zuckerberg-had--a-bold-plan-to-replace-meta-staff-with-ai-heres--how-it-imploded/**
- Investigação da Reuters reproduzida pelo TimesLIVE sobre tensões organizacionais e técnicas no projeto de reorganização da Meta.

[12] How team organization influences the ability to solve automation failures — AI & Society

- URL: **https://link.springer.com/article/10.1007/s00146-025-02761-5**
- Estudo experimental com duplas auto-organizadas e hierárquicas diante de falhas de automação, destacando a comunicação.

[13] Using AI More Does Not Reassure Workers, Managers Do — Gallup

- URL: **https://www.gallup.com/workplace/713231/ai--not-reassure-workers-managers-do.aspx**
- Pesquisa sobre preocupação com a IA, respeito e apoio percebido no trabalho.

[14] L'IA ne devrait pas seulement servir à produire dix fois plus. Elle devrait aussi nous obliger à comprendre dix fois mieux ce que nous produisons — Malorean

- URL: **https://malorean.net/articles/2026-07-23-lia--produire-dix-fois-plus-comprendre-dix-fois-mieux.pt-br.html**
- Reflexão do autor sobre oportunidades de aprendizagem perdidas quando a IA assume depuração e correções que antes formavam desenvolvedores juniores.

[15] The distinct effects of Information Technology and Communication Technology on firm organization — NBER

- URL: **https://www.nber.org/papers/w14975**
- Distingue os efeitos do acesso à informação e da comunicação com superiores sobre a autonomia das decisões locais.

[16] Algorithmic Management: From Technology to Politics and Theory — Weizenbaum Journal of the Digital Society

- URL: **https://ojs.weizenbaum-institut.de/index.php/wjds/article/view/4_3_9/163**
- Examina funções de gestão e avaliação exercidas por sistemas algorítmicos e a presença de supervisão humana.

[17] Pourquoi les organisations préfèrent les décisions molles — Malorean

- URL: **https://malorean.net/articles/2026-06-02--pourquoi-les-organisations-preferent-les--decisions-molles.pt-br.html**
- Ensaio sobre consensos aparentes que adiam escolhas difíceis, escondem concessões e dificultam atribuir responsabilidades.

11 — Malorean Corp. — O tempo recuperado

Dois meses depois da demonstração do módulo de ausências, Hubert Félix Duca abriu a reunião de segunda-feira sem perguntar como andavam os desenvolvimentos.

— Então, o que aprendemos na semana passada?

Alice compartilhou a tela.

Diante dela estavam a síntese dos últimos retornos dos clientes, os resultados dos experimentos em andamento e algumas decisões ainda abertas. Durante a noite, um agente preparara um primeiro agrupamento das informações. Ela o revisara pela manhã, descartara várias associações enganosas e acrescentara duas perguntas que ele não teria conseguido formular sozinho.

— Na rastreabilidade, nosso cliente piloto consegue encontrar as fabricações afetadas muito mais rapidamente quando os dados estão completos — começou. — Mas ainda existem casos históricos em que precisamos mostrar claramente nossos limites. Vamos testar uma nova apresentação com o responsável pela qualidade.

Hubert concordou.

— E os outros clientes?

— Cedo demais para concluir — Alice respondeu. — É justamente por isso que não ampliamos a implantação.

Bob tomou a palavra. Trabalhava em outro conjunto de iniciativas de Produto e acabara de conversar com as equipes de pré-vendas e suporte.

— Também temos três pedidos que parecem semelhantes — disse. — O agente os agrupou numa única ideia de funcionalidade. Mas existem pelo menos dois problemas diferentes por trás deles. Quero esclarecer o público-alvo antes de decidir investir.

— De quantos desenvolvedores você precisa? — Hubert perguntou.

Bob sorriu.

— Nenhum, por enquanto.

Hubert devolveu o sorriso.

Bob olhou uma notificação e acrescentou:

— Aliás, o suporte começou ontem a testar uma nova classificação dos pedidos recorrentes. Alguém da equipe a preparou com um agente, usando cópias anonimizadas, para ver se as informações chegam mais rápido às pessoas certas.

— Quem iniciou isso? — Hubert perguntou.

