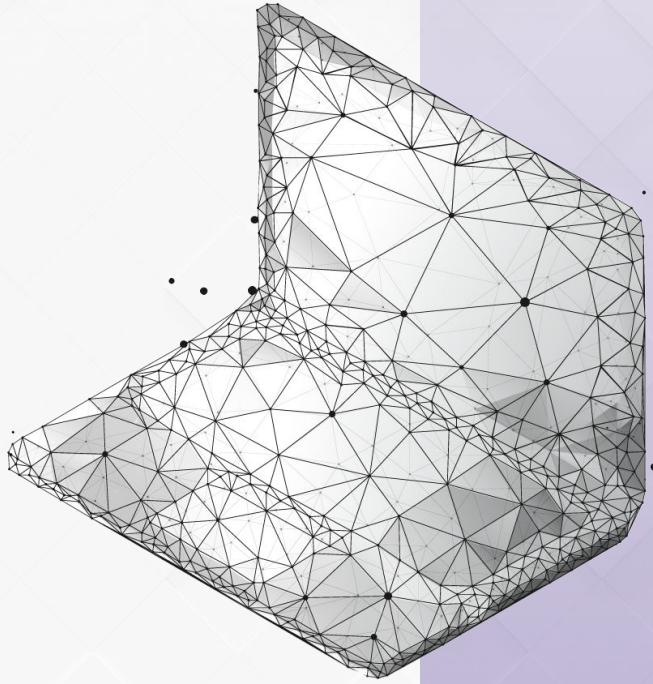


Parte 3



Avaliando o Alinhamento da Tecnologia com o Product-Market Fit





Overview

- Nesta seção, vocês aprenderão a avaliar como a tecnologia de uma startup apoia a busca pelo product-market fit e a visão de longo prazo da empresa.
- O foco será entender se a tecnologia da empresa está configurada para atender a uma necessidade de mercado real, além de identificar se há um excesso de complexidade que possa prejudicar o crescimento.





Revisitando a última aula



feedbacks da última aula:

4.4



Discussão de dever de casa:

Buscar startups do seu portfólio investido ou não e investigar as escolhas de tecnologia, por que elas foram feitas e se tiveram impacto positivo ou negativo para o crescimento da startup.



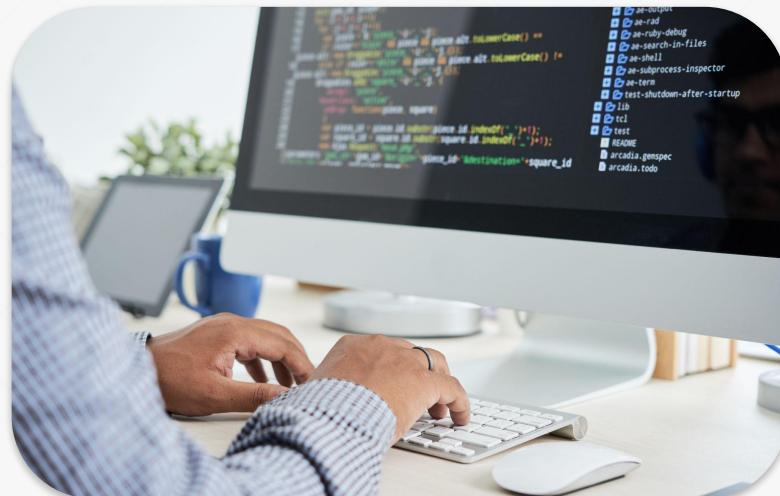
Discussão de dever de casa:

Analisar uma startup do seu portfólio investido ou não com as perguntas de inspeção e análise de tecnologias, e evidenciar os green, yellow e red flags, e a qualidade das tecnologias escolhidas.



🏁 Introdução

O ciclo de desenvolvimento de software



É importante dar um passo atrás e tentar visualizar o ciclo de desenvolvimento de software de forma macro e então entender cada etapa do ciclo da perspectiva de um desenvolvedor, para assim tangibilizar todo processo de execução que permite transformar um roadmap em realidade.

Há vários modelos de ciclos de vida de desenvolvimento, mas todos eles muito similares. O modelo que vamos abordar possui 6 etapas macro.

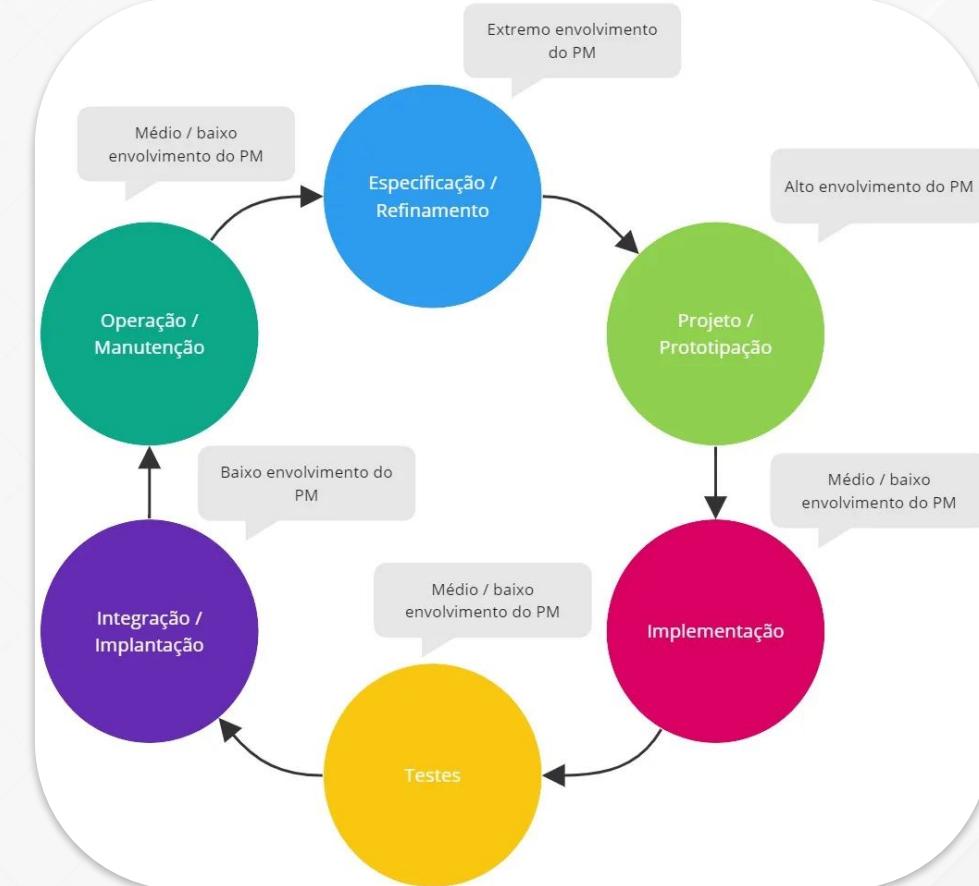
- **Especificação / Refinamento:** definição do escopo e detalhamento das especificações funcionais e não funcionais
- **Projeto / Prototipação:** criação de modelos e documentações para representar e comunicar a solução aos implementadores
- **Implementação:** implementação da solução, parte por parte, até estar pronta para implantação
- **Testes:** garantia de que as funcionalidades funcionam como especificadas, de forma manual ou automatizada
- **Integração / Implantação:** implantação do software em ambiente produtivo e exposição aos usuários
- **Operação / Manutenção:** monitoramento e endereçamento de problemas após a implantação e uso da funcionalidade



Essas etapas acontecem de forma cíclica e concorrente, onde vários itens podem estar em diferentes etapas ao mesmo tempo. **Em frameworks ágeis como Scrum, esses ciclos podem ser chamados de Sprints.**

Times que mantêm consistência nos ciclos de desenvolvimento têm maior chance de sucesso.

