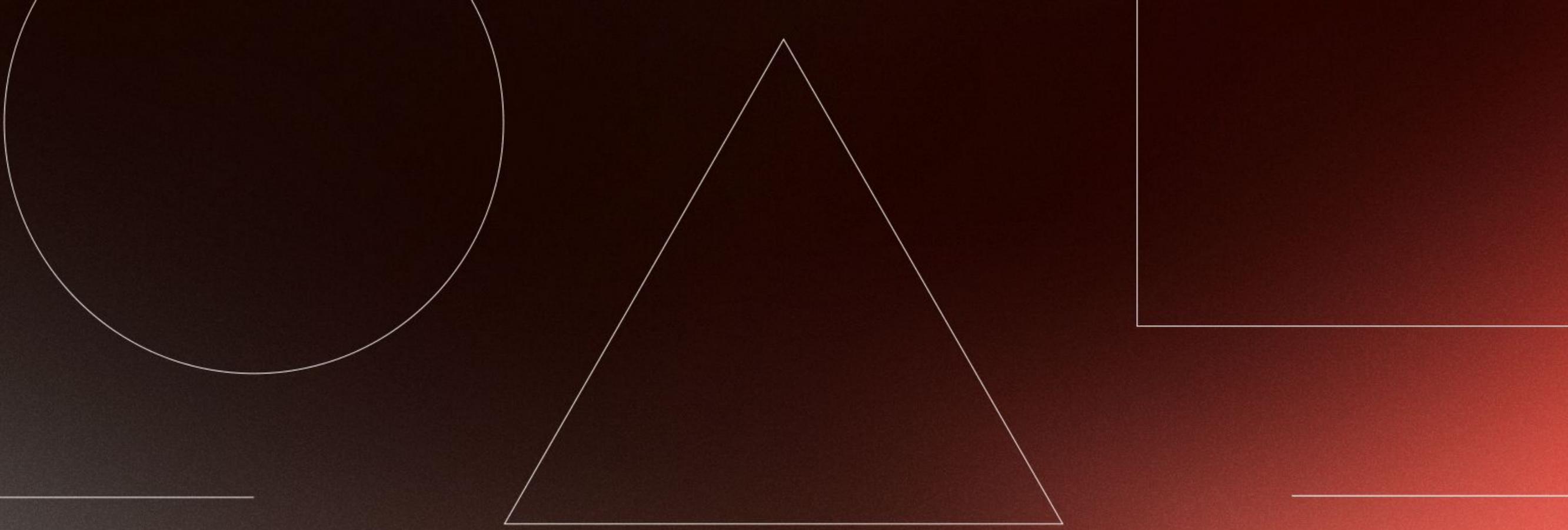


**ioasys**

# **E-book Design System**

An abstract graphic composed of thin purple lines on a black background. It features a large circle in the upper right, a triangle below it, and a rectangle at the bottom right. Several horizontal lines intersect these shapes, creating a layered, architectural feel.



**Olá!**

Neste ebook, vamos te apresentar o jeito ioasys de ver e construir um design system. Sabemos que muito mais do que componentes, regras e fluxos, um design system institui uma nova maneira de trabalhar. Por isso, é fundamental entender como ele funciona para aproveitar ao máximo os seus benefícios. Os conteúdos deste ebook estão divididos em 8 partes, seguindo as etapas de construção e implementação de um design system na prática. Vamos lá?

## SUMÁRIO

#01

**O que é um Design System?**

#02

**Design Systems são sobre pessoas,  
não só sobre pixels.**

#03

**Design Principles: a verdade  
ancorada do Design System.**

#04

**Design Tokens: espelhando design e tecnologia.**

#05

**Componentes: projetando do  
menor para o maior.**

#06

**Se não está documentado, não existe.**

#07

**Design Ops: governança e cultura.**

#08

**Versionamento e atualização.**

# Estamos cada vez mais **digitais** e, mais importante ainda, cada vez mais **omnichannel**.

Isso implica em vários produtos em diferentes plataformas, com linguagens e layouts diferentes. Trabalhar tela por tela, desenvolvendo o mesmo componente várias vezes tem se tornado impraticável, seja pelo número de inconsistências ou pela perda de tempo. Para acelerar e otimizar a forma com que produzimos interfaces muitas empresas têm resgatado a ideia de design systems. Não é um conceito novo, mas sua aplicação no digital traz diversas vantagens para a otimização do dia a dia de trabalho, que iremos abordar a seguir.



## Interface elements

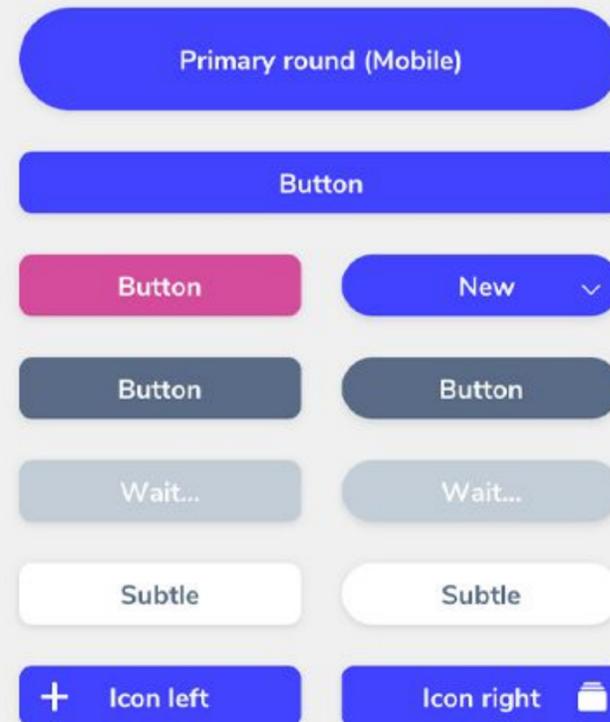
Write your comment here...

Pending

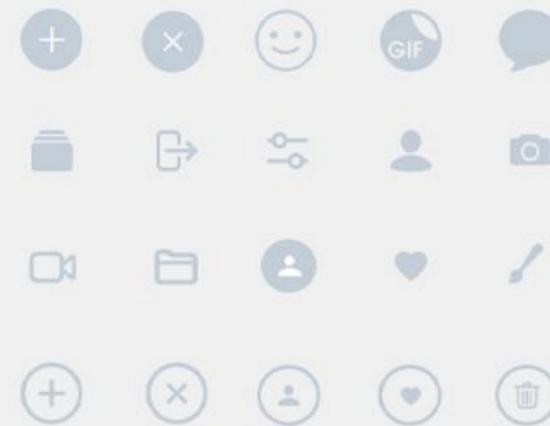
Approved

Write your comment here...

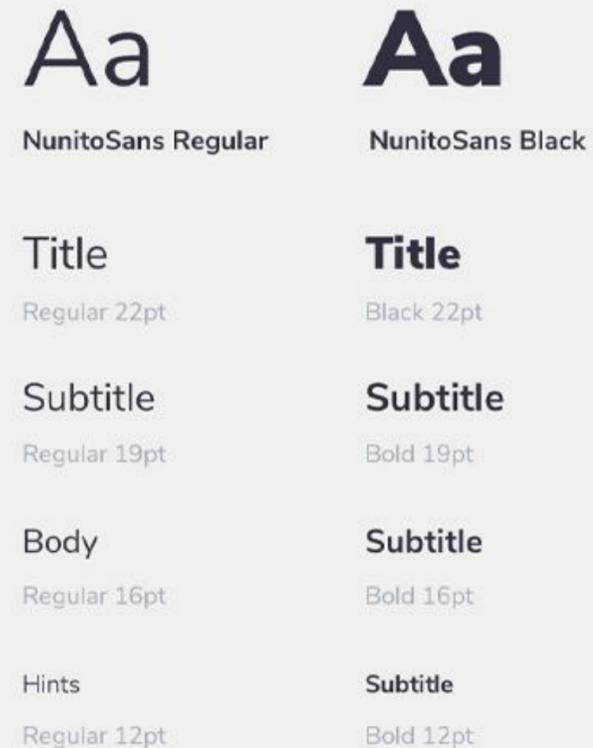
## Buttons



## Icons



## Type



# Afinal, o que é um design system?

Como o próprio nome diz, é um sistema de design. Surgiu há séculos, inicialmente aplicado no campo do design gráfico, para otimizar a maneira com que as pessoas trabalhavam e promover consistência nas interfaces, produtos e plataformas. Assim, eram criados padrões para cores, headlines, imagens, tamanhos e ilustrações que promoviam a identidade e o posicionamento de uma marca. Mas o conceito se popularizou no século passado, quando começou a ser empregado pela Nasa em seus projetos. Agora, presenciamos uma nova onda de popularização, desta vez, aplicada ao cenário digital.



Design systems propõem um jeito diferente de **trabalhar, projetar e desenvolver interfaces.**

Um jeito mais eficiente e assertivo, que promove ao mesmo tempo um ganho em consistência e flexibilidade no dia a dia de desenvolvimento de produtos digitais. O design system funciona como um ecossistema de bibliotecas de componentes reutilizáveis espelhados entre tecnologia e design, que podem ser agrupados para criar aplicações escaláveis. Esses componentes são guiados por princípios de uso e gestão de estilo centralizada, promovendo a qualidade nos produtos digitais por meio de padrões claros.



Dessa maneira, ele propõe uma abordagem para resolvermos problemas pensando do menor para o maior — a partir de variáveis de estilo. Quando bem empregado, aumenta a qualidade das interfaces e, conseqüentemente, melhora a interação humano-computador. Logo, a curva de aprendizado do usuário se torna mais rápida e o desenvolvimento de novos produtos fica mais eficiente. A padronização faz com que não seja necessário tomar decisões que já foram tomadas, como cores, espessuras de borda e tipografia.

Portanto, o **design system** funciona como um **repositório de microdecisões.**

Na prática ele funciona como um guia para designers, desenvolvedores, líderes digitais, product owners, scrum masters e demais stakeholders, que usam informações disponibilizadas para criar os produtos digitais de uma empresa.

**Um design system nunca está pronto**, está em constante evolução. Mas, como sabemos, um bom produto vem sempre acompanhado de boas métricas. Para o seu sucesso, é necessário manter uma comunicação frequente com seus usuários, criando e acompanhando métricas de adoção, que em conjunto com uma abordagem mais qualitativa fornecerá dados confiáveis para guiar decisões assertivas.



# Vantagens de um Design System

Bom, como você pode imaginar, o uso de um design system possui diversas vantagens. A principal delas é o **ganho na rapidez** e na **consistência**. Quando você for construir um novo produto poderá usar um repositório de componentes já pronto, testado e de alta qualidade. Assim, ele agiliza o seu trabalho. Você não precisa perder tempo pensando em espaçamentos, cores e tipografias, estes elementos já foram determinados. Dessa maneira, o processo de desenvolvimento de produtos digitais se torna mais rápido e mais barato. O time to market da solução se torna menor e os processos passam a ser ágeis de verdade.





