

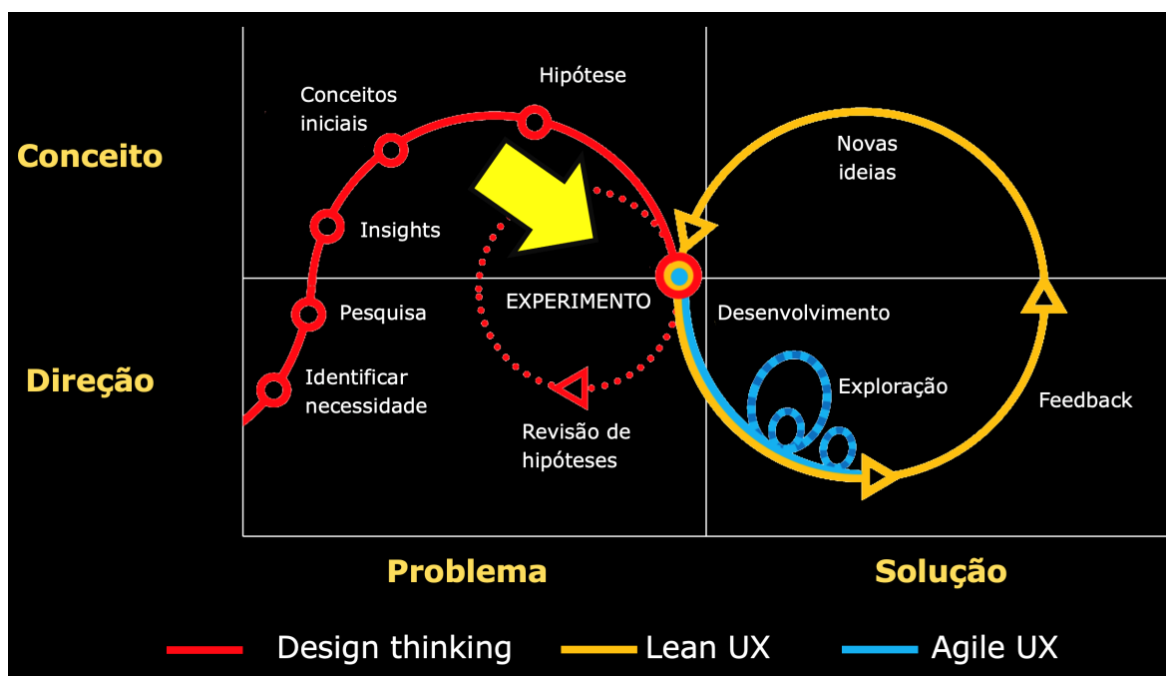
TRANSFORMAÇÃO DIGITAL EM MULTI-X

INTRODUÇÃO

Nesta etapa, passaremos por um conteúdo de teor prático sobre um importante pilar dos projetos digitais: o design. O tema vem ganhando importância cada vez maior em projetos digitais – mas não exclusivamente. A explicação mais simples é o fato de que todo projeto digital necessariamente precisa de um suporte de telas, e, com telas, não há manuseio e experiências táteis que ajudem no uso de um produto ou serviço. Tudo é resolvido por interfaces gráficas e conteúdo. Grandes corporações já internalizaram times de UX, que são dominados por profissionais de design gráfico/design de interfaces.

Passaremos por conteúdos que irão desde a prototipação até os princípios básicos do design de interfaces contemporâneo.

TEMA 1 – PROTOTIPAÇÃO: CONCEITO E BREVE HISTÓRIA



Fonte: Cao, 2015.

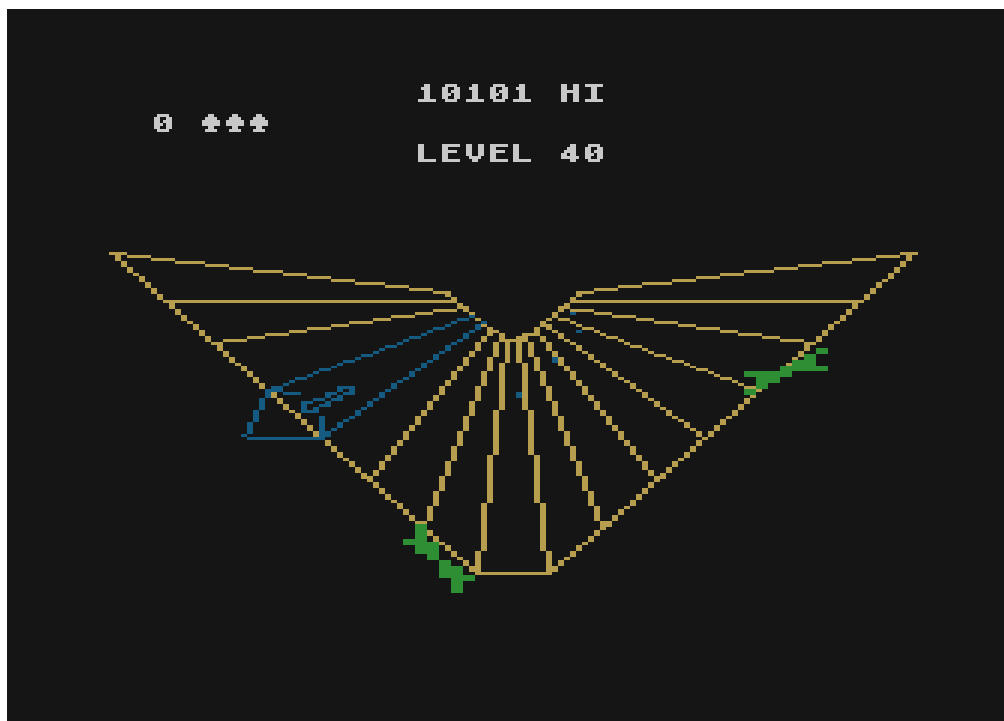
A matriz acima consolida e organiza as 3 metodologias mais usadas (e complementares) do UX Design atual. A prototipação é um estágio que sucede as fases de pesquisa e os estudos iniciais e precede o desenvolvimento (de interface e código). Considerando a importância da velocidade e dos ciclos curtos de testes e aprimoramentos, o protótipo é vital para minimizar erros oriundos de interpretação e subjetividade. É por meio dele que o usuário tem uma noção bastante próxima da solução final: como os elementos se