Sua adaptabilidade às mudanças de mercado e necessidades dos usuários é essencial. Essa agilidade é vital para startups buscando product-market fit e crescimento eficiente.



Análise de Roadmap

Análise VUVE



Para interação

- O que vocês avaliam quando são apresentados um roadmap de produto em um pitch deck ou reunião de conselho?



⚠️ O Problema com Roadmaps de Produto

Os clientes não estão tão entusiasmados com a idéia como nós estamos, então eles escolhem não pagar por eles (**o valor não existe**).

Às vezes, os clientes querem usar, mas quando usam é tão complicado que é mais trabalho do que vale, o que resulta no mesmo outcome acima (**a usabilidade não existe**).

Às vezes, o problema é que como os clientes podem ter amado a ideia, mas pode ser mais complicado para construí-la do que imaginamos, e simplesmente não temos o tempo e recursos para entregá-la (**sem executabilidade**).

E, às vezes o problema é que encontramos sérios problemas legais, financeiros ou outras restrições de negócios que impedem o lançamento da solução (**a viabilidade não existe**).

Isso implica que avaliar um roadmap de produto é, primariamente, avaliar o time de produto, principalmente que:

- Entende profundamente o que **problema** que está sendo resolvido
- Entende profundamente o **usuário** que sofre do problema em questão
- Entende profundamente os **riscos** para o negócio em lançar essa solução
- Entende profundamente os **objetivos** que o negócio quer alcançar com essa solução
- Consegue **construir a solução no custo e prazo estimados**, e iterar rápido até atingir os objetivos esperados

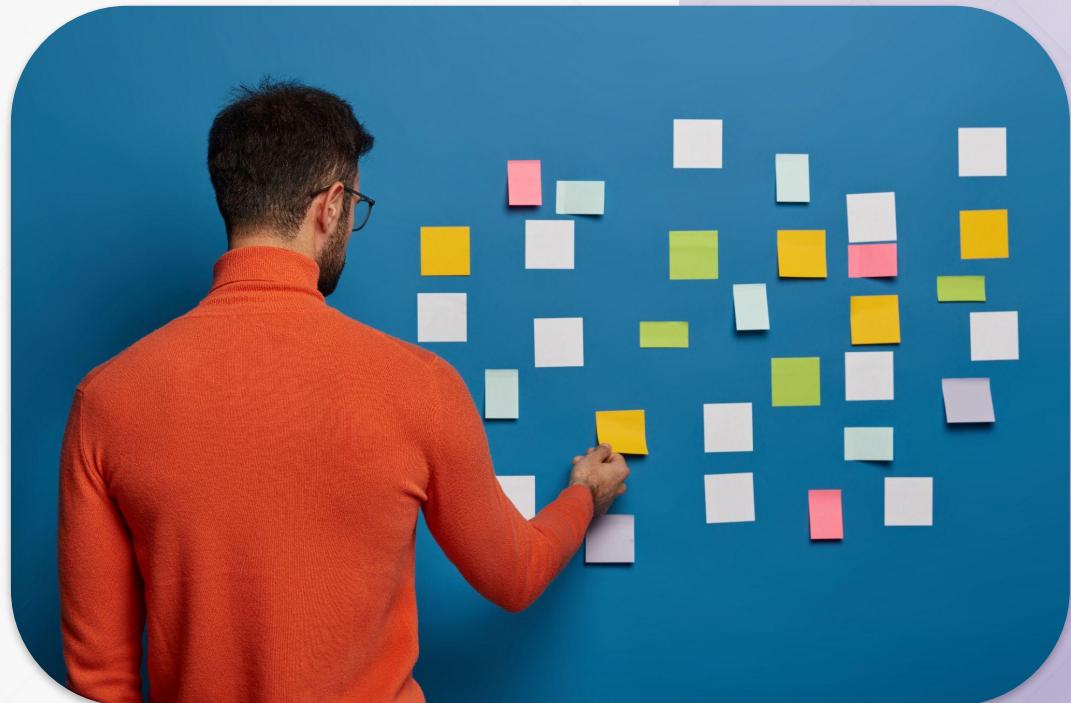




Analisando um Roadmap

A estrutura VUVE é um modelo que permite aos analistas avaliar o roadmap de uma empresa com uma visão prática e holística, considerando:

- Valor (User Value)
- Usabilidade (Usability)
- Viabilidade (Business Value)
- Executabilidade (Feasibility)





Valor (User Value)

O usuário ou cliente escolher usar ou comprar isso?

Avalia o **valor percebido para o usuário final**. Pergunta-se: essa funcionalidade resolve um problema real? O cliente enxerga valor nela?

Exemplo: Para uma empresa de delivery, adicionar uma funcionalidade de rastreamento em tempo real pode ser essencial para a satisfação do cliente, entregando valor claro para o usuário.

Pergunta-chave: "Esta funcionalidade realmente responde a uma necessidade do mercado ou é um 'nice-to-have' que pouco agrupa?"





Valor (User Value)

O usuário ou cliente escolher usar ou comprar isso?

Colhendo evidências

- Usando ferramentas de teste de demanda como fake-doors e landing pages permitem colher evidências de demanda muito antes (e com muito menos recursos) de qualquer funcionalidade ser construída.

[Dennis Nerush | Fake door — The MVP before the MVP | HackerNoon](#)

The screenshot shows the Buffer landing page with the headline "Tweet more consistently with buffer". Below it is a three-step process:

- 1 Choose times to tweet.
For example, 3 times a day at 9:30, 13:30 and 17:30.
- 2 Add tweets to your buffer.
Manually or with our handy browser extensions.
- 3 buffer does the rest. Relax.
We tweet for you. Just keep that buffer topped up!

A green arrow points from the bottom of this section down to the second Buffer screenshot.

The screenshot shows the Buffer landing page with the headline "Tweet more consistently with buffer". Below it is a message: "Hello! You caught us before we're ready. We're working hard to put the finishing touches onto buffer. Things are going well and it should be ready to help you with Twitter very soon. If you'd like us to send you a reminder when we're ready, just put your email in below." At the bottom are input fields for "Your email" and "Remind me". A green arrow points from the top of this section up to the first Buffer screenshot.



Valor (User Value)

O usuário ou cliente escolher usar ou comprar isso?

Colhendo evidências

- Técnicas qualitativas como entrevistas de willingness-to-pay, onde é possível colher evidências se usuários pagariam com dinheiro, reputação, tempo ou dados pela funcionalidade também são uma ótima forma de validar que o produto / funcionalidade tem sinais de valor.

[Dennis Nerush / Fake door — The MVP before the MVP | Hacker Noon](#)

The image shows three distinct sections of Buffer's website, each with a green arrow pointing to it from the main title above:

- Top Section:** A call-to-action box with the headline "Tweet more consistently with  buffer". It lists three steps:
 - 1 Choose times to tweet. (For instance, 3 times a day at 8AM, 12PM and 11PM)
 - 2 Add tweets to your buffer. (Up to 50 tweets in your buffer)
 - 3 buffer does the rest. Relax. (We tweet for you. Just check back later)A green arrow points from the main title to this section.
- Middle Section:** A pricing table comparing three plans:

Free	Standard	Max
\$0/mo	\$5/mo	\$20/mo
What? Clip your tools. Find your feet.	Who? Great for most users.	Who? Best for business & enterprise
1 tweet per day	10 tweets per day	Unlimited tweets per day
5 tweets in your buffer	50 tweets in your buffer	Unlimited tweets in buffer

A green arrow points from the main title to the "Max" plan section.
- Bottom Section:** A "Fake door" landing page with the headline "Hello! You caught us before we're ready." It includes a message: "We're working hard to put the finishing touches onto Buffer. Things are going well and it should be ready to help you with Twitter very soon. If you'd like us to send you a reminder when we're ready, just put your email in below." A green arrow points from the main title to this section.