O design system  
**automatiza a parte**  
**burocrática** para que  
você se concentre na  
**parte estratégica.**

Otimizando a intuição das interfaces, os fluxos de navegação e a visualização de dados. Além de criar produtos completos e experiências incríveis. Outra vantagem é que ao padronizar as interfaces aceleramos, também, a curva de aprendizado dos usuários finais dos sistemas. Por fim, além de garantir a consistência entre produtos, ainda potencializamos a identidade e a marca da empresa.

Entretanto, para um design system obter sucesso real, ele precisa ser **adotado pelas pessoas**. Continue lendo, que vamos apresentar algumas estratégias para você construir design systems que sejam realmente parte da cultura da sua empresa.

#02

# **Design Systems são sobre pessoas, não só sobre pixels**

Um design system só entrega valor quando aplicado a um problema real. Por isso, antes de entender como podemos usá-lo, é importante identificar quais problemas estamos tentando solucionar. É preciso mergulhar no contexto da empresa e entender quais são as suas principais necessidades e dores. Não existe fórmula mágica para essa imersão, mas nesse processo é imprescindível conversar com stakeholders de vários perfis diferentes.

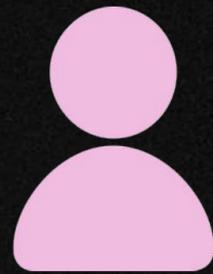


É sempre importante lembrar que por trás dos pixels estão **pessoas.**

São elas que vão aplicar o design system diariamente e se beneficiar dos componentes e guidelines desenvolvidos. O sucesso do seu design system depende da adoção contínua, por isso é fundamental que ele seja desenvolvido por meio de uma estratégia centrada no usuário desde o primeiro dia.

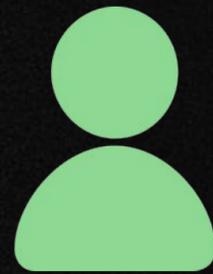
Uma situação comum com a qual nos deparamos é que muitas pessoas não sabem o que é um design system e não entendem como ele pode impactar positivamente o seu dia a dia de trabalho. Então, é essencial fazer o gerenciamento de entregas de valor. Você pode começar mapeando quem vão ser as pessoas que vão usar o seu design system, direta ou indiretamente, e enumerar como elas poderão se beneficiar dele. Dessa maneira, fica mais fácil traçar estratégias de engajamento para cada perfil.

# Design systems não se limitam a designers



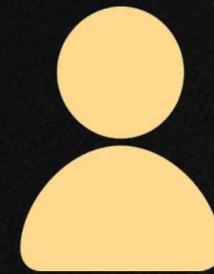
## Broader public.

Não usam os produtos digitais ou tem contato com o Design System, mas são impactados pelo uso da marca.



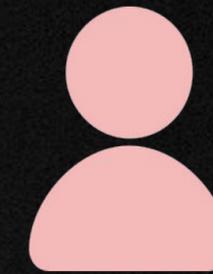
## End Users

Usam os produtos construídos através do Design System, mas não necessariamente o conhecem.



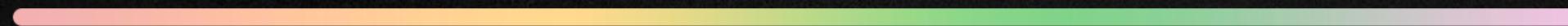
## Stakeholders

Não lidam com o Design System diretamente, mas estão envolvidos no dia a dia de construção de produtos digitais.



## Makers

São os que constroem os produtos digitais na empresa, lidando com o Design System diariamente.



A diversidade é fundamental neste ponto: não se limite a designers e pessoas desenvolvedoras.

Para o design system ser um sucesso você não precisa de uma equipe, você precisa da **empresa inteira.**

Uma boa estratégia é usar uma triangulação de métodos para envolver diversos stakeholders, mesmo que pontualmente. Mesclar técnicas quantitativas e qualitativas pode aumentar o nível de engajamento e reduzir a margem de erro dos resultados encontrados. Uma combinação de métodos possível é fazer surveys, entrevistas em profundidade e workshops, cada um direcionado a um perfil específico.





É importante ressaltar que a conversa com as pessoas deve ser respaldada por um inventário profundo dos produtos digitais. Estas duas tarefas devem ocorrer paralelamente. Assim, você vai poder diagnosticar como os produtos são construídos, quais são os componentes mais usados, quais são os padrões recorrentes e as principais inconsistências — e durante as entrevistas e workshops você vai conseguir entender porque eles acontecem.

Os pixels são importantes,  
**mas não são suficientes  
sozinhos.**



Vale lembrar que o design system é um **organismo vivo**. Para se manter relevante no contexto da empresa ele deve evoluir e se adequar às necessidades emergentes. Você deve envolver as pessoas desde o início da criação, mas deve continuar ouvindo e engajando seus usuários na medida em que ele cresce e se torna mais robusto. Por isso é tão importante enxergar um design system como um produto, não como um projeto. Ele tem um início, mas não vai ter um fim.

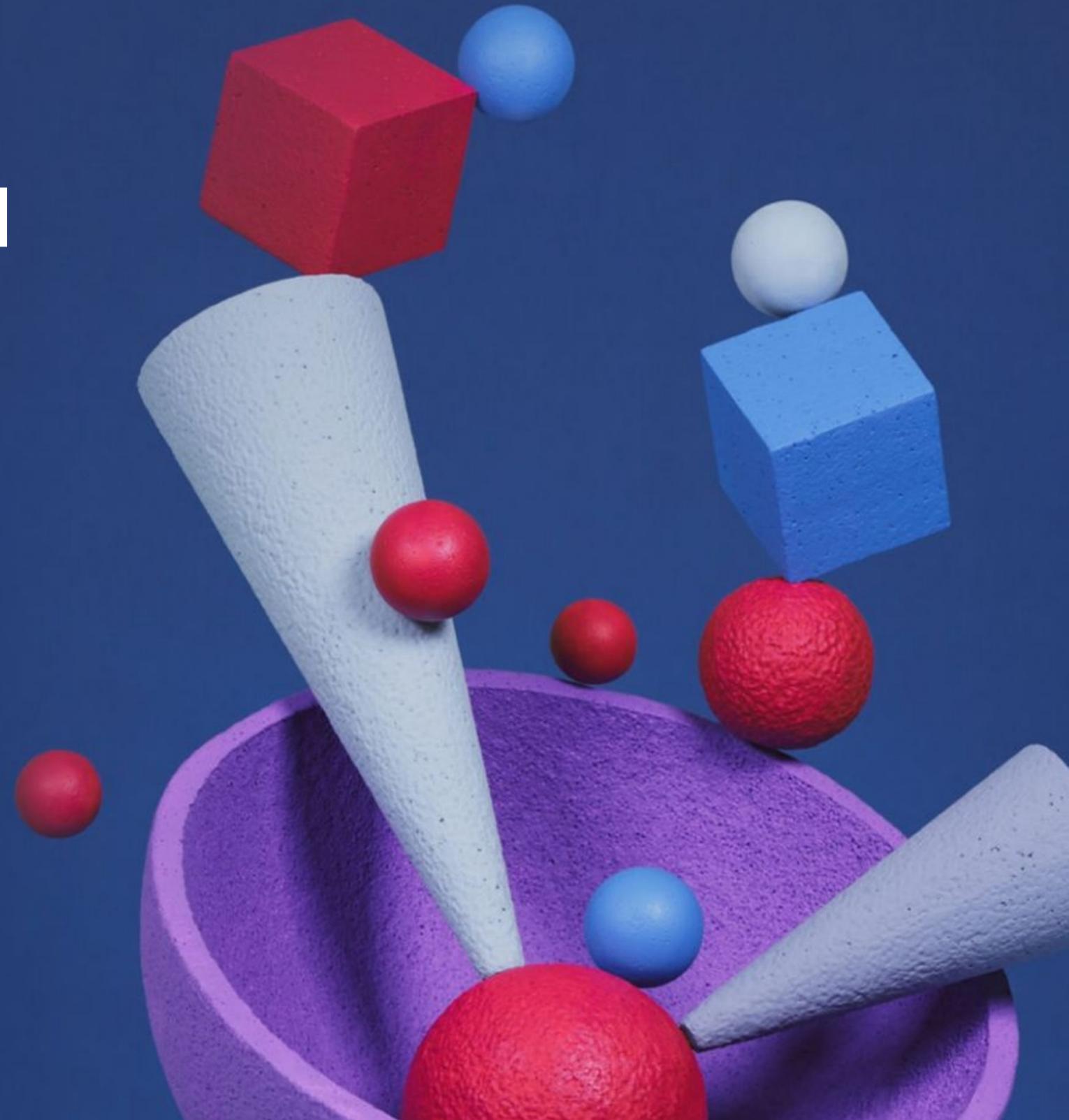
**Não se preocupe: vamos abordar mais a fundo esse assunto no final do nosso ebook.**

Quando você tiver o **contexto** e a **cultura da empresa** bem mapeados, será possível estabelecer os design principles do seu design system a partir de uma base sólida. Vamos explicar o que eles são e porque são tão importantes.

#03

# Design Principles: a verdade ancorada do Design System

Se você chegou até aqui, deve ter lido que “design systems são organismos vivos” e que “eles devem evoluir para se manter relevantes na cultura da empresa” várias vezes. Sim, a capacidade de adaptação e mudança é imprescindível: **o seu design system deve ser flexível**. Mas vamos explicar melhor sobre o que não deve mudar. Você já consegue adivinhar o que é?



“I very frequently get the question:  
what’s going to the change in the  
next 10 years’

I almost never get the question:  
‘what’s not going to the change  
in the next 10 years?’

I submit to you that second  
question is actually the more  
important of the two.”

- Jeff Bezos, CEO da Amazon.





Essa fala do Jeff Bezos é super interessante! As pessoas sempre perguntam para ele o que vai mudar em 10 anos, mas o que ele está interessado mesmo é em saber o que não vai mudar em 10 anos. Bom, pelo menos quando se trata de um design system, nós já sabemos a resposta: **os design principles.**

Os design principles são um conjunto de regras acordadas que vão **guiar o time na tomada de decisão durante a construção do design system** e de todos os produtos digitais que vão nascer a partir dele. Eles têm o poder de unificar os times em torno do que é mais importante para os produtos, evitando desvios que comprometem o resultado desejado.