— O suporte. Alice confirmou que o escopo continuava interno e reversível, com uma pessoa verificando os resultados.

— Você estava sabendo?

— Da ideia, sim. Não dos detalhes do teste.

Hubert teve o impulso de perguntar por que ninguém o avisara antes de começar. Conteve-se. Haviam combinado justamente que iniciativas limitadas e reversíveis não exigiriam mais sua aprovação.

— Contem-me se funcionar — disse finalmente. — E, se alguém precisar cuidar da manutenção, quero saber quem.

Ele não precisara autorizar o teste. Continuava responsável pelos recursos que destinariam a uma possível ampliação.

Não fazia muito tempo, essa resposta teria parecido estranha numa reunião de Produto. O objetivo era principalmente decidir o que os desenvolvedores deveriam fazer e em qual ordem. Agora Alice e Bob dedicavam mais tempo a escolher problemas, verificar a coerência das iniciativas e intervir quando alguma decisão ultrapassava os limites definidos.

Seus títulos não tinham mudado. Nos documentos administrativos, continuavam sendo Product Managers. Mas o trabalho cotidiano já não parecia exatamente o mesmo de dois meses atrás.

Talvez a mudança mais visível fosse aquilo que já não chamava a atenção.

Um desenvolvedor conversava com um cliente. Outro propunha um Mini PRD para melhorar seu módulo. Uma vendedora criava uma variante de demonstração no ambiente preparado para isso. Alguém do suporte sinalizava diretamente uma incoerência numa regra de Produto.

Isso já não provocava grandes discussões sobre fronteiras entre profissões.

A Malorean não anunciara que todo mundo poderia fazer tudo. A empresa simplesmente deixara de supor que uma ação precisaria passar obrigatoriamente por quem tradicionalmente ocupava determinado cargo.

As duas equipes de cinco desenvolvedores continuavam existindo. A maioria dos integrantes agora assumia alguns assuntos de ponta a ponta: compreender o problema, propor o escopo, conversar com as pessoas

envolvidas, desenvolver, testar e acompanhar os resultados.

Alguns adotaram essa forma de trabalhar mais rapidamente. Dois desenvolvedores ainda preferiam receber bastante acompanhamento na definição do problema. Outro era excelente em desafios técnicos complexos, mas não tinha vontade de multiplicar reuniões com clientes. Ninguém tentava transformá-los à força em cópias de Lucas.

Também continuavam existindo tickets.

Eles serviam para coordenar uma intervenção, registrar uma tarefa específica ou acompanhar uma correção. Mas ninguém mais tentava colocar ali toda a razão de ser de um assunto, todas as decisões anteriores e todas as informações úteis a um agente.

Em vez disso, cada responsável mantinha um espaço de contexto na ferramenta de documentação compartilhada. Ali ficavam os objetivos, as decisões, as fontes, os riscos, as perguntas abertas e os links para informações já existentes: tickets, propostas comerciais, retornos do suporte, anotações de reuniões, testes e até conversas antigas recuperadas.

Não era necessário copiar tudo. O essencial era que alguém pudesse entender por que uma decisão fora tomada, onde verificar uma informação e quem tinha autoridade para mudá-la.

O primeiro documento de Alice servira de modelo.

Bob adotara o método, depois os desenvolvedores e várias pessoas de suporte, vendas e implantação. Muitos descobriram, como Alice, que podiam passar uma hora explicando oralmente suas escolhas ao ChatGPT, em vez de gastar dias formalizando tudo sozinhos.

Nem tudo o que estava na cabeça das pessoas se tornara milagrosamente acessível. Mas parte das razões, exceções e conhecimentos tácitos começava a ser compartilhada.

O trabalho remoto tornara esse movimento mais natural. Desde a pandemia, a Malorean já aprendera a escrever mais para que colegas espalhados pela França trabalhassem sem esperar por conversas de corredor. A IA acelerara a extração e a organização do conhecimento. Não inventara a necessidade de compartilhá-lo.

Alice abriu uma página chamada **Decisões de Produto — Regras compartilhadas**.