Valor (User Value)

O usuário ou cliente escolher usar ou comprar isso?

Colhendo evidências

- Outras técnicas quantitativas como testes A/B, listas exclusivas e uso de ferramentas de analytics também podem ajudar a colher evidências de que o produto / funcionalidade possui valor para o usuário / cliente alvo.

[Optimizely | What is A/B testing? With Examples](#)





Valor (User Value)

Avaliando o roadmap:

- Os founders estão conscientes dos riscos de valor dos produtos / funcionalidades contidas no roadmap de produto?
- Eles têm um processo estruturado para validar o valor percebido antes de investir recursos no desenvolvimento? Quais foram as evidências colhidas que corroboraram para o roadmap atual?
- Quais métricas estão sendo utilizadas para medir o sucesso e o impacto das novas funcionalidades em termos de valor para o usuário?
- Existe um plano de contingência caso uma funcionalidade não entregue o valor esperado?





Usabilidade (Usability)

O usuário consegue entender como usar isso?

Refere-se à facilidade de uso da tecnologia ou funcionalidade para o usuário final. Mesmo uma funcionalidade poderosa perde seu valor se for complexa de usar ou se não for intuitiva.

Exemplo: Em aplicativos de saúde, a interface precisa ser intuitiva e de fácil acesso, especialmente em situações de uso em emergências.

Pergunta-chave: "As funcionalidades são intuitivas e facilitam o uso do produto para o usuário final?"





Usabilidade (Usability)

O usuário consegue entender como usar isso?



Colhendo evidências

Usando **protótipos** para **testes de usabilidade** com usuários reais para identificar fricções o mais cedo possível, que podem residir nas nomenclaturas, fluxos, visual designs e modelos mentais. Esses testes permitem **ajustes rápidos e iterativos** antes do lançamento final, **economizando tempo e recursos**.



Usabilidade (Usability)

Avaliando o roadmap:

- Os founders estão conscientes dos riscos de usabilidade dos produtos / funcionalidades contidas no roadmap de produto?
- Eles estão realizando testes de usabilidade regularmente? Quais foram as fricções descobertas durante os últimos testes de usabilidade?
- Existe um processo estruturado para coletar feedback dos usuários e incorporá-lo no desenvolvimento do produto?
- A equipe de design está envolvida desde o início do processo de criação de novas funcionalidades?





Viabilidade (Business Value)

A solução é viável para o negócio?

Avalia se o produto / funcionalidade é financeiramente sustentável, se possui restrições legais ou éticas e se contribui para os objetivos de negócio da empresa. Isso inclui considerar como uma funcionalidade ou tecnologia gera valor para a empresa e se é **economicamente viável**.

Exemplo: Uma plataforma que depende de uma grande quantidade de dados precisa avaliar se os custos de infraestrutura para suportar esses dados são viáveis ao longo do tempo.

Pergunta-chave: "A funcionalidade / produto maximiza ou incrementa a receita da empresa de forma que o custo para desenvolvê-la tenha um retorno em algum momento no futuro?"





Viabilidade (Business Value)

A solução é viável para o negócio?



Colhendo evidências:

Validar com as diversas áreas de negócio sobre os possíveis riscos que o produto / funcionalidade podem trazer para aquele contexto, incluindo:

Marketing: Como o produto / funcionalidade afeta a marca, reputação, habilitação de vendas, go-to-market, competitividade e diferenciação?

Vendas: Como o produto / funcionalidade afeta a atual máquina de vendas? Ela será capaz de vende-lo?

Customer Success / Suporte: Como o produto / funcionalidade afeta os processos de customer success?

Financeiro: Quais são as restrições de custo e expectativas de receita em torno da funcionalidade do produto? Existe um modelo de P&L?

Jurídico: Quais são os riscos de privacidade, compliance, propriedade intelectual e outros riscos legais que em relação ao produto / funcionalidade?

Desenvolvimento de negócios: Quais parcerias e outros relacionamentos da empresa esse produto / funcionalidade pode afetar e como?

Segurança: Quais são os riscos de segurança cibernética relacionados ao produto / funcionalidade e como vão atacá-los?



Viabilidade (Business Value)

Avaliando o roadmap:

- Os founders estão conscientes dos riscos de viabilidade dos produtos / funcionalidades contidas no roadmap de produto?
- Existe sign off formal das áreas de negócio impactadas pelo produto / funcionalidade? Como esse processo é documentado e acompanhado?
- Há um plano claro para mitigar os riscos identificados em relação à viabilidade do produto / funcionalidade?
- Existe um processo estruturado para avaliar o impacto financeiro e estratégico de cada nova funcionalidade?
- Como o roadmap se alinha com os objetivos de longo prazo e a estratégia geral da empresa?





Executabilidade (Feasibility)

Conseguimos construir isso?

Analisa a viabilidade técnica do que está proposto no roadmap, ou seja, se a empresa tem os recursos e a equipe para executar o que é planejado. Isso envolve avaliar a complexidade técnica, a disponibilidade de talentos e ferramentas necessárias.

Exemplo: Startups que planejam implementar tecnologias complexas como IA precisam de uma equipe capacitada e infraestrutura adequada para realizar o projeto com sucesso.

Pergunta-chave: "A equipe pode implementar estas funcionalidades no tempo e orçamento disponíveis?"





Executabilidade (Feasibility)

Conseguimos construir isso?



Colhendo evidências

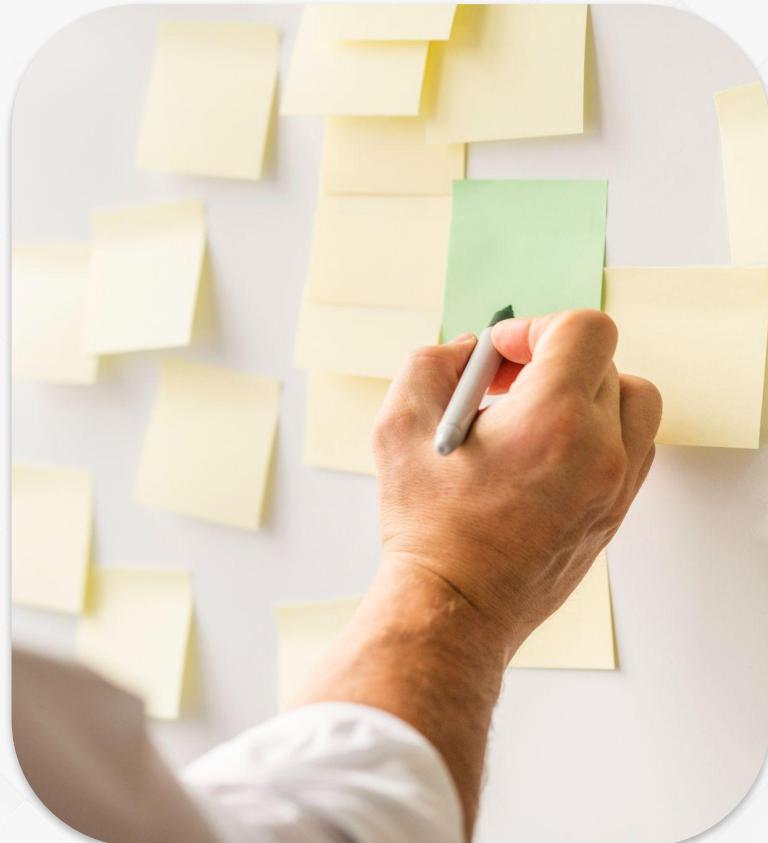
Avaliar a execução dos times no passado são a melhor forma de colher evidências se são capazes de executar o roadmap proposto. Métricas de cycle time, custos de infraestrutura e dependências, e desenhos organizacionais e arquitetura são evidências que podem corroborar para a capacidade de execução do roadmap proposto.