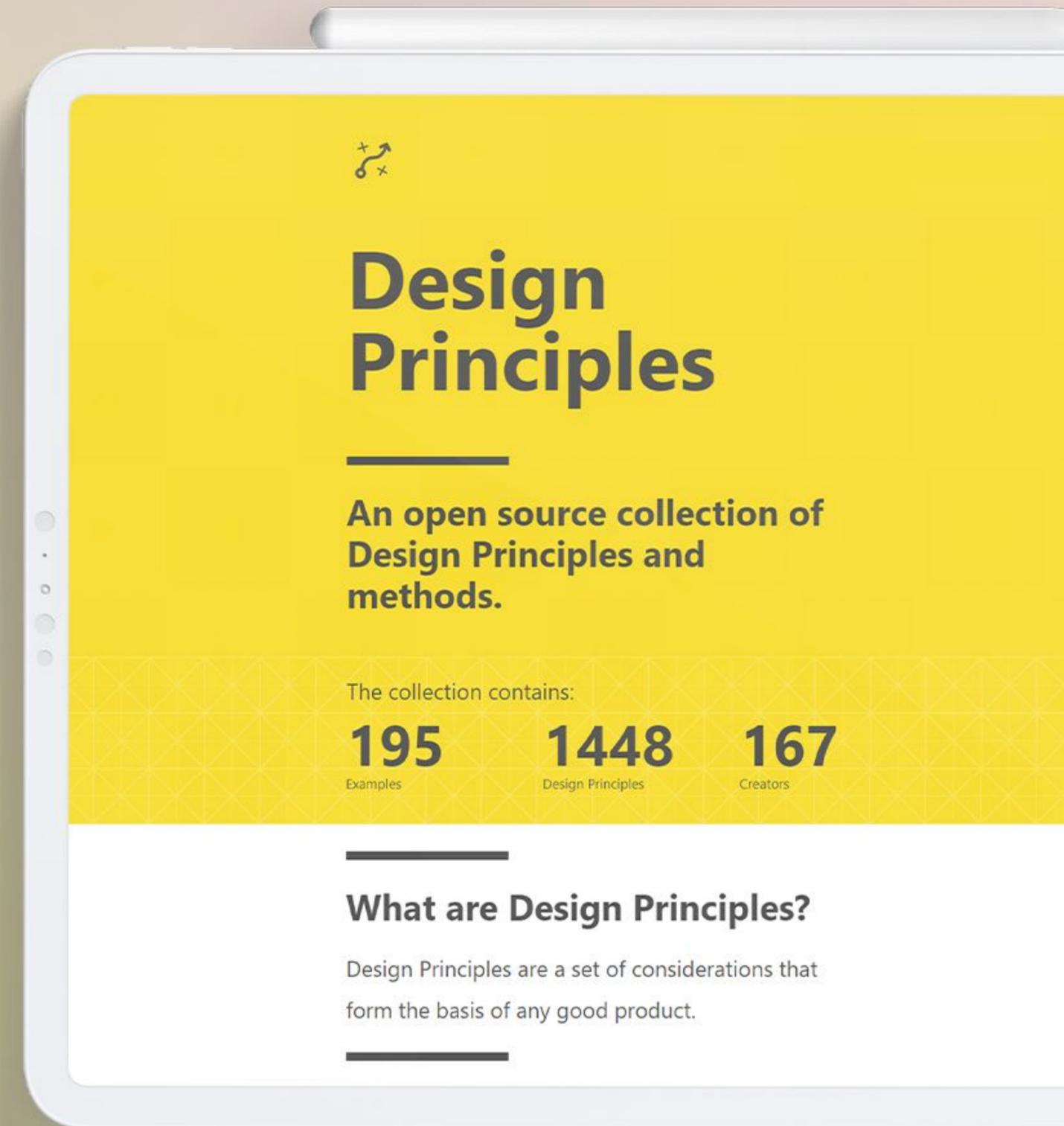
São grandes facilitadores para o processo de desenvolvimento de produtos digitais, mas para serem efetivos eles devem ser específicos, tangíveis e refletir a cultura da empresa. Só assim serão verdadeiramente úteis para o time.

Por isso, é fundamental estabelecer os design principles através de processos de cocriação.

# Quem usa **design principles**

Por meio de pesquisas e benchmarkings, fica claro que os principais design systems do mundo possuem um denominador comum: todos têm design principles sólidos e os apresentam de maneira clara. Aqui vamos apresentar três cases para você, mas vamos indicar também um site incrível que possui um repositório enorme de design principles para você se inspirar:

[principles.design](https://principles.design)



## DICAS ÚTEIS

# #1 Salesforce/Lightning

## Clarity

Elimine a ambiguidade. Permita que as pessoas vejam, entendam e ajam com confiança.

## Efficiency

Simplifique e otimize os fluxos de trabalho. Preveja de forma inteligente as necessidades para ajudar as pessoas

## Consistency

Crie familiaridade e fortaleça a intuição aplicando a mesma solução ao mesmo problema.

## Beauty

Demonstre respeito pelo tempo e atenção das pessoas por meio de um projeto atencioso e elegante.

# #2 Vtex.

## Plug-and-play

Trabalhar mais rapidamente por meio de desafios básicos, para que possamos enfrentar os de ordem superior.

## Empowering

Todas as equipes têm o poder de oferecer uma experiência no nível da linha de base, mesmo se você não tiver time experiente

## Consistent

O Design System deve ser a principal ferramenta para promover o alinhamento e melhorar a consistência em tudo.

## Reusable

Uma cultura de reutilização reduz a redundância de código e aumenta a qualidade geral com soluções bem testadas.

## #3 Orbit/Kiwi

### Taking away visual decisions.

As pessoas não precisam pensar em qual cor escolher, o que pode ser combinado e o que não pode. Todos os componentes precisam ter nossas decisões de projeto integradas e fornecer apenas uma lista de variações.

### Promote consistency

Há muito o que pensar ao trabalhar com um kit de interface do usuário - remova as partes móveis o máximo possível. Nomeação e escopo consistentes para nossos componentes devem ajudar a todos.

### Use appropriate defaults

Alternar entre variações ou visualizar valores padrão leva alguns segundos para cada uso de um componente. Devemos usar padrões apropriados para que o conteúdo ajude as pessoas no dia a dia de trabalho.

### Usability over maintainability

O uso do kit Orbiu UI deve ser fácil e intuitivo para que qualquer designer possa começar a usá-lo sem treinamento extensivo. Todos os casos extremos ou exceções devem ser documentados, idealmente com exemplos.

---

Para a **Orbit**, o principal objetivo do design system é simplificar. As microdecisões devem ser reduzidas ao máximo e os produtos devem prezar por uma boa usabilidade e experiência, trazendo exemplos e padrões claros. Vale ressaltar que a consistência está presente nos **design principles das três empresas**, mas cada uma aborda esse conceito de uma maneira. Por isso, é importante contextualizar o que cada princípio significa.

# Design principles para construir Design principles

Não existe uma regra universal ou número mágico de design principles, **o importante é que eles consigam transmitir os valores**, cultura e o que a empresa entende como uma boa experiência. A partir do estudo de design principles de mais de 20 design systems de empresas de diferentes segmentos ao redor do mundo, foi possível identificar algumas tendências e boas práticas que podem auxiliar no processo de definição de bons princípios para o seu design system.



## DICAS ÚTEIS

#1

### **Menos, mas melhor**

O foco não deve estar na quantidade, mas sim na qualidade. Normalmente os design systems possuem entre três e seis design principles.

#3

### **Princípio chiclete**

Para garantir a efetividade, os princípios devem ser memoráveis e devem estar presente no dia a dia das pessoas. Você pode inclusive elaborar slogans ou frase de efeito.

#2

### **Sejamos objetivos**

Não existe limite de caracteres, mas é importante que os princípios sejam fáceis de compreender por todas as pessoas envolvidas no design system.

#4

### **Só vale se der para usar.**

Se não puder ser aplicado em um produto ele perde a utilidade. Evite direcionamentos subjetivos, como “fácil de usar” ou muito específicos, como “use grids de 8px”.



Depois que você estabelecer os **design principles** que vão guiar o seu design system, já pode começar a colocar a mão na massa e produzir os componentes!

Vamos te contar qual é o principal segredo para construir bibliotecas de componentes escaláveis.

A man with glasses and a dark shirt is sitting at a desk, looking at a laptop. A woman with her hair in a ponytail, wearing a light-colored patterned shirt, is sitting next to him, looking at the laptop and holding a pen to her chin. They are in a modern office environment with blurred background elements like plants and office furniture.

#04

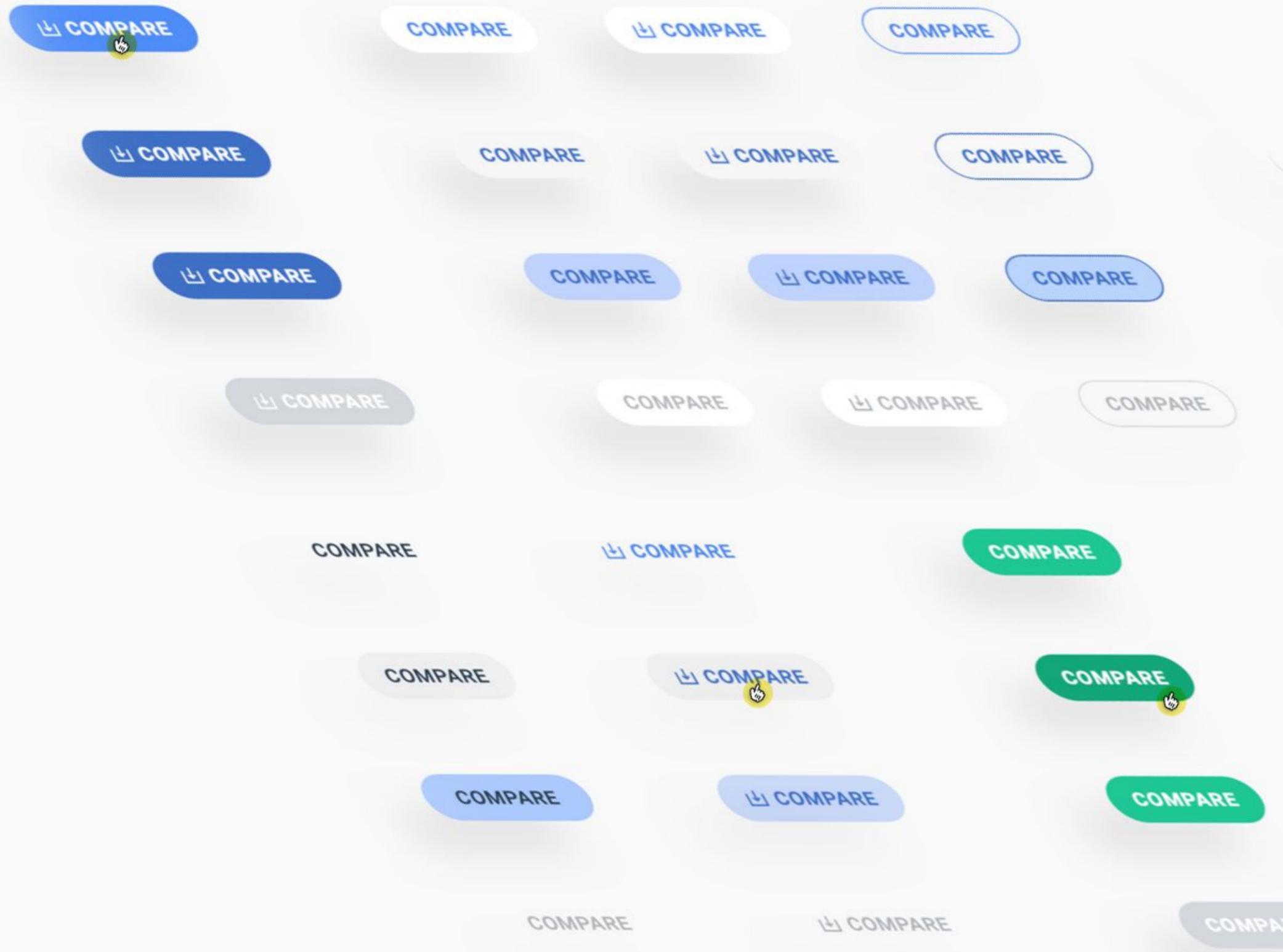
# Design Tokens: espelhando design e tecnologia

Sabemos que um problema comum nas empresas é que o que é **projetado no design nem sempre é reproduzido no desenvolvimento**. Com este ebook você já aprendeu que uma das principais vantagens de um design system é otimizar esse processo, eliminando microdecisões e instituindo uma linguagem comum entre times. Agora, vamos apresentar como isso acontece na prática.