— Tivemos outra confusão na semana passada — explicou a Bob. — A documentação de uma equipe usava “validado” para dizer “revisado tecnicamente”. O suporte usava a palavra para dizer “comportamento confirmado com o cliente”.

Bob suspirou.

— Essa palavra nunca significou a mesma coisa para todo mundo.

— Justamente. Acrescentei um link para a decisão que tomamos depois da história dos lotes.

Algumas semanas antes, Lucas e Thomas quase deixaram coexistir dois comportamentos incompatíveis entre rastreabilidade e expedição. As mudanças haviam sido interrompidas nos testes. Desde então, as equipes aprenderam a desconfiar de termos que pareciam óbvios num módulo, mas mudavam de significado em outro.

O contexto ajudara. Também criara novos problemas.

Algumas páginas estavam desatualizadas. Decisões provisórias tinham sido copiadas como se fossem regras definitivas. Certa vez, um agente recuperara uma antiga hipótese comercial como se comprometesse a empresa.

Precisaram estabelecer uma disciplina: cada contexto identificava o responsável, as fontes, as decisões confirmadas e os pontos ainda incertos. Mudanças estruturantes deveriam ser registradas; informações sensíveis não eram disponibilizadas indiscriminadamente a todas as ferramentas.

— Poderíamos ter evitado tantas páginas — Bob observou.

— Talvez. Mas as contradições existiam antes também. Só que ficavam na nossa cabeça e ninguém conseguia vê-las.

Alice releu a última alteração do colega.

— Aprovo o princípio. Não essa frase sobre os clientes alemães: ainda não sabemos o suficiente.

Bob corrigiu a linha.

Muitas vezes, o trabalho de Guardião era isso. Não aprovar cada tela, mas impedir que uma hipótese se tornasse regra sem querer.

Na outra equipe, Thomas acabara de conversar com um desenvolvedor recém-chegado.

Havia passado uma hora com ele examinando o código histórico do ERP. O desenvolvedor usara um assistente

para entender um mecanismo antigo de folha de pagamento e depois perguntara por que uma exceção parecia contrariar a regra geral.

Thomas encontrara a razão da exceção numa entrega antiga.

— Se você se limitar ao resumo, pode achar que precisa limpar o código — disse. — Mas, antes de limpar, precisamos entender o que estamos prestes a eliminar.

— Você pode registrar isso no contexto da equipe?

— É o que estou fazendo.

Thomas compartilhou a nova nota. Incluíra a fonte, a explicação e dois casos de teste.

Ainda não entregava a produção de suas alterações técnicas a agentes como Lucas fazia. Seu ceticismo não desaparecera. No entanto, usava de bom grado os novos espaços de documentação e colaborava mais com os colegas para transmitir sua experiência.

Vivien, que acompanhara a integração do novo desenvolvedor, ligou para Thomas pouco depois.

— Como vão as coisas?

— Bem. Ele é curioso. Faz perguntas antes de apagar alguma coisa.

— Que bom.

— E finalmente reservei tempo para escrever por que certos controles existem.

Vivien sorriu.

— Então mudou de ideia sobre IA?

— Não. Mudei de ideia sobre guardar essas explicações só para mim.

Ele fez uma pausa.

— Mas isso exige tempo. Se me pedirem para acompanhar os colegas e continuarem enchendo meus dias com desenvolvimento, não vai funcionar.

— Já falei com Hubert. Reservamos tempo para mentoria no planejamento. Não é algo que faremos apenas quando todo o resto estiver concluído.

Vivien sabia do que falava. Durante anos, ouvira equipes anunciarem que formariam os recém-chegados “assim que sobrasse um pouco de tempo”. Esse momento nunca chegava.

Conversaram mais um pouco sobre as necessidades do novo colega. Depois Vivien encerrou a chamada.

Não abrira o painel de entregas uma única vez.

Na hora do almoço, Hubert pediu a Vivien que entrasse numa videochamada.

— Tenho uma pergunta — começou. — Quanto tempo você passa hoje fazendo o que fazia no ano passado?

Vivien refletiu.

— Não o suficiente para justificar meu cargo antigo em tempo integral, se é isso que você quer saber.

Hubert não respondeu imediatamente.