Executabilidade (Feasibility)

Avaliando o roadmap:

- Como o time vai construir isso?
- Quais são as skills técnicas necessárias?
O time têm essas skills?
- Quanto tempo é necessário? Como chegaram nessa estimativa?
- Será necessária alguma mudança arquitetural? Quais?
Essas mudanças estão orçadas?
- Possuem todos os componentes/dependências? São internas ou de terceiros?
- A performance será aceitável?
- Qual será a infraestrutura necessária? Conseguem suportar os custos?
- Escalará para o nível necessário, com os custos viáveis?
Como sabem disso?



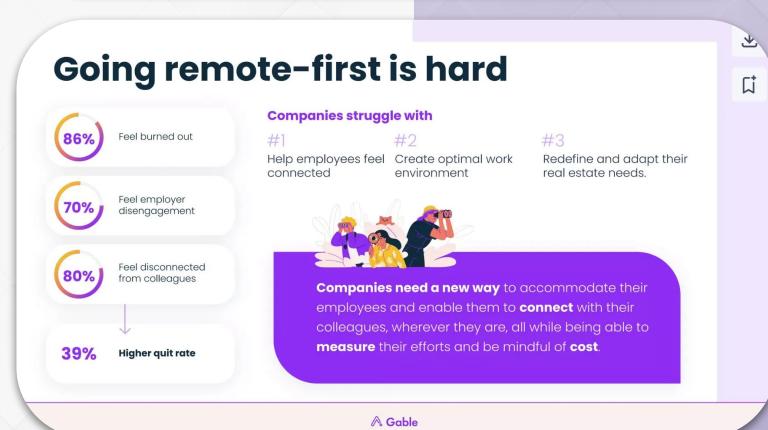
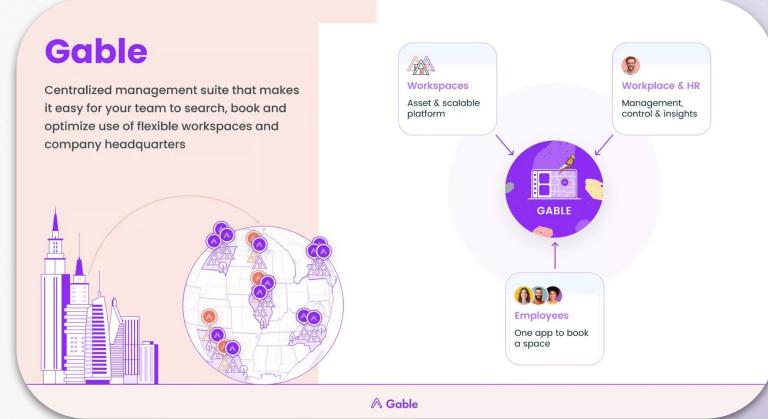
Análise de Roadmap

Análise VUVE - Gable



Para interação

- Usando a estrutura VUVE quais são os riscos (yellow e red flags) que consegue encontrar nos exemplos ao lado?



Análise de Roadmap

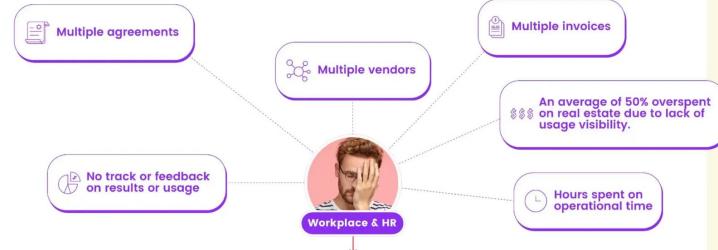
Análise VUVE - Gable



Para interação

- Usando a estrutura VUVE quais são os riscos (yellow e red flags) que consegue encontrar nos exemplos ao lado?

Main Issues



▲ Gable

Product roadmap



▲ Gable

Análise de Roadmap

Análise VUVE - Fibery



Para interação

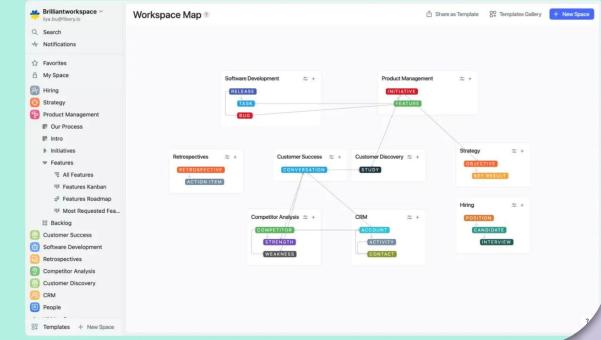
- Usando a estrutura VUVE quais são os riscos (yellow e red flags) que consegue encontrar nos exemplos ao lado?

Product *

We've solved many technical problems often faced in workplace management by creating:

1. Connected spaces

2. Customizable business domains



Problems 😰

Companies use too many **poorly connected** work management tools.

COMPANY LIFE CYCLE WITH CURRENT TOOLS

Trello
Google Docs

Basecamp
Google Docs

asana
Aha!
Dovetail
Google Docs

Jira
Confluence
G Suite
Aha!
pipedrive
zendesk

Office 365
TFS
Aha!
zendesk
G Suite
pipedrive

Too many existing work management tools do not scale with a company, forcing a **painful migration** every couple of years.

Análise de Roadmap

Análise VUVE - Fibery



Para interação

- Usando a estrutura VUVE quais são os riscos (yellow e red flags) que consegue encontrar nos exemplos ao lado?

Solution 😎

Fibery is a connected **all-in-one workspace for product teams & startups**. Companies integrate Fibery at an early stage- Fibery grows with the company as it scales, replacing many scattered tools in the process.

COMPANY LIFE CYCLE WITH FIBERY



Plans 🍋

We are looking for a **~\$4.5M round** and are planning to use it to:

Build a strong marketing team.

Setup and improve channels: partners network, community, content marketing, influencers, and word of mouth.

Launch Fibery 2.0 in August/September 2023 with streamlined use cases for product teams.

Have **+\$5K MRR** and **~\$1M ARR** till Dec 2023. Have \$2-3M ARR in Dec 2024.