# Otimizando Microdecisões

Quando vamos construir um elemento de uma interface do que nós precisamos? Vamos pensar em um componente simples, como um botão. Para desenhar um botão nós começamos com o formato: é um botão retangular? Ele vai possuir bordas arredondadas?

Depois pensamos em cores: qual é a cor default? Quando ele é clicado muda de cor? Podemos ir além, pensando em **tipografias**, ícones, sombras, entre outros.





## Otimizando Microdecisões

Cada uma dessas variáveis é uma microdecisão que precisamos tomar na hora de construir um componente. Quando pensamos em uma tela multiplicamos este número de microdecisões ainda mais e quando pensamos em um produto inteiro esse número se torna assustador.

E para agravar ainda mais a situação, cada pessoa toma a decisão de um jeito, e no final temos uma compilação de tipos de botões e outros componentes em nossos produtos digitais.

A woman with curly hair and a man are looking at a screen together. The woman is wearing a grey top and a necklace. The man is wearing a white shirt. The background is a soft, out-of-focus light blue and white.

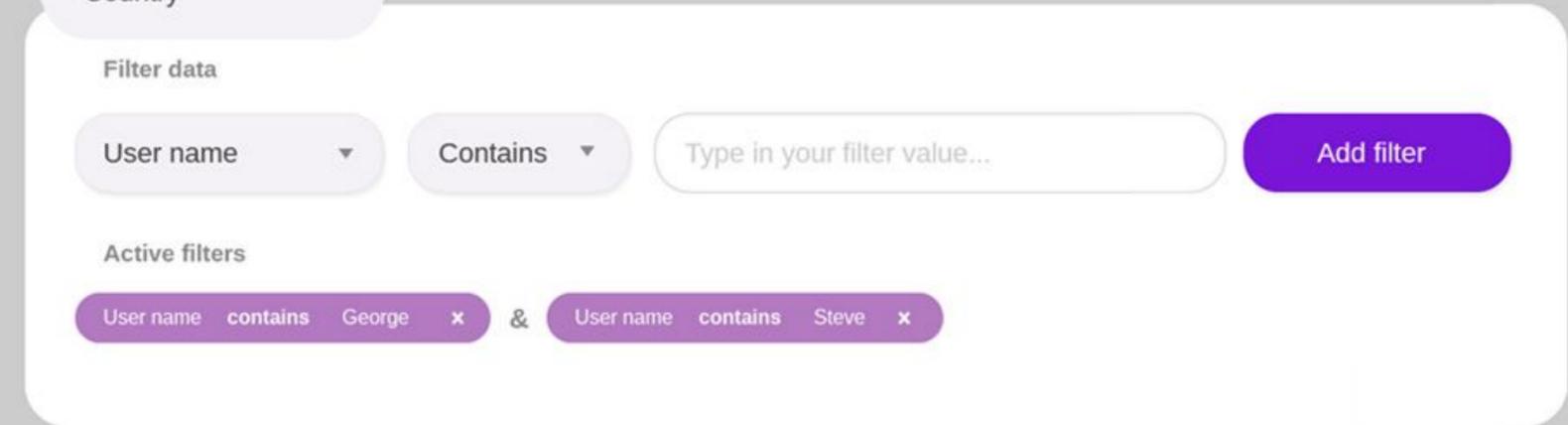
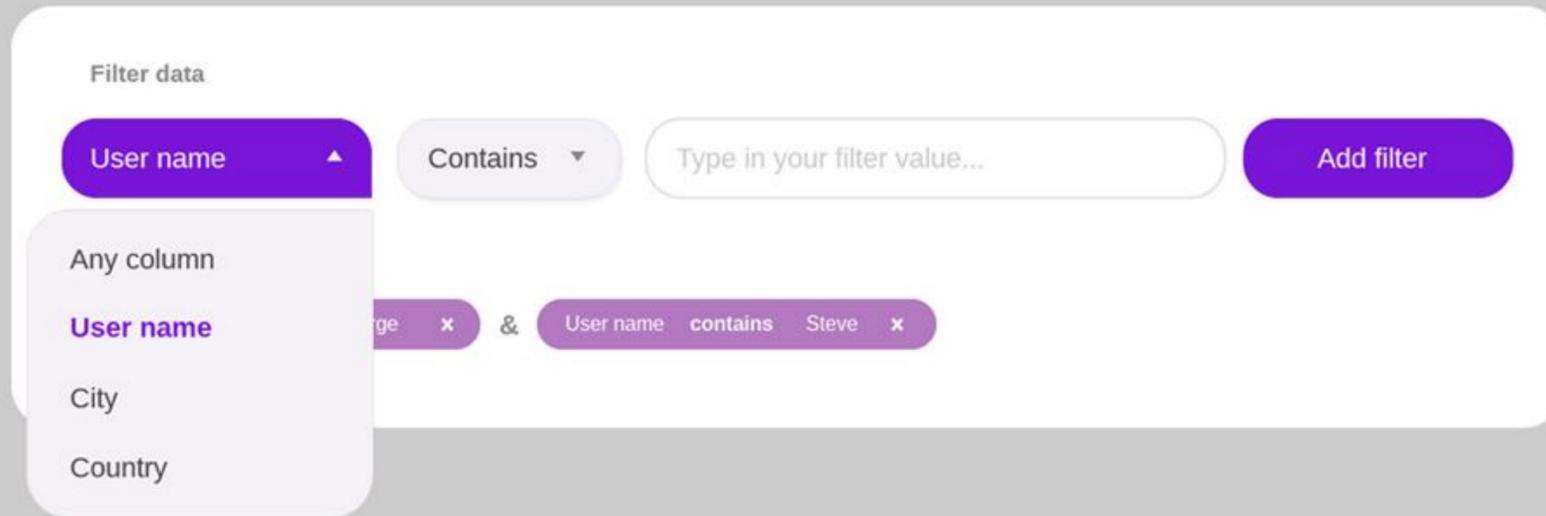
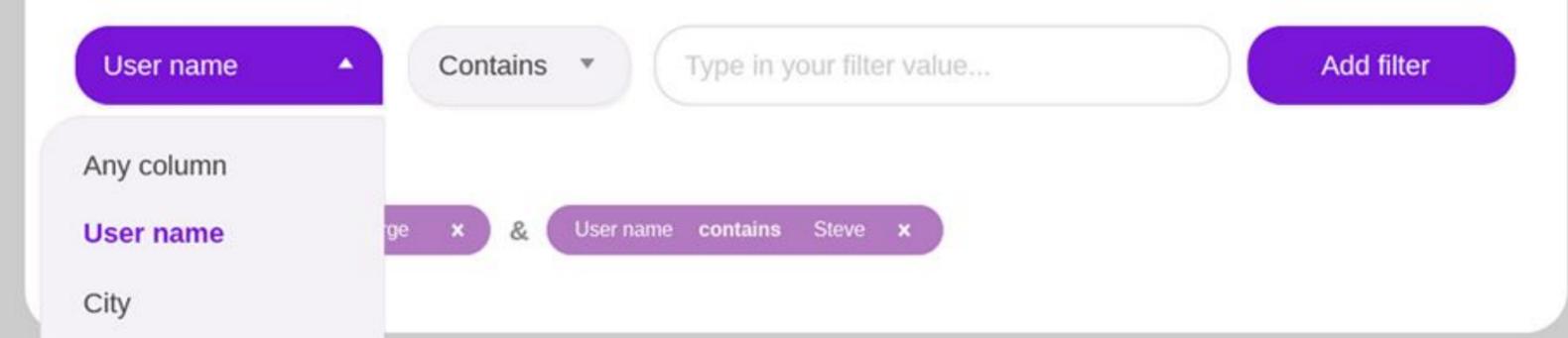
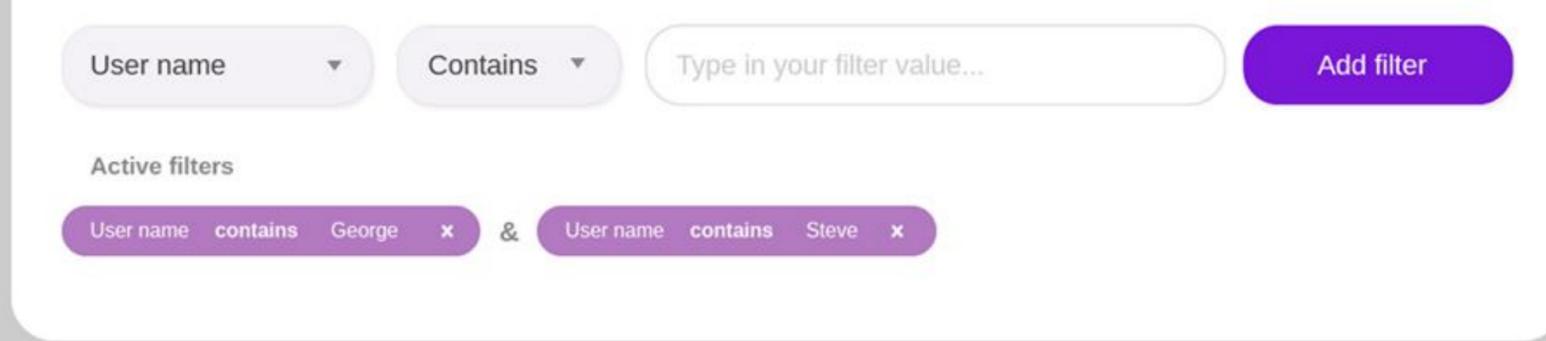
DICAS ÚTEIS

## Para que gastar tempo pensando em decisões que já foram tomadas?

O design system funciona como um repositório de microdecisões. Assim, cada variável em que precisamos pensar quando vamos construir um componente é padronizada de acordo com critérios de qualidade e acessibilidade. Quando você for construir um produto digital não precisa pensar em cores, tipografia, iconografia, espaçamento, sombras e comportamentos — essas decisões já foram tomadas para você.