Durante muito tempo, Vivien fora o ponto por onde passava a coordenação das duas equipes técnicas. Gerenciava entregas, dependências, parte das urgências e as conversas entre desenvolvedores e responsáveis de negócio. Quando a organização era menos autônoma, esse trabalho ocupava facilmente seus dias.

Agora já não era assim.

Cada iniciativa tinha um responsável claramente identificado. Desenvolvedores tomavam boa parte das decisões locais. As informações circulavam sem que ele próprio precisasse transmiti-las. As ferramentas preparavam relatórios e sinalizavam dependências, embora suas afirmações ainda precisassem ser verificadas.

Os problemas humanos, os conflitos e a mentoria não desapareceram. Vivien continuava ajudando a resolvê-los.

Mas já não precisavam estar concentrados num cargo dedicado à supervisão diária das entregas.

— Não me incomoda ter menos reuniões inúteis — continuou. — Pelo contrário. Mas não passei todos esses anos aprendendo a coordenar equipes para virar alguém que verifica se um agente preencheu uma planilha direito.

— Não vou pedir isso — Hubert respondeu.

Haviam preparado uma transição. As decisões técnicas importantes caberiam aos Guardiões de cada domínio. As pessoas de referência para acompanhamento teriam tempo reservado. Hubert continuaria responsável pelas decisões de carreira e pelos problemas humanos que não podiam ser deixados a um documento. Vivien cuidaria da transição dos últimos assuntos de coordenação.

Nem tudo fora perfeito. Precisaram resolver uma situação em que duas pessoas achavam que a outra acompanhava um funcionário com dificuldades. Outra equipe também esperara demais por uma decisão porque ninguém havia sido designado responsável.

Em cada caso, corrigiram as responsabilidades e a maneira de torná-las visíveis.

— Tem uma coisa que não quero — disse Hubert. — Não quero eliminar seu cargo e fingir que o trabalho que você fazia não tinha valor.

— Eu também não.

— Então precisamos descobrir onde sua experiência pode ser útil agora.

Vivien olhou para a xícara.

— Acho que tenho uma ideia.

Alguns dias antes, a esposa voltara a falar da Alemanha. Ela era alemã. Havia algum tempo, os dois conversavam sobre a possibilidade de morar lá. Sem urgência: tinham sua vida na França, seus hábitos e amigos. Mas a ideia voltava com frequência, cada vez com mais detalhes.

8

Vivien sempre adiara a decisão.

Gostava demais do trabalho na Malorean. Conhecia as equipes, sua história, o software e os clientes. Sair lhe daria a sensação de abandonar algo que ajudara a construir.

Mas também gostava de desafios. Naquela etapa da carreira, surpreendia-se desejando enfrentar um novo.

No sábado anterior, os dois tinham olhado anúncios de imóveis na Alemanha sem admitir exatamente o que

estavam fazendo. Depois, a esposa fechara o computador.

— Sabe — disse —, não precisamos continuar falando disso por mais dez anos.

Vivien rira.

Mas, na segunda-feira, ainda pensava no assunto.

— Alemanha? — Hubert repetiu.

Vivien assentiu.

— Falamos disso nas reuniões de estratégia há anos. Temos alguns clientes lá, mas são principalmente filiais de grupos franceses. Ainda não aprendemos de verdade a vender nosso ERP para empresas alemãs.

— É verdade.

— E todo mundo diz que podemos entrar nesse mercado quando o produto estiver pronto. Só que nunca saberemos o que falta se não conversarmos seriamente com as empresas de lá.

Hubert escutava com atenção.

— Você gostaria de cuidar do desenvolvimento desse mercado?

— Sim. Sem fingir que virei especialista no mercado alemão da noite para o dia. Conheço nosso produto,

nossas restrições, nossas equipes e a maneira como os clientes o implantam. Sei reunir pessoas em torno de um problema complicado. Vou precisar aprender o resto.

— Você já trabalhou com vendas?

— Não como Julie ou Philippe. E não tenho intenção de ensinar a eles como fazer seu trabalho. Quero trabalhar com os dois, com a equipe de implantação e com nossos especialistas de negócio para entender as necessidades das empresas alemãs e ver se podemos atendê-las.