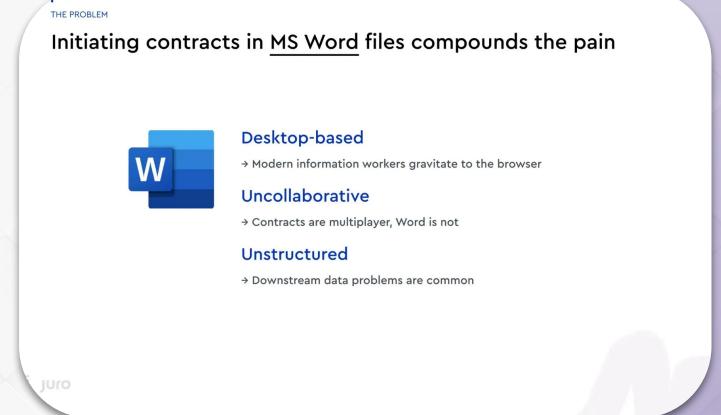
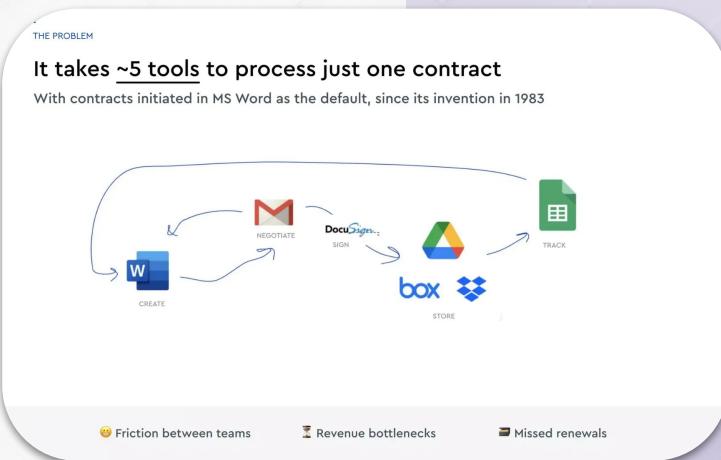
Análise de Roadmap

Análise VUVE - Juro



Para interação

- Usando a estrutura VUVE quais são os riscos (yellow e red flags) que consegue encontrar nos exemplos ao lado?



Análise de Roadmap

Análise VUVE - Juro



Para interação

- Usando a estrutura VUVE quais são os riscos (yellow e red flags) que consegue encontrar nos exemplos ao lado?

SOLUTION TODAY

We're making contracts browser-native
Basing our all-in-one platform around a unique editor that's built on a structured data model

SOLUTIONS

- 1. Self-serve automation**
→ Initiate contracts from integrated tools
- 2. Collaborative workflow**
→ Make updates and approve contracts
- 3. Data-rich repository**
→ Query your contract data in real time

ULTIMATE VISION

And a wider aim to become the default way to agree terms
While DocuSign owns eSignature for files, we will change the paradigm for contract collaboration

2021 2022 The future →

1 SINGLE PLAYER System of automation Automation tools for simple, routine contract workflows.	2 MULTIPLAYER System of record Enable handling of all contracts in Juro across teams.	3
--	--	----------

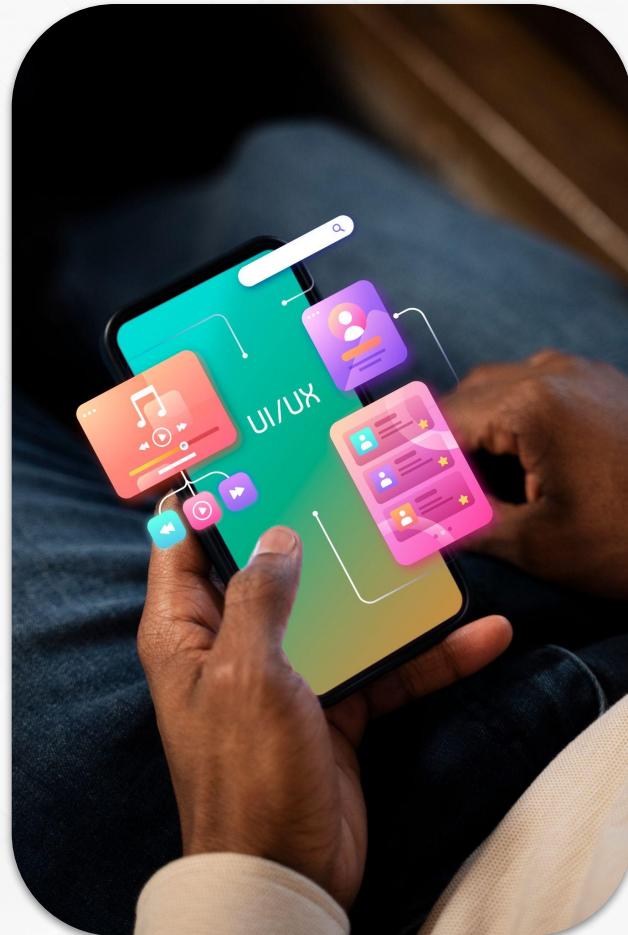
🔍 Focando no que importa: Diferenciando entre funcionalidades must-have e nice-to-have

Distinguir entre **funcionalidades essenciais** e supérfluas é crucial para evitar gastos desnecessários de tempo e dinheiro, garantindo que a tecnologia está focada no product-market fit. Além disso, os negócios de tecnologia mais bem sucedidos no longo são aqueles que conseguem aplicar tecnologia da melhor forma, nas suas **funcionalidades mais imprescindíveis**.



Funcionalidades Must-Have:

- Essenciais, definem o core do produto e atendem a necessidades cruciais do mercado e usuários.
- Frameworks como o modelo Kano e Jobs-to-be-done ajudam a identificar funcionalidades core.
- Tecnologia aplicada de forma engenhosa em Must-have cria vantagem competitiva difícil de copiar.
- Em mercados competitivos, usar tecnologia para elevar a experiência principal é a única forma de motivar usuários a largarem soluções alternativas.





Funcionalidades Nice-to-Have:

- Complementares, agregam valor, mas não são indispensáveis para resolver o problema principal do cliente.
- Frameworks como modelo Kano e MoSCoW ajudam a identificar funcionalidades nice-to-have.
- Nice-to-have complementam a experiência principal, mas se considerados vantagem competitiva, a reação correta é o ceticismo.
- Tecnologias avançadas aplicadas a nice-to-have não tendem a criar empresas vencedoras e sustentáveis a longo prazo.



Para interação

Esses exemplos de produtos / funcionalidades são must-have ou nice-to-have?

O rastreamento em tempo real em apps de delivery.

Must-have ou nice-to-have?



Para interação

Esses exemplos de produtos / funcionalidades são must-have ou nice-to-have?

O rastreamento em tempo real em apps de delivery.

Must-have.

É crucial. Reduz ansiedade do cliente e aprimora a experiência. Geolocalização e atualizações instantâneas geram vantagem competitiva, promovendo transparência e confiabilidade.



Para interação

Esses exemplos de produtos / funcionalidades são must-have ou nice-to-have?

Uma startup de projetos implementa IA para previsão de prazos.

Must-have ou nice-to-have?



Para interação

Esses exemplos de produtos / funcionalidades são must-have ou nice-to-have?

Uma startup de projetos implementa IA para previsão de prazos.

Nice-to-have.

Essa tecnologia avançada, embora atrativa, não atende a necessidade principal, se o que os usuários precisam é se organizar e pode tornar o uso mais complexo.



Para interação

Esses exemplos de produtos / funcionalidades são must-have ou nice-to-have?

Um app de fitness com visualização 3D de exercícios.

Must-have ou nice-to-have?



Para interação

Esses exemplos de produtos / funcionalidades são must-have ou nice-to-have?

Um app de fitness com visualização 3D de exercícios.

Nice-to-have.

Embora atrativo, não é essencial para o objetivo principal de manter a forma e monitorar o progresso. A tecnologia 3D pode ser interessante, mas não é fundamental para o valor central do aplicativo.



Para interação

Esses exemplos de produtos / funcionalidades são must-have ou nice-to-have?

Um sistema de pagamento seguro e confiável para plataformas de e-commerce.

Must-have ou nice-to-have?



Para interação

Esses exemplos de produtos / funcionalidades são must-have ou nice-to-have?

Um sistema de pagamento seguro e confiável para plataformas de e-commerce.

Must-have.

É essencial. Criptografia e processamento em tempo real garantem confiança e viabilidade do negócio, sendo cruciais para operar no mercado.