Portanto, você pode se concentrar na parte estratégica, planejando como combinar todos os elementos para criar as melhores experiências possíveis para os usuários.

Chamamos estas variáveis de **design tokens**, e eles são nossa arma secreta para escalar o design system.



## Afinal, o que é um Design Token?

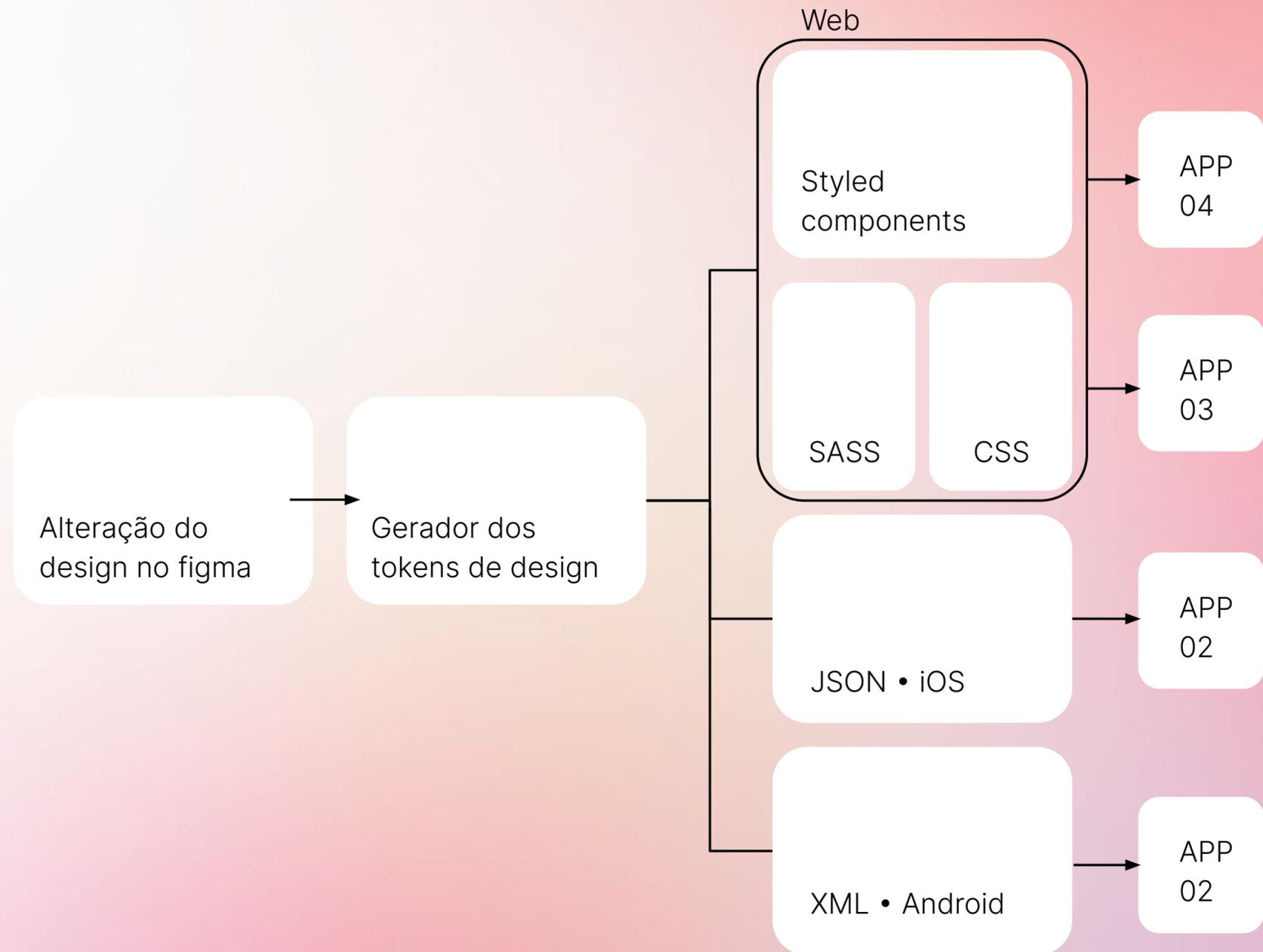
Tecnicamente falando, os design tokens são variáveis semânticas de estilo. Eles operacionalizam a relação entre o design e a tecnologia a partir do espelhamento de todas as opções visuais que um elemento na interface pode ter. Essas opções visuais incluem espaçamentos, tipografia, cores, opacidade, sombra e demais variáveis necessárias para o desenho de um componente.

Ou seja, os Design Tokens são uma maneira independente de armazenar variáveis.

O objetivo é que os elementos que compõem as interfaces dos produtos possam ser compartilhados de maneira consistente entre plataformas, como IOS, Android e sites comuns.

# Como funciona na prática?

Falamos muito sobre instituir uma linguagem comum. Na prática, ela funciona através de uma integração da ferramenta de design com os códigos. Os tokens são gerados por meio de uma leitura do styleguide no design, de maneira automática. Quando um designer altera uma cor no Figma, por exemplo, o gerador de design tokens lê esta informação e a traduz automaticamente para diversas linguagens (CSS, SASS, Styled Components, entre outros).





# Tipos de tokens

Existem três tipos principais de design tokens que podemos aplicar na prática. Eles começam do mais global e vão até um desdobramento mais específico. A seguir, vamos apresentar cada um deles e pontuar algumas diretrizes de uso.



#246DC2



blue-400

# Tokens globais

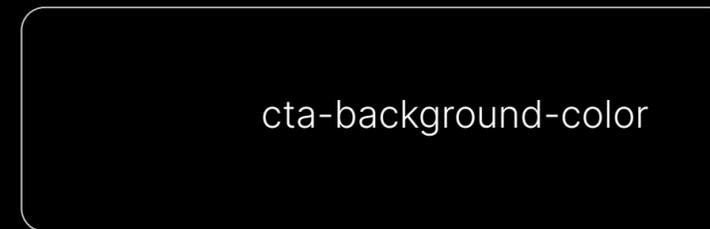
São os **valores primitivos da linguagem de design**, como paletas de cores, sombras e tipografia. Eles são aplicados diretamente pelo time de engenharia e são herdados por todos os outros tipos de tokens. Isso significa que são os mais abrangentes e fundamentais do design system. Assim, são os mais fáceis de referenciar no código, mas são os menos vinculados à lógica de linguagem de design. Portanto, use apenas quando não houver aliases para o seu caso de uso.



#246DC2



blue-400



# Tokens de **alias**

São os que se referem a um contexto específico. Isso significa que eles ajudam a comunicar o objetivo pretendido de um token e são eficazes quando temos um valor que pode ser aplicado em vários locais. Isso significa que um token global como light-gray pode ser traduzido como token de alias background-color, por exemplo. Por isso, os tokens alias são o tipo mais recomendado para a construção dos produtos.



# Tokens específicos de componentes

São uma representação ainda mais específica de todos os valores associados a um componente específico. Usualmente, herdam informações de tokens alias, mas são nomeados de forma que as equipes de engenharia sejam o mais específicas possível em suas aplicações. O token alias de background-color, por exemplo, pode ser traduzido em um token de componente button-cta-background-color. A criação destes tokens é mais complexa, mas potencializa ainda mais a consistência nos produtos digitais.

No final você deve ter folhas estilo em design e código parecidas com essas. É importante disponibilizar os tokens para todos os usuários do design system, para facilitar processos de contribuição. Mas não se preocupe: vamos abordar este assunto em detalhes quando falarmos sobre governança.



Depois que você  
estabeleceu os **design  
tokens** e **design principles**,

já pode começar a construir os seus componentes!

Existem várias maneiras de projetar, mas a abordagem que mais gostamos é começando do menor para o maior. Continue lendo que vamos explicar esta técnica em maiores detalhes.

#05

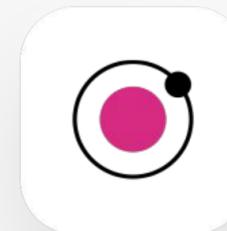
# Componentes: projetando do menor para o maior

Componentes são todos os elementos visuais que compõem as interfaces de um produto digital. Existem diferentes metodologias para a organização desses elementos por meio de categorias e nomenclaturas específicas.

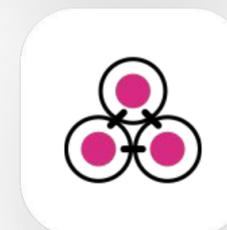
Agora, vamos apresentar como eles funcionam na **prática**.

# Tipos de componentes no **design system**

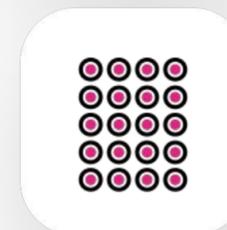
Talvez a abordagem de classificação de componentes mais conhecida seja o Atomic Design, cunhada por Brad Frost. Ela categoriza os componentes fazendo uma analogia com os sistemas biológicos — o que vem de encontro com o pensamento de que o design system é um organismo vivo. Dessa maneira, Frost classifica os componentes de acordo com sua granularidade: átomos, moléculas, organismos, templates e páginas.



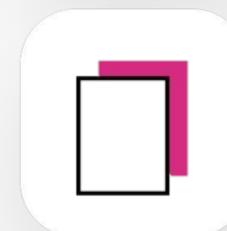
Atoms



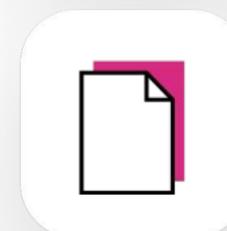
Molecules



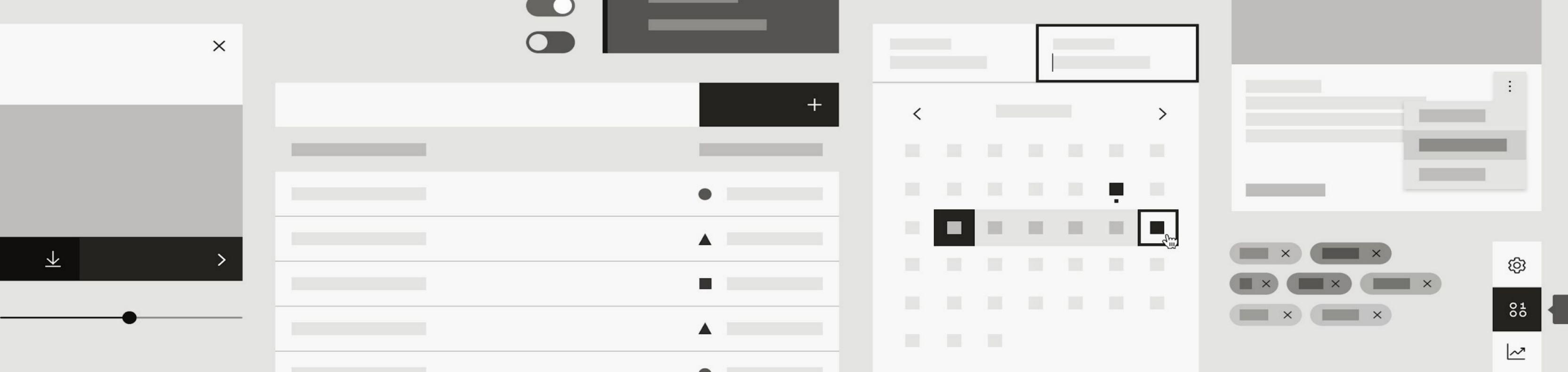
Organisms



Templates



Pages



Átomos, como o próprio nome sugere, seriam os elementos indivisíveis presentes nas interfaces, como um **<label>**, um **<button>** ou um **<icon>**. Assim como na biologia as moléculas surgem a partir de uma combinação de átomos, na abordagem de Frost, os componentes classificados como uma molécula seriam a combinação de componentes indivisíveis. Deste modo, um **<label>** e um **<icon>**, por exemplo, poderiam ser combinados para formar um **<icon button>**. Organismos seriam o agrupamento destes componentes moléculas, como um **<card>**.