Vivien hesitou.

— Minha esposa é alemã. Faz um tempo que conversamos sobre morar lá. Seria uma oportunidade de fazer algo que realmente quero, sem sair da Malorean.

Hubert ficou em silêncio por um momento.

— Você sabe que não é só um projeto de tradução?

— Claro. O ERP envolve regras e processos que variam de país para país. Traduzir as telas não basta. Precisamos saber o que podemos realmente prometer e o que teremos de desenvolver, adaptar ou recusar.

— E podemos errar.

— Provavelmente vamos errar em algumas coisas. Prefiro descobrir isso com clientes a investir um ano em funcionalidades que ninguém quer.

Hubert olhou sua lista de projetos.

Durante muito tempo, a Alemanha fora uma ambição para depois. Primeiro precisavam terminar os projetos atrasados, contratar mais desenvolvedores, liberar capacidade para adaptar o ERP, preparar demonstrações e treinar as equipes.

Algumas dessas dificuldades não tinham desaparecido. Mas várias haviam mudado de escala.

Preparar uma demonstração exigia menos esforço. Era possível testar variações do produto sem envolver sistematicamente desenvolvedores. O suporte e a pré-venda produziam documentos e compartilhavam conhecimentos com mais facilidade. Os desenvolvedores tinham mais capacidade para explorar uma possibilidade limitada.

Isso não significava que estivessem sem trabalho. Significava que podiam escolher dedicar parte de sua energia a algo diferente de apenas absorver a fila de pedidos franceses.

— Prepare uma proposta para a primeira etapa — Hubert disse. — Não uma promessa de entrada no mercado. Reuniões, as perguntas que queremos responder, as

competências de que você vai precisar e uma maneira de decidir se continuamos.

Vivien sorriu.

— Um Mini PRD para a Alemanha?

— Não zombe de mim. Se conseguimos definir um assunto técnico dessa maneira, pelo menos podemos formular as hipóteses de um projeto de mercado.

— Está bem. Mas quero manter o direito de descobrir que estávamos errados.

— Estou contando com isso.

Na semana seguinte, Hubert reuniu Alice, Bob, Vivien, Julie e Philippe.

Apresentou a proposta sem começar perguntando quantas funcionalidades precisariam desenvolver.

— Agora temos uma capacidade que não tínhamos há dois meses. Gostaria de dedicar parte dela a compreender o que significaria uma presença real no mercado alemão.

Julie foi a primeira a reagir.

— Já posso preparar demonstrações em alemão. Também podemos começar a contatar clientes

potenciais rapidamente. Com IA, conseguimos adaptar os materiais em algumas horas.

— Os materiais, talvez — Philippe disse. — Mas um dirigente alemão que compra um ERP não escolhe um fornecedor apenas porque a apresentação está no idioma dele.

Julie cruzou os braços.

— Não foi isso que eu disse.

— Eu sei. Só estou dizendo que precisamos compreender como escolhem seus fornecedores, quem consultam e que nível de confiança exigem. Ainda não conhecemos bem esse mercado.

Vivien interveio antes que a conversa se transformasse num debate sobre ferramentas.

— É justamente isso que quero aprender. Julie, vamos precisar da sua capacidade de preparar abordagens e demonstrações rapidamente. Philippe, vamos precisar da sua experiência com grandes contas e das conversas que não se resolvem com uma tela bonita. E os especialistas em implantação conhecem as perguntas que os clientes às vezes descobrem tarde demais.

Alice tomou a palavra.

— Do lado de Produto, teremos de ser muito claros sobre os limites. Podemos explorar rapidamente, mas não prometer funcionalidades regulatórias que ainda não validamos.

Bob assentiu.

— Também teremos de escolher quais empresas queremos compreender primeiro. A Alemanha não é um único tipo de cliente. Se tentarmos atender todo mundo, vamos repetir os erros que aprendemos a evitar aqui.

Hubert olhou os rostos na tela.

Pedira a Vivien que propusesse uma abordagem. Ninguém solicitara a criação imediata de uma equipe permanente, a reserva de todo o desenvolvimento ou uma data de abertura comercial.

Mesmo assim, várias pessoas já começavam a identificar o que podiam oferecer.