Para interação

Esses exemplos de produtos / funcionalidades são must-have ou nice-to-have?

Sistemas de recomendação personalizados em plataformas de streaming de música.

Must-have ou nice-to-have?



Para interação

Esses exemplos de produtos / funcionalidades são must-have ou nice-to-have?

Sistemas de recomendação personalizados em plataformas de streaming de música.

Must-have.

São cruciais. Eles mantêm os usuários engajados, sugerindo conteúdo alinhado aos seus gostos, o que aumenta o tempo de uso e a retenção de assinantes.



Para interação

Esses exemplos de produtos / funcionalidades são must-have ou nice-to-have?

Um app de videoconferência com tradução em tempo real via IA.

Must-have ou nice-to-have?



Para interação

Esses exemplos de produtos / funcionalidades são must-have ou nice-to-have?

Um app de videoconferência com tradução em tempo real via IA.

Nice-to-have.

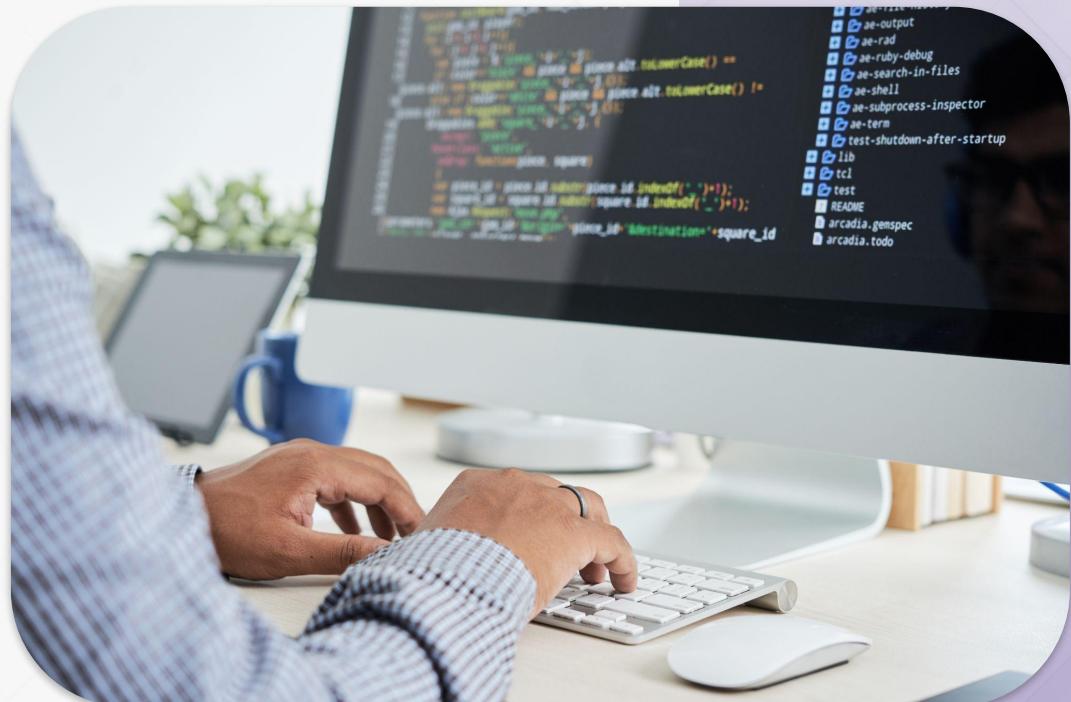
Embora inovador, não é essencial para a maioria dos usuários que buscam apenas chamadas confiáveis. Esta tecnologia avançada, apesar de impressionante, não é crucial para a proposta de valor principal do app.



MVP vs

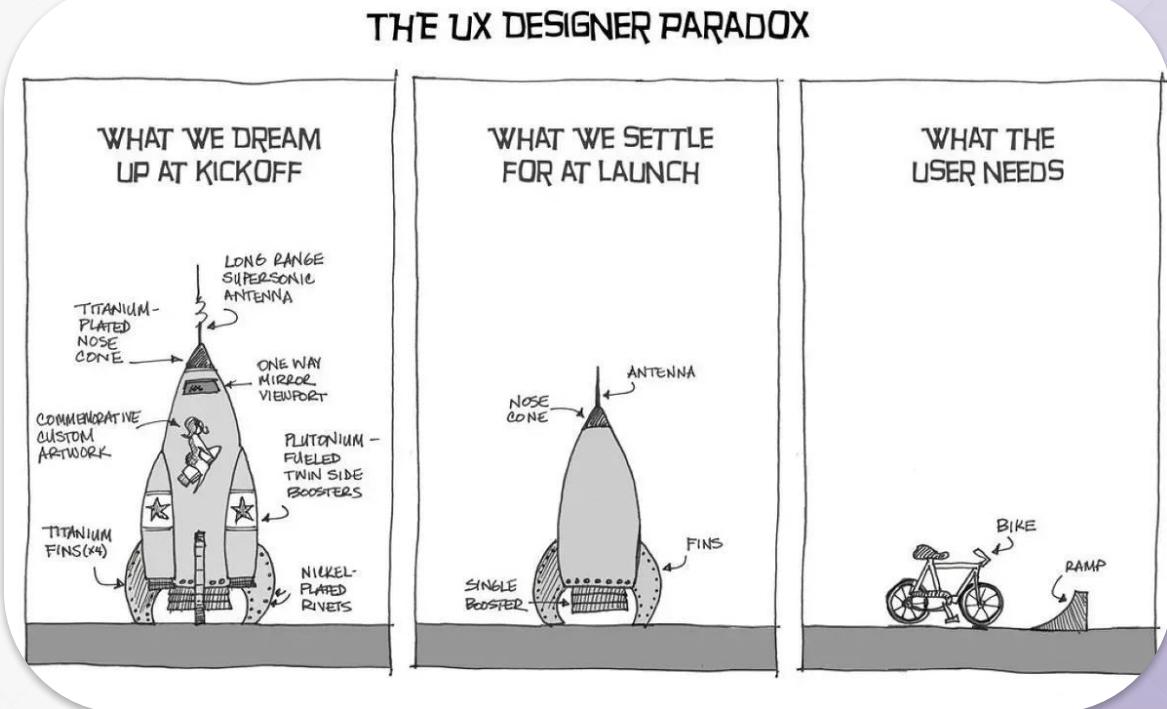
Over-Engineering

Uma abordagem de MVP (Produto Mínimo Viável) ajuda a testar hipóteses rapidamente e ajustar o produto com base no feedback do mercado. Por outro lado, o over-engineering representa o **desenvolvimento de funcionalidades excessivas e complexas**, que podem atrasar o lançamento e consumir recursos desnecessariamente.





MVP (Produto Mínimo Viável)





MVP (Produto Mínimo Viável)

Um MVP permite que startups testem o mercado com o menor conjunto de funcionalidades possível, garantindo que os recursos sejam gastos de forma mais previsível e permitindo pivotar rapidamente se necessário.



Em um MVP, prioriza-se a entrega rápida das funcionalidades essenciais, adiando aspectos como escalabilidade e testes até que o produto ganhe validação.