comportam, que ação leva a que destino, que fluxos estão presentes para resolver os seus problemas.

Além de eliminar a subjetividade do processo, o protótipo tem uma estrutura muito mais simples para ajustes, tornando o ciclo de refinamento mais ágil e menos custoso.

1.1 A linha do tempo da protipação

- **1970** — Desenvolvimento de software ainda dominado pelo método Cascata. Sistemas que eram operados essencialmente por linha de comando, sem grandes desafios de simulação e prototipação;
- **1975** – Reconhecimento e desenvolvimento da Arquitetura de Informação, um importante marco para chegar ao que temos hoje. Reconhecer a arquitetura é reconhecer o usuário, suas características e necessidades na operação de sistemas e softwares;
- **1980** – Primeiros protótipos visuais digitais. Ainda bastante rudimentares e construídos pixel a pixel, linha a linha de código, mas já cumpriam o principal papel de avaliação da solução final;
- **1983** – Protótipo do jogo Tempest, para o Atari 5200, nunca lançado;



- **1986** — Lançamento do primeiro software de design: Adobe Illustrator. Um marco na indústria gráfica o Illustrator é utilizado até hoje pela maioria

esmagadora dos designers gráficos. A Adobe, empresa que produz o Illustrator, tem hoje o Adobe XD, software muito mais adaptado para prototipação e design web;

- **1990** — Lançamento do Adobe Photoshop;
- **2000-2003** — Lançamentos dos primeiros softwares de prototipação complexa: Omnigraffle, Axure e iRise;
- **2001** — Manifesto Ágil;
- **2005 a 2008** — Lançamentos dos primeiros sistemas SaaS de prototipação: MockupScreens, Gliffy, Jumpchart, Balsamiq, Protoshare e Justinmind;
- **2011 a 2013** — Lançamentos da nova geração de softwares de prototipação Hi-Fi, permitindo que os trabalhos se aproximem muito do produto final: UXPin, InVision, Flinto, POP e Marvel.

TEMA 2 – PROTOTIPAÇÃO: PRESENTE E FUTURO

A demanda pela evolução da prototipação é contínua, e o motivo é bastante óbvio: quanto mais fiel à solução final, melhor será o resultado de sua avaliação. Podemos dizer que, hoje, o que limita a evolução desta área são os softwares disponíveis. Conforme novas funcionalidades surgem, os designers rapidamente se adaptam e as utilizam.

No entanto, por quais caminhos essas evoluções seguem? A fidelidade de imagem aumenta, a quantidade de telas aumenta, os devices e softwares evoluem. Mas, temos algumas apostas mais conceituais e estruturantes pra essa área, que devem nortear o desenvolvimento das próximas versões e soluções de software de prototipação:

- Velocidade: módulos prontos por *device*, recursos de ajuste para tela, conexão entre áreas/páginas; tornar o ciclo de repetição o mais veloz possível.
- Colaboração:
 - Interação direta: a próxima fronteira é tornar a prototipação acessível para outras áreas além do design, para que nele possam colaborar diretamente. Não falo aqui de comentários, textos (recurso que já existe em sistemas como Invision e Adobe XD, por exemplo), mas

de tornar a edição do protótipo algo mais acessível e democrático. Sabemos o quanto pode ser polêmico “liberar” a edição de uma interface gráfica para profissionais sem o conhecimento mais aprofundado, mas certamente há campo para esse desenvolvimento. Lembremos que, até poucas décadas atrás, a fotografia era uma arte para poucos.

- Low/no-code: acesso à prototipação sem a necessidade de codificação ou operações mais complexas.

2.1 Microinterações

Pode parecer um tanto polêmico diante do conceito de velocidade e produtividade, mas a corrente que defende as microinterações defende que o sucesso de um projeto pode residir nos detalhes.