Isso se tornara um hábito na Malorean.

— Muito bem — Hubert disse. — Eis minha decisão: vamos financiar uma primeira fase de exploração. Vivien será responsável. Ela deverá nos mostrar quais problemas das empresas alemãs já sabemos resolver, quais exigem adaptação e quais não queremos assumir. Depois decidiremos se investimos mais.

Acrescentou:

— E ninguém será colocado nesse projeto só porque uma ferramenta economizou seu tempo. Vamos avaliar o que cada pessoa realmente consegue liberar sem abandonar os clientes que já atendemos.

Vivien anotou o último ponto. Passara anos demais lidando com planejamentos para não saber que uma capacidade liberada no papel nem sempre estava disponível de fato.

Nos dias seguintes, conversou com toda a equipe de pré-vendas e implantação.

Parte do trabalho deles mudara desde que as demonstrações passaram a ser preparadas com configurações reutilizáveis e que os resumos das conversas com clientes se tornaram mais fáceis de produzir. Duas pessoas explicaram que agora passavam menos tempo recriando os mesmos materiais e gostariam de participar das primeiras entrevistas na Alemanha.

— Posso ajudar a preparar as perguntas sobre implantação — uma delas ofereceu. — E acompanhar algumas reuniões à distância.

— E seus clientes franceses? — Vivien perguntou.

— Não posso largar tudo. Mas consigo reservar tempo se colocarmos isso no planejamento.

Outra pessoa, envolvida em várias implantações delicadas, disse que não tinha disponibilidade.

Vivien não insistiu.

Prepararam uma divisão provisória: alguns horários reservados para compreender as expectativas das empresas alemãs, reuniões com filiais que já eram clientes quando fosse útil e nenhuma promessa de dedicação permanente antes de medir o esforço real.

O tempo economizado não se transformara sozinho em crescimento. Ainda era preciso decidir o que fazer com ele e aceitar não preencher imediatamente cada hora recuperada com os pedidos de ontem.

À noite, Alice recebeu uma mensagem de Lucas.

Vi o projeto da Alemanha. Já temos um mecanismo de configuração que poderia ajudar a testar diferenças nos processos. Posso preparar um ambiente exploratório sem mexer no produto padrão. Interessa?

Ela releu a mensagem.

Dois meses antes, uma proposta parecida provavelmente teria provocado uma discussão sobre a necessidade de um desenvolvedor esperar até receber uma tarefa. Agora, a primeira pergunta era mais simples: seria útil para a estratégia e dentro de quais limites ele poderia avançar?

Alice abriu o contexto da primeira fase alemã. Vivien já o preenchera com hipóteses, responsáveis, limites e questões em aberto.

Ela respondeu:

Sim, desde que você defina o escopo com Vivien e que não transformemos uma hipótese de negócio em comportamento padrão sem validação. Comece pelo que queremos aprender, não pelo que sabemos construir.

Lucas respondeu quase imediatamente.

Entendido. Vou preparar um Mini PRD.

Alice sorriu.

Lembrou-se da sexta-feira em que ele lhes mostrara um módulo de ausências desenvolvido em dois dias. Naquele momento, todos tentavam entender como conseguira produzir tanto, tão rápido.

Naquela noite, ninguém mais se surpreendia com sua capacidade de criar uma demonstração rapidamente.

A questão era por que precisariam dela.

Vivien terminou o dia mostrando à esposa as datas dos primeiros encontros previstos com empresas na Alemanha.

Ainda não fechara nenhum contrato. A Malorean não abrira um escritório nem adaptara o ERP a todas as exigências do mercado. Precisariam descobrir os hábitos locais, verificar restrições jurídicas e regulatórias e aprender a conquistar a confiança de empresas que não os conheciam.

Havia muito mais incógnitas do que certezas.

— Isso dá medo? — perguntou a esposa.

— Um pouco.

— Você parecia tão seguro quando coordenava suas equipes.

Vivien sorriu.

— Passei anos aprendendo aquele trabalho.

— E agora?

Olhou o documento que começara a preparar para a primeira visita.

Perguntas. Hipóteses. Nomes de pessoas para conhecer.