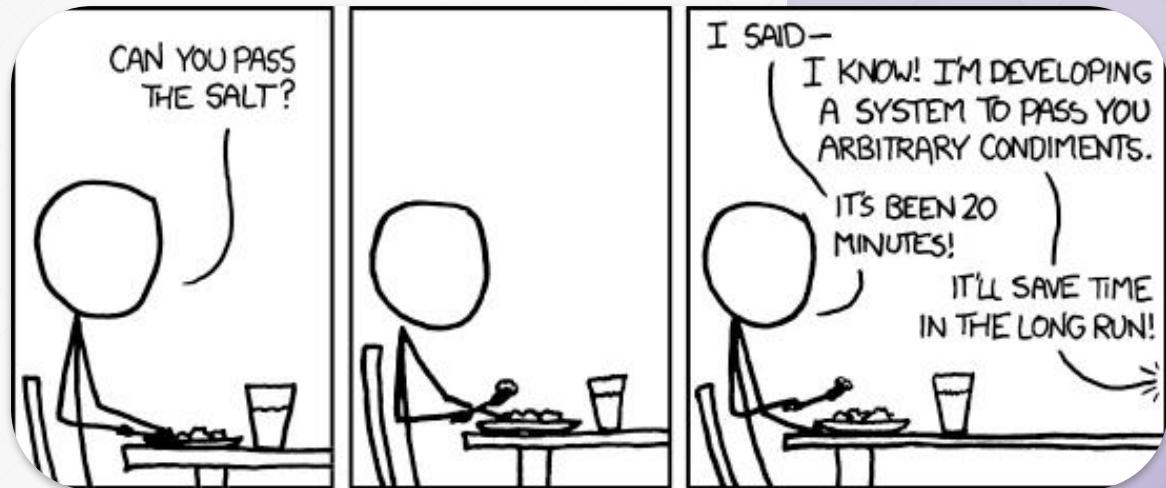


O tech founder deve estar ciente dos trade-offs e gerenciá-los adequadamente, economizando recursos e limitando o comprometimento com uma solução específica.





Over-Engineering





Over-Engineering

Adicionar funcionalidades ou complexidade além do necessário pode resultar em custos altos, atrasos e um produto difícil de escalar, com impactos e custos diretos e indiretos difíceis de medir.



A alta complexidade criada por over-engineering torna difícil acompanhar e controlar o débito técnico, reduzindo a produtividade e eficiência do time conforme o produto escala.



Isso pode tornar a startup incapaz de continuar entregando valor para seus clientes.



⚠️ Sinais de Over-Engineering

Early

O produto possuem mais micro-serviços que funcionários ou usuários.

Early

Mid

Late

O produto está com 100% de cobertura de testes unitários (e time se gaba disso).

Early

A documentação do código é extensa e detalhada, mas o produto ainda não tem usuários reais ou validação de mercado.

Early

Mid

O produto possui um framework próprio cheio de abstrações genéricas.

Early

Mid

Late

O time não usa serviços e bibliotecas para resolver problemas comuns e reescreve tudo do zero.

Early

Mid

Late

O time adota qualquer nova tecnologia ou padrão sem nenhum tipo relevância para o problema em questão.

⚠️ Sinais de Over-Engineering

Early Mid Late

A arquitetura do produto é excessivamente complexa para o estágio atual, com múltiplas camadas de abstração e padrões de design que não oferecem benefícios claros.

Early Mid Late

O produto oferece muitas configurações e opções personalizáveis raramente usadas, complicando a interface e prejudicando a experiência do usuário.

Mid Late

A equipe de desenvolvimento constantemente refatora o código em busca da "perfeição" técnica, mesmo quando não há problemas de desempenho ou escalabilidade evidentes.

Early Mid Late

A equipe implementa tecnologias avançadas (como blockchain ou IA) sem demonstrar como estas agregam valor real ao produto ou resolvem problemas específicos dos usuários.



Para interação

Esses exemplos de produtos / funcionalidades são over-engineering ou MVP?

Uma startup de gerenciamento de tarefas investiu em IA para priorização automática.

Over-engineering ou MVP?



Para interação

Esses exemplos de produtos / funcionalidades são over-engineering ou MVP?

Uma startup de gerenciamento de tarefas investiu em IA para priorização automática.

Over-engineering.

Prematuro. O sistema complexo consumiu meses, mas os usuários iniciais preferiram listas simples, achando a solução elaborada desnecessária.



Para interação

Esses exemplos de produtos / funcionalidades são over-engineering ou MVP?

Uma empresa começou como um site para alugar espaços durante eventos.

Over-engineering ou MVP?



Para interação

Esses exemplos de produtos / funcionalidades são over-engineering ou MVP?

O Airbnb começou como um site para alugar espaços durante eventos.

MVP.

Essa abordagem permitiu validar rapidamente o conceito de hospedagem compartilhada, adicionando novas funcionalidades conforme o interesse crescia.



Para interação

Esses exemplos de produtos / funcionalidades são over-engineering ou MVP?

Uma startup de e-commerce investiu em um sistema de recomendação avançado com 300 MAU.

Over-engineering ou MVP?



Para interação

Esses exemplos de produtos / funcionalidades são over-engineering ou MVP?

Uma startup de e-commerce investiu em um sistema de recomendação avançado com 300 MAU.

Over-engineering.

O projeto consumiu recursos excessivos, mas os usuários iniciais preferiam navegação simples. A complexidade dificultou adaptações baseadas no feedback inicial.



Para interação

Esses exemplos de produtos / funcionalidades são over-engineering ou MVP?

Uma startup começou oferecendo sincronização básica de arquivos e armazenamento em nuvem.

Over-engineering ou MVP?



Para interação

Esses exemplos de produtos / funcionalidades são over-engineering ou MVP?

O Dropbox começou oferecendo sincronização básica de arquivos e armazenamento em nuvem.

MVP.

Esta abordagem minimalista permitiu validar rapidamente o conceito e a demanda do mercado, antes de investir em recursos mais complexos.



Para interação

Esses exemplos de produtos / funcionalidades são over-engineering ou MVP?

Uma startup de rede social investiu em realidade virtual imersiva.

Over-engineering ou MVP?



Para interação

Esses exemplos de produtos / funcionalidades são over-engineering ou MVP?

Uma startup de rede social investiu em realidade virtual imersiva.

Over-engineering.

Prematuro. O projeto custoso não atendeu às necessidades dos primeiros usuários, que preferiram interfaces convencionais para interações online.



Para interação

Esses exemplos de produtos / funcionalidades são over-engineering ou MVP?

Uma startup começou com um aplicativo compartilhamento de fotos, com filtros básicos.

Over-engineering ou MVP?



Para interação

Esses exemplos de produtos / funcionalidades são over-engineering ou MVP?

O Instagram começou com compartilhamento de fotos e filtros básicos.

MVP.

Essa estratégia permitiu validar rapidamente o conceito e a demanda do mercado, criando uma base de usuários engajada antes de adicionar recursos mais complexos.

?

Perguntas-Chaves para Avaliar Product-Market Fit (Do Ponto de Vista de Tecnologia)



? Perguntas-Chaves para Avaliar Product-Market Fit (Do Ponto de Vista de Tecnologia)

A tecnologia oferece vantagem competitiva duradoura ou é facilmente replicável? Por que ela não poderia ser replicada facilmente?

Existem funcionalidades tecnológicas que diferenciam significativamente o produto dos concorrentes?

Como esta tecnologia se alinha com o problema e as necessidades atuais dos usuários?

Existem alternativas mais simples que poderiam atender às necessidades imediatas do produto e do mercado?

Quais são os trade-offs entre implementar esta funcionalidade agora versus adiar para uma fase posterior do desenvolvimento?

Como a tecnologia adotada se alinha às tendências e expectativas futuras do mercado?

Como a arquitetura tecnológica da empresa suporta a escalabilidade e o crescimento futuro?

Como a infraestrutura da empresa garante a segurança e privacidade dos dados dos usuários, conforme as regulamentações?

Quão flexível é a arquitetura tecnológica para permitir pivots rápidos ou expansões para novos segmentos de mercado, se necessário?