Seguindo a lógica, a classificação evolui da menos para a mais complexa. Esta é uma ótima metodologia, respeitada e amplamente aplicada em vários design systems. Mas quando vamos avaliar sua praticidade, muitas vezes gastamos tempo e energia discutindo se uma **<galeria>** seria um organismo ou uma molécula, quando no final o que importa é que ela é uma galeria. Por isso, acreditamos que seja mais eficiente usar uma abordagem simplificada.



Na prática, o que realmente precisamos são duas categorias: **o que é indivisível** e o que é uma **combinação de partes indivisíveis**. A partir desta distinção podemos combinar os componentes indivisíveis com tokens de espaçamentos para criar componentes mais complexos de maneira escalável. Normalmente chamamos estes componentes indivisíveis de Base Components e os componentes mais complexos de Components, pois estes últimos são frequentemente aplicados nas interfaces.



# Nomenclatura

Se design system é sobre estabelecer uma consistência, precisamos estabelecer uma linguagem comum. O que o time de design chamava de dialog, o time de engenharia chamava de modal e a equipe de negócios se referia como “telinha de mensagens” passa a ter um nome só — e isso facilita a troca de informações de uma maneira incrível! Principalmente, mas não exclusivamente, em cenários de trabalho remoto.

Mas como estabelecer um sistema de nomenclatura comum? Spoiler alert: não é uma tarefa fácil! Ainda não encontramos uma metodologia para isso, mas é importante tentar sempre **ser o mais simples possível**. Uma das primeiras dúvidas que surge é: “vamos usar o inglês ou o português?” Não há dúvidas de que precisamos promover a acessibilidade e o português realmente é mais democrático. Mas você se lembra que a verdade absoluta está na tecnologia? Pois é, as linguagem de código já tem termos muito fortes e consolidados em inglês. Descobrimos que o nível de esforço para traduzir alguns termos geraria uma refatoração grande na hora de desenvolver.



Além disso, alguns termos como “**switch**” e “**checkbox**” teriam traduções grandes e não muito funcionais. Como a palavra de ordem é consistência, chegamos à conclusão que a nomenclatura dos componentes seria em inglês. Não usamos o português no título de nenhum componente, nem mesmo para indicar o estado — usamos “default”, “focus”, “disabled”, entre outros. Mas todos os componentes sempre possuem uma documentação em português que detalha o seu uso, aplicação e boas práticas. Estes termos foram rapidamente absorvidos e aplicados no dia a dia, pois seguem uma lógica padrão.

Agora que temos uma linguagem definida para os componentes, como fazer a nomenclatura em si? Estabelecemos uma regra simples: “o que é” + “especificidade” + “estado”. Então o botão primário se torna Button/Primary/Default ou um input de data ativo é chamado de Input/Date/Active. Você pode definir outras regras, mas o mais importante é que elas estejam claras para todas as pessoas usuárias e que elas sigam o mesmo padrão em todas as plataformas e documentações. Não se preocupe, falaremos mais sobre documentação por aqui.

# Mudança de mindset de projeto

Você já aprendeu com este ebook que o design system institui uma nova maneira de trabalhar. Isso acontece principalmente porque o layout das telas projetado pelos designers passa a ser apenas uma **representação visual da interface**. Quando projetamos a partir do design system, a comunicação entre design e engenharia se torna tão fluída que podemos projetar uma tela apenas listando quais componentes devem estar nela. Você já pensou em passar uma lista ao invés de um layout para o desenvolvedor?





Outro ponto que ganha muita eficiência é a **prototipação**. Wireframes passam a não fazer mais sentido quando trabalhamos com um conjunto de componentes responsivos e de alta qualidade, prontos para serem aplicados nas interfaces. Dessa maneira, protótipos de alta fidelidade podem ser desenvolvidos e testados rapidamente pelos usuários, dando mais agilidade para a validação de ideias, testes de hipóteses e processos de inovação.

Esse aspecto abre diversas possibilidades para cocriação, principalmente em design sprints e workshops. Em uma sessão de brainstorm, por exemplo, o time consegue rapidamente tangibilizar as ideias e construir interfaces inteiras usando os componentes. Depois fica fácil conduzir testes de usabilidade com protótipos de alta fidelidade que poderão gerar insights mais coerentes e tangíveis, possibilitando uma validação eficiente da hipótese construída pelo time.



Trabalhar com componentes abre várias possibilidades e impacta positivamente o trabalho de diversos perfis que atuam na construção de produtos digitais. Mas para que tudo isso aconteça é essencial ter uma boa documentação. Vamos te mostrar mais detalhes deste aspecto.



#06

# Se não está documentado, não existe

Como você viu até aqui, design systems são sobre criar um novo modelo de trabalho, baseado em uma linguagem comum. Para que seja realmente escalável é fundamental que tenha uma boa documentação.

# Mas o que é documentação?

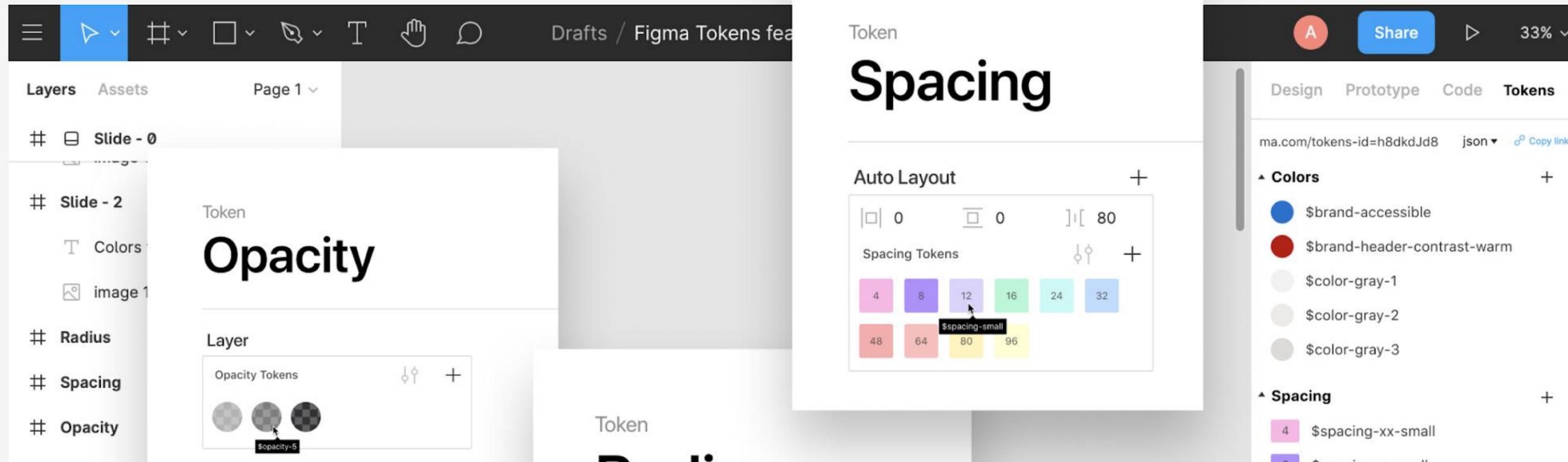
Documentação é o registro das informações necessárias para o pleno funcionamento do seu design system. Ela possui vários níveis: existe uma documentação para cada componente, documentações para conjunto de componentes, documentações de boas práticas e até documentações de processos de atualização e versionamento. Parece ser muito complexo, mas fazer uma boa documentação é fundamental para que o design system seja escalável e possa ser utilizado por várias pessoas. A nossa dica de ouro é: se não estiver documentado, não existe. **Não se esqueça disso!**





Documentamos todos os componentes, processos e métricas que estabelecemos para promover a consistência e qualidade entre os produtos digitais. Mas fazer todas as documentações é uma arte — ao mesmo tempo em que precisamos registrar todos os detalhes, ninguém lê documentações longas demais. Por isso, o segredo é gerenciar entregas de conhecimentos para perfis para os quais eles façam sentido. Para a sobrevivência do design system, é crucial **catalogar as informações e as entregas para as pessoas certas**, no momento certo e no formato ideal.

Para isso, existem dois tipos essenciais de documentação: **documentação técnica** e **documentação funcional**.



## Documentação **técnica**

A documentação técnica vai ser consumida principalmente pelos desenvolvedores e designers. Nela, registramos todos os padrões de estilo, design tokens, comportamentos, parâmetros. Para isso, precisamos falar a língua deles e entregar as informações da maneira com que estão mais habituados. Para a documentação técnica de design, uma boa estratégia é usar o **Figma**, **Sketch** ou **Adobe XD**.

O Figma em especial permite a colaboração e o versionamento em etapas da biblioteca, sendo muito estratégico para documentar os design systems. A documentação técnica para os desenvolvedores deve incluir pacotes de componentes e arquivos de readme. Uma boa estratégia é usar o NPM. Nele é fundamental ter uma folha bem construída de design tokens.



## Documentação **funcional**

A documentação técnica se concentra em registrar todas as informações importantes sobre os componentes, mas sabemos que isso não é suficiente. É importante estabelecer guidelines de boas práticas para que os componentes sejam aplicados nas interfaces da maneira correta. Fornecer estas informações é um dos objetivos da documentação funcional. Nela serão registrados todos os **guias de uso da marca, tom e voz, acessibilidade, testes, qualidade** e demais informações que sejam relevantes para a empresa.

Design Systems tendem a ser complexos e abrangentes por natureza, o que pode resultar em um portal extenso com uma navegação confusa. Por isso é imprescindível que a documentação funcional contemple também guias de onboarding, que demonstrem para usuários de diferentes perfis de como usar o design system. O onboarding desempenha um papel fundamental na adoção contínua, por isso é essencial que seja elaborado com atenção e cuidado.

A photograph of three people in a modern office setting. On the left, a woman with long brown hair, wearing a light blue shirt, is smiling and looking towards the center. In the middle, a woman with long dark hair and glasses, wearing a grey top, is looking at a tablet. On the right, a man with dark hair and a goatee, wearing a red and black checkered shirt, is holding the tablet and looking at it. The background shows a modern office with glass partitions and exposed ceiling pipes.

Com a documentação completa, o design system pode começar a promover a **consistência e a qualidade entre produtos.**

Mas para que ele evolua, é essencial ter padrões de governança claros. Portanto, vamos te explicar sobre estes padrões.

#07

# Design Ops: governança e cultura

Design systems são complexos por natureza, construir e manter um design system pode se tornar facilmente um full-time job. Entretanto, existem diversas estratégias para coordenar esse sistema e fazer com que ele evolua e se torne permanente na cultura da empresa.

A primeira coisa que precisamos falar é sobre a **governança** — essa palavra pouco amigável, mas tão importante. A governança em um design system é responsável por garantir que o sistema evolua e atenda as necessidades da empresa com o passar do tempo, facilitando processos e tomadas de decisões. O desenvolvimento de produtos digitais é o que mantém o design system vivo. Ao mesmo tempo que eles bebem na fonte do design system, eles também o modificam, gerando necessidades de componentes novos, atualizações ou informações adicionais, que suportam seus objetivos finais.

Sendo assim, um design system **nunca está 100% pronto**, dependendo de estratégias e processos que permitam sua evolução.

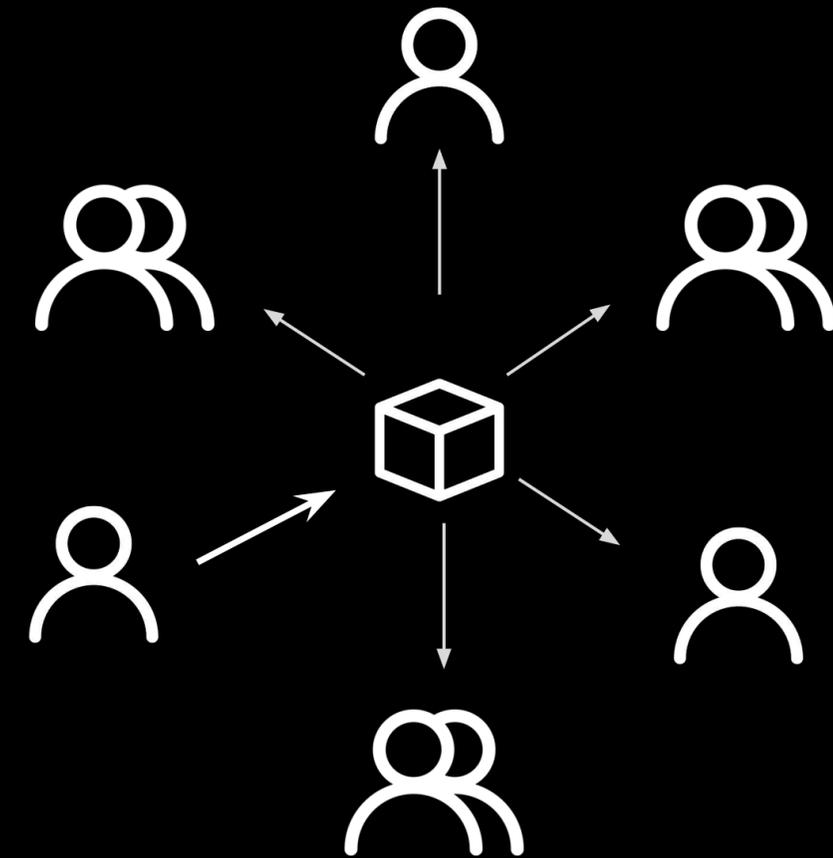
A seguir, iremos abordar brevemente três modelos de governança habitualmente aplicados à design systems.



# Modelo **solitário**

Em alguns casos, o design system começa com a iniciativa de uma pessoa dentro da empresa. Este cenário pode emergir de diversas maneiras. É possível que a pessoa tenha obtido o buy-in de diferentes stakeholders no início, que posteriormente acabaram abandonando ou esquecendo o projeto. Talvez seja a iniciativa de um diretor ou designer apaixonado pela causa, ou que veja a potencialidade ou necessidade do design system dentro da empresa. É muito raro que apenas uma pessoa seja capaz de sustentar o desenvolvimento do sistema e lidar com demandas, atualizações e dúvidas decorrentes de sua adoção.

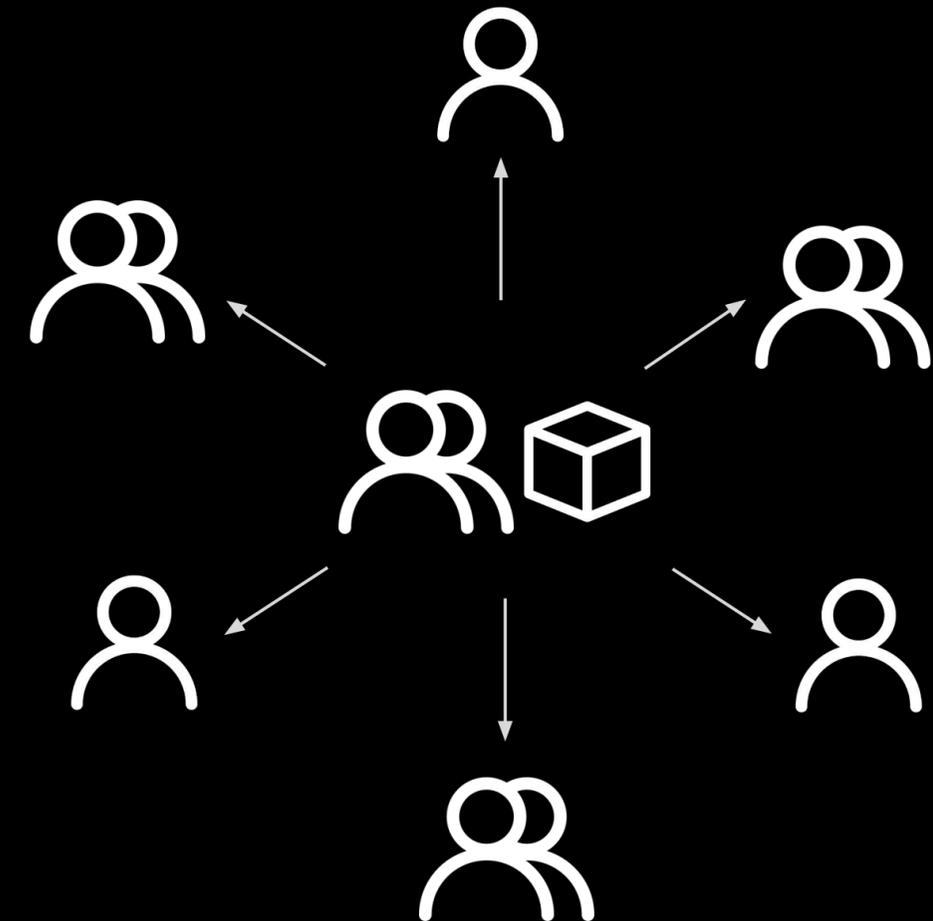
Existem casos de sucesso, que normalmente são design systems mais restritos e com caráter imediatista, como o [PantSuit Design System](#), criado por Mina Markham para unificar a identidade visual da campanha presidencial de Hillary Clinton. Entretanto, na maior parte dos design systems, este modelo se mostra eficiente para iniciar o projeto, mas na medida em que o sistema cresce, se torna cada vez mais difícil administrar as atualizações e manter os stakeholders e makers engajados, resultando em uma solução superficial.



# Modelo **centralizado**

Outra alternativa para modelos de governança é a centralização. Neste caso, ao invés de ter apenas uma pessoa responsável, é estabelecido um time para construir, coordenar e escalar o design system dentro de uma companhia. Normalmente, este time é completamente dedicado ao design system. Com esse modelo é possível obter uma visão sistêmica do contexto de produção de produtos digitais da empresa. Existe um grande ganho em consistência e eficiência de produção, além da redução de vieses, como o favorecimento de produtos específicos.

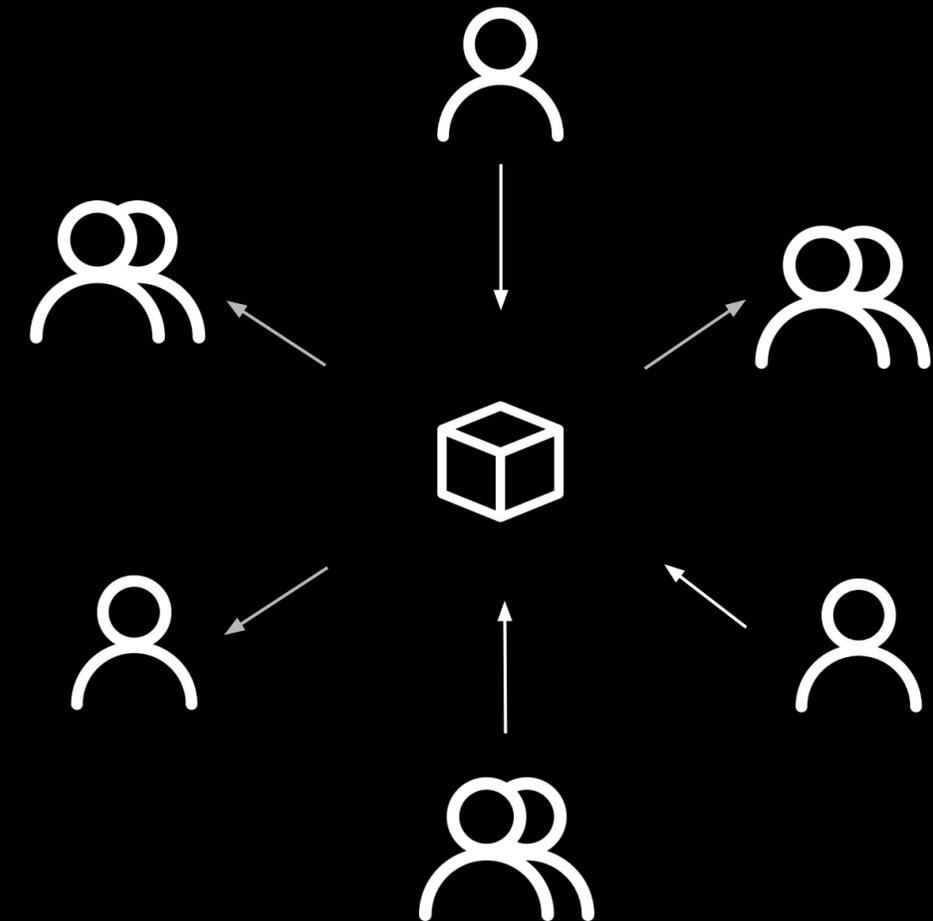
Porém, em muitas empresas que utilizam esta abordagem foi observada uma tendência de afastamento dos membros do time da rotina de desenvolvimento, resultando na dificuldade de compreensão das necessidades e dores dos desafios específicos que envolvem o dia a dia de produção. Ainda sim, algumas estratégias podem ser adotadas para mitigar este fator, como a realização de auditorias e entrevistas periódicas.



# Modelo **federativo**

Baseado no pensamento sistemático popularizado pelo Design System da Hubspot, esta abordagem se baseia no empoderamento de designers e líderes digitais alocados em times de produtos, que são convidados a colaborar com o design system durante um certo período de tempo. Logo, a percepção dos desafios do dia a dia são trazidas para discussão, aumentando a legitimidade das soluções e o potencial de evangelização. A diversificação de integrantes do time entre equipes de desenvolvimento de produtos diferentes potencializa a diluição de necessidades, reduzindo possíveis vieses.

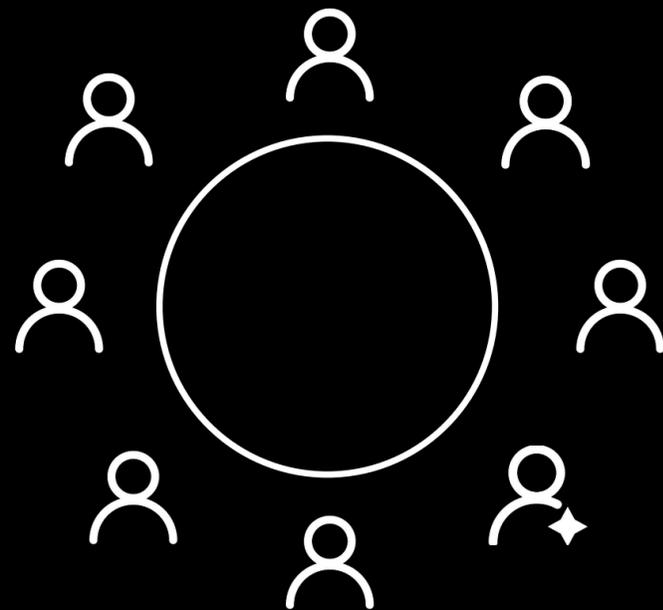
Entretanto, por não estarem 100% dedicados ao design system, existe o risco do sistema ser despriorizado, frente à necessidades e desafios específicos dos produtos. Além disso, este modelo introduz desafios práticos de tomada de decisão, caso não seja estabelecido um canal eficiente para comunicação entre os membros dos grupos. Ao estabelecer um time federado é importante contar com representatividade entre produtos, plataformas e disciplinas.





# Modelos **híbridos**

A escolha de um modelo deve ser baseada e adaptada para a **cultura e estrutura da empresa** para qual o design system está sendo desenvolvido. Conheça dois cases de sucesso: Salesforce e Etsy.



Design System Library

# O modelo Cíclico da **Salesforce**

A Salesforce possui mais de 200 designers de produto, nove grandes grupos de produtos digitais e diversas aquisições, tornando muito difícil manter a consistência neste amplo universo. A solução trazida pelo seu design system se tornou muito popular, ao mesclar as abordagens federada, centralizada e solitária estabelecendo os Theme Design Councils, os Theme Leads e o Process Lead, respectivamente.

Os Theme Design Councils são compostos por designers que trabalham com produtos que abordam temas relacionados. O objetivo deste grupo é criar as guidelines de design para os assuntos relativos ao tema. Existem diversos Councils, cada um sendo composto por 6 a 12 integrantes. Os encontros acontecem semanalmente e abordam tópicos que requerem alinhamento ou desenvolvimento dentro do design system.



Cada tópico recebe um dono, que executa as tarefas combinadas e as apresenta para o Council para avaliação, incorporando posteriormente os feedbacks recebidos. Cada Theme Design Council possui um Theme Lead, que atua como facilitador das reuniões e ponto primário de contato, organizando as tarefas e centralizando as comunicações. Os Theme Leads de cada Council se reúnem periodicamente para compartilhar o que está sendo desenvolvido, proporcionando uma visão global do design system.

Esta reunião é facilitada pelo Process Lead, que é a pessoa responsável pelo design system dentro da empresa. O Process Lead é responsável por identificar mudanças, alinhar expectativas, definir regras e documentar os processos do design system. Ele trabalha de maneira full time, a partir de uma visão sistêmica, dando suporte à todos os Councils e aos demais usuários do design system. Em última instância, atua como educador, comunica progressos, updates e mudanças, mantendo o design system vivo e aderente às necessidades da empresa.



# ○ **“Design System Council”** da Etsy

O design system da Etsy passou por várias fases ao longo dos anos, experienciando diversos modelos de governança. Em 2016, foi criado o primeiro time dedicado, a partir de uma abordagem centralizada, mas não exclusiva. Dessa maneira, um grupo de designers e engenheiros era responsável por manter e construir o design system, acumulando essas tarefas com a atuação em times de desenvolvimento de produtos digitais na empresa. Ao longo do tempo as demandas foram crescendo e foi se tornando cada vez mais claro a necessidade de um time full-time para manter o design system.

O time acreditava que ao possibilitar que uma equipe de designers, engenheiros e product managers mantivesse o foco exclusivamente na manutenção, evolução e aprimoramento do design system, o restante dos colaboradores poderiam se dedicar na construção de experiências para os produtos digitais da Etsy. Entretanto, foi criado um mecanismo para que qualquer colaborador pudesse sugerir modificações ou acréscimos ao design system, mas a maior responsabilidade era do time central, inteiramente dedicado ao sistema.



Ao passar do tempo, os times de produtos digitais foram trabalhando em um ritmo cada vez mais acelerado, ao ponto em que a equipe do design system não conseguisse endereçar as crescentes demandas de necessidades específicas, suporte e feedback. Para remediar esta situação, a equipe instituiu o Design System Council, que mudou a abordagem em direção para um modelo mais federativo.

Entretanto, além de contar com representatividade entre produtos, plataformas e disciplinas, a equipe foi além, agregando especialistas em diversos assuntos, ou gurus. Dessa maneira, o comitê contava com uma equipe centralizada, um time de consultores em plataformas, produtos e disciplinas diferentes e um grupo de especialistas das mais diversas áreas: um guru em acessibilidade, um guru em customer experience, um guru em análise de dados...

O Design System Council passou a se reunir uma vez a cada duas semanas para avaliar, propor e tomar decisões sobre mudanças no sistema. O time de consultores e gurus se tornaram responsáveis pela divulgação dos updates para suas equipes. Os consultores participavam do comitê por seis meses e depois eram substituídos, fazendo com que mais pessoas fossem envolvidas. Esta estratégia aumentou a visibilidade do Design System dentro da empresa, mantendo os tópicos de discussões e as atualizações aderentes às necessidades dos times e produtos digitais. O comitê potencializou o sentimento de dono e o posicionamento de colaboradores, como gurus ou experts, dentro da empresa teve um impacto positivo. Por fim, foi possível perceber ainda que esta iniciativa potencializou o pensamento sistêmico e global, fazendo com que os times conhecessem iniciativas e soluções além dos produtos em que estavam trabalhando.



Seja qual for o modelo que você escolher, é importante estabelecer um canal de **comunicação eficiente**

e papéis e responsabilidades claros para dar velocidade nos processos.

#08

# Versionamento e atualização

Como vimos até aqui, **o design system é um organismo vivo sempre em evolução.**

Sendo assim, nem todos os componentes dos produtos digitais são incluídos nas bibliotecas e style guides dele. Por mais que a consistência seja essencial, é importante prezar pela flexibilidade, de maneira com que as métricas dos produtos que estão sendo construídos sejam a prioridade.

Dessa forma, existe abertura para a criação de novos componentes ou variações de componentes existentes. Estas criações, caso tenham potencial para serem utilizadas por outros produtos digitais da empresa, devem ser posteriormente adicionadas no design system, possibilitando o uso e acesso por outros times através do versionamento da biblioteca.





Se existir a necessidade de um componente muito específico para o atendimento de uma demanda de um produto em especial que não tenha grande potencial para ser usado em outros produtos digitais da empresa, **ele não precisa necessariamente ser incorporado ao design system.**

Nesse caso, o time do produto não precisa seguir o passo a passo de contribuição, mas é essencial seguir os design tokens e os padrões de estilo e qualidade estabelecidos.

# Metodologia de versionamento

Recomendamos o uso da metodologia de versionamento Semver para coordenar todas as atualizações do design system. Em linhas gerais, a metodologia Semver estabelece um método de versionamento baseado em três números **MAJOR.MINOR.PATCH:**

Uma mudança no número **MAJOR** deve ocorrer quando são realizadas mudanças na API que desencadeiam quebras ou incompatibilidades.

Uma mudança no número **MINOR** deve ocorrer quando são adicionados novos componentes, que não geram incompatibilidade.

Uma mudança no número **PATCH** deve ocorrer quando são realizadas correções de bugs ou pequenos aprimoramentos nos componentes, que não geram incompatibilidade.

O Figma e o Zeroheight são só representações visuais do código, então o mais importante é versionar o NPM.

# Tipos de releases

Além de estabelecer uma cadência de releases, é importante diferenciar as releases especiais. Todas as releases de novas versões do design system devem ser comunicadas a todos os usuários. Entretanto, quando pensamos em potencializar a adoção contínua do design system na empresa, mudanças diferentes exigem maneiras distintas de divulgação e comunicação. Dessa maneira, é uma boa prática estabelecer duas categorias de versões: as Releases regulares e as Releases especiais.



# Releases **especiais**

As Releases especiais, por sua vez, também devem seguir a metodologia padrão de registro no Changelog, mas devem ser utilizadas para a divulgação do design system, uma vez que elas incluem o lançamento de novas features, componentes ou guidelines de grande valor para os usuários. Elas podem ser usadas, também, para comunicar mudanças urgentes, como atualizações devido à quebras de segurança. Recomendamos a construção de newsletters ou outros meios de divulgação para dar ênfase neste tipo de release.

# Releases **regulares**

As Releases regulares são aquelas originadas de mudanças corriqueiras no design system, como adição de componentes ou variações de componentes existentes, correção de erros pontuais ou pequenas mudanças no estilo. Elas devem seguir a metodologia padrão de registro de mudanças no Changelog, respeitando as boas práticas Semver. Elas devem ser comunicadas aos times de produtos digitais por meio das ferramentas de design, desenvolvimento e da documentação funcional.

Agora, você já tem as informações iniciais para começar a construir um design system! Aproveite o conteúdo deste ebook e releia sempre que precisar, todos os detalhes são importantes para você conquistar um ótimo resultado. Quer aprofundar no conhecimento sobre Design System? Fizemos um curso especialmente para você.

Faça sua pré-inscrição!

**ioasys**