Uma primeira lista do que ainda não sabia.

— Agora — respondeu —, tenho uma coisa nova para aprender.

Ela se aproximou para ver a tela.

— E nós, como ficamos?

— Podemos começar a conversar seriamente sobre nossa mudança.

Ela colocou a mão no ombro dele.

Vivien pensou no antigo painel de entregas, nas reuniões de segunda e em todo o trabalho que permitira manter os projetos juntos. Gostara daquilo. Não lamentava aqueles anos.

Também não lamentava ter aceitado que as equipes já não precisavam dele da mesma maneira.

Na tela, o primeiro encontro na Alemanha estava confirmado.

Para a Malorean, aquilo ainda não era uma conquista nem um sucesso.

Era um começo.

12 — Conclusão

Profissões com fronteiras mais flexíveis, responsabilidades mais claras

Os cargos já não definem exatamente o que cada pessoa é capaz de fazer. Um desenvolvedor pode investigar uma necessidade, um Product Manager pode construir uma solução, um profissional de vendas pode criar um protótipo. A IA abre caminhos antes reservados a outras especialidades. Isso não torna seus conhecimentos intercambiáveis.

Já não somos responsáveis apenas pelo que executamos pessoalmente. Podemos assumir uma iniciativa que atravessa várias áreas, enquanto os Guardiões zelam pela coerência de seus respectivos domínios. A capacidade de agir passa de uma função para outra; a responsabilidade não pode se diluir nesse movimento.

O risco é duplo: continuar dizendo “isso não é função minha” ou supor que “se consigo fazer, posso decidir sozinho”. Nem a defesa dos territórios profissionais nem sua eliminação bastam para organizar o trabalho.

A autonomia depende de um contexto compartilhado

Mais liberdade só faz sentido quando cada pessoa dispõe das informações necessárias para julgar. Sem contexto compartilhado, uma decisão pode parecer correta localmente e contrariar uma necessidade, uma escolha ou uma restrição mais ampla.

Documentar, portanto, não é acumular atas de reunião. É tornar acessíveis o problema a resolver, o objetivo, as hipóteses, as razões das decisões, os limites e as regras de escalonamento. Explicitar esses elementos permite agir sem precisar encontrar, a cada etapa, a pessoa que detém a informação. O contexto se torna uma ferramenta de decisão, tanto para pessoas quanto para agentes.

É aí que muda a postura dos Guardiões. Eles não fiscalizam todas as iniciativas: tornam os limites compreensíveis, zelam pela coerência e intervêm quando uma decisão ultrapassa esse âmbito. Sem salvaguardas, a liberdade pode virar caos; se tudo precisa subir para aprovação, ela não passa de uma promessa.

A gestão também evolui. Parte do acompanhamento e da coordenação pode ser feita de outra maneira. Mas ainda existem pessoas a apoiar, experiências a transmitir, conflitos a resolver e escolhas difíceis a assumir. Essas

responsabilidades não desaparecem junto com as tarefas que antes lhes davam forma.

Fazer mais daquilo que importa

Será grande a tentação de medir essa transformação pelo número de funcionalidades produzidas ou de horas economizadas. A IA realmente pode nos fazer produzir mais. Mas preencher imediatamente o tempo recuperado com novas tarefas significaria apenas acelerar o funcionamento antigo, sem questionar sua finalidade.

Também podemos dedicar essa capacidade a compreender melhor as necessidades, testar nossas hipóteses, melhorar o que já existe e abandonar o que não cria valor. Nada disso acontece automaticamente. É preciso escolher onde investir a atenção e os recursos liberados.

Na Malorean Corp., Vivien não descobre apenas que seu trabalho anterior passou a ocupar menos espaço. Descobre que pode colocar sua experiência a serviço de outro problema, sem garantia de sucesso. O progresso não está só no que foi automatizado, mas também no que essa transformação tornou possível.

No início da história, todos se surpreendiam com a rapidez com que Lucas construía software. No fim, a

pergunta passou a ser: por que construir, para quem e com quais consequências?

O propósito da IA não é nos fazer produzir cada vez mais. É nos dar os meios para fazer mais daquilo que realmente importa.