Exemplos de empresas onde a tecnologia foi um diferencial competitivo chave



Caso de Estudo



Desenvolveu uma tecnologia de sincronização de arquivos robusta e eficiente, permitindo que os usuários acessassem seus arquivos em qualquer dispositivo, em tempo real. Essa inovação simplificou o compartilhamento e colaboração em documentos, tornando-se um diferencial significativo no mercado de armazenamento em nuvem.

Technical advantages

- Client & server written in Python: 5-10x faster development time
- Leverage Amazon S3, EC2 to achieve scale & reliability
- Obsession with performance
- Seamless compression, encryption, binary diffing, failure recovery

www.getdropbox.com

Platform

- Onramp for all UGC from desktop → web
- Convenient for users: drag and drop instead of sign-in and browser upload
- Beneficial for web apps: more content uploaded, less friction & fewer failures
- Bigger picture: all files in Dropbox, delegate access to web apps

Dropbox: \$15K VC investment turned into \$16.8B. Dropbox's initial pitch deck



Caso de Estudo



Desenvolveu uma plataforma de gerenciamento de mídias sociais que simplificou o agendamento e análise de postagens. Sua tecnologia permitiu que pequenas empresas e indivíduos gerenciar eficientemente sua presença nas redes sociais, criando um diferencial significativo no mercado de marketing digital.

Milestones

- Launched web app
January 2011
- 55,000 users (\$150K revenue)
October 2011
- Launch the API
October 2011
- Integrated in 50 apps
December 2011
- 100,000 users (\$288K revenue)
January 2012
- 1 million users (\$3.6M revenue)
January 2013



Buffer: The slide deck we used to raise half a million dollars



Caso de Estudo

BuzzFeed

Desenvolveu uma plataforma de criação e distribuição de conteúdo viral que utiliza análise de dados avançada para prever e otimizar o engajamento do público. Sua tecnologia de inteligência artificial e aprendizado de máquina permite criar conteúdo altamente compartilhável e personalizado, diferenciando-se no mercado de mídia digital.

The image shows two main sections of a BuzzFeed pitch deck:

- New Front Page:** This section displays the BuzzFeed homepage with a dark theme. It features three main categories: "EDITORIAL BUZZ", "ALGORITHMICALLY DETECTED BUZZ", and "USER GENERATED BUZZ". Below these are sections for "Raw Buzz", "Raw Links", and "Hours".
- Optimization:** This section shows a detailed dashboard titled "Landing Page Performance Metrics". It includes various charts and graphs related to search engine optimization (SEO) and referrer optimization. Labels point to specific metrics: "SEARCH ENGINE & REFERER OPTIMIZATION", "LANDING PAGE PERFORMANCE METRICS", "MEASURE VIRAL EFFECTS", and "WIDGET TRAFFIC WITH STATS".

Buzzfeed: Pitch Deck



Caso de Estudo



INTERCOM

Desenvolveu uma plataforma de comunicação com clientes que utiliza machine learning para personalizar e otimizar interações. Sua tecnologia permite que empresas se comuniquem com seus clientes de forma mais eficaz e contextual, diferenciando-se no mercado de suporte ao cliente e engajamento.

The problem

- * Building meaningful relationships with customers is hard / impossible for SaaS providers (meaningful relationships → loyal customers → profitable customers and organic growth)
- * Building meaningful relationships requires: 1. customer discovery, 2. customer research, 3. customer communication, 4. relationship management
- * The current tools available are complex, ineffective, not built for SaaS... there's no one tool to do customer relationship management and messaging for SaaS providers.
- * (Even e-mail fails: generating targeted e-mail lists takes a lot of work, e-mail campaigns have terrible engagement because e-mail is out of context)

The solution

- * Intercom is...
 - simple install like Google Analytics,
 - customer base browsing (e.g. signed-up today, on premium plan),
 - individual customer research (e.g. see Twitter followers, company),
 - advanced in-app messaging (user sees message on next login or at specific event and can reply to messages),
 - customer relationship management (Intercom calculates relationship metric over time and allows app owner find people who need attention).

Intercom: First pitch deck



Caso de Estudo



Desenvolveu uma plataforma de rede profissional que utiliza algoritmos avançados para conectar pessoas e oportunidades de trabalho. Sua tecnologia de análise de dados e machine learning permite recomendações personalizadas de empregos, conexões e conteúdo, criando um ecossistema único no mercado de networking profissional e recrutamento.

The Network Enables Revenue

LinkedIn Your network is bigger than you think

Value of the network	Revenue model (in detail)
LinkedIn • Network-based reputation system creates trust: key to high transaction volume and utility	1. InLeads: contextual search text ads 2. Opportunities: network-filtered job listings 3. Network Plus: network-extending subscription Reputation system is not directly monetized (Universal service stays free for all users)

Building the network is the challenge, and LinkedIn has the foundation

"Executives should try LinkedIn. At best, it could be incredibly useful. At worst, the site guards your privacy and doesn't waste your time."

"LinkedIn could change the way executive recruiting is conducted."

August 2004 CONFIDENTIAL

January 2004 PC



LinkedIn's Market Opportunity Is Large

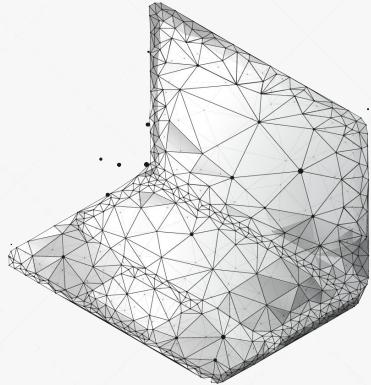
LinkedIn Your network is bigger than you think

Comparables	Comparable market sizes (2004)	
1. InLeads (Targeted ads)	Google AdWords for people Lexis-Nexis for services Yellow pages/directories	Search ads: \$2.8B Lexis-Nexis: \$2.3B Offline yellow pages, directories, and local ads \$10B+
2. Opportunities (Listings)	Monster et al Classifieds	Job sites: \$0.8B (excluding classifieds)
3. Network Plus (Subscriptions)	Match.com et al	Personals sites: \$0.6B

Sources: SEC Filing, SearchEngineWatch.com, Kelsey Group, Mediawatch, Goldman Sachs, Marketdata Enterprises, CIBS Marketwatch
 August 2004 CONFIDENTIAL

[LinkedIn: Series B Pitch Deck](#)





 SBC School



Resultados

Após esta seção, os analistas terão uma visão crítica sobre como a tecnologia apoia o product-market fit, identificando alinhamento com o mercado, sinais de over-engineering e o que é essencial no roadmap. Isso permitirá perguntas estratégicas sobre a execução tecnológica, ajudando na avaliação do potencial de sucesso a longo prazo.

Bibliografia

- <https://betterprogramming.pub/overengineering-why-we-do-it-and-10-ways-to-tackle-it-460663d35ff31>
- <https://medium.com/@rdsuhbas/10-modern-software-engineering-mistakes-bc67fbef4fc8>
- <https://www.mindtheproduct.com/overengineering-can-kill-your-product/>
- <https://github.com/EnterpriseQualityCoding/FizzBuzzEnterpriseEdition>
- <https://gist.github.com/lolzballs/2152bc0f31ee0286b722>
- <https://www.vestbee.com/blog/articles/startup-fundraising-how-to-create-a-perfect-roadmap-slide-in-your-pitch-deck>
- <https://visme.co/blog/pt-br/pitch-decks/>
- <https://evolvemvp.com/exemplos-de-pitch-que-ajudaram-as-startups-a-captar-milhoes/>
- <https://userpilot.com/blog/fake-door-testing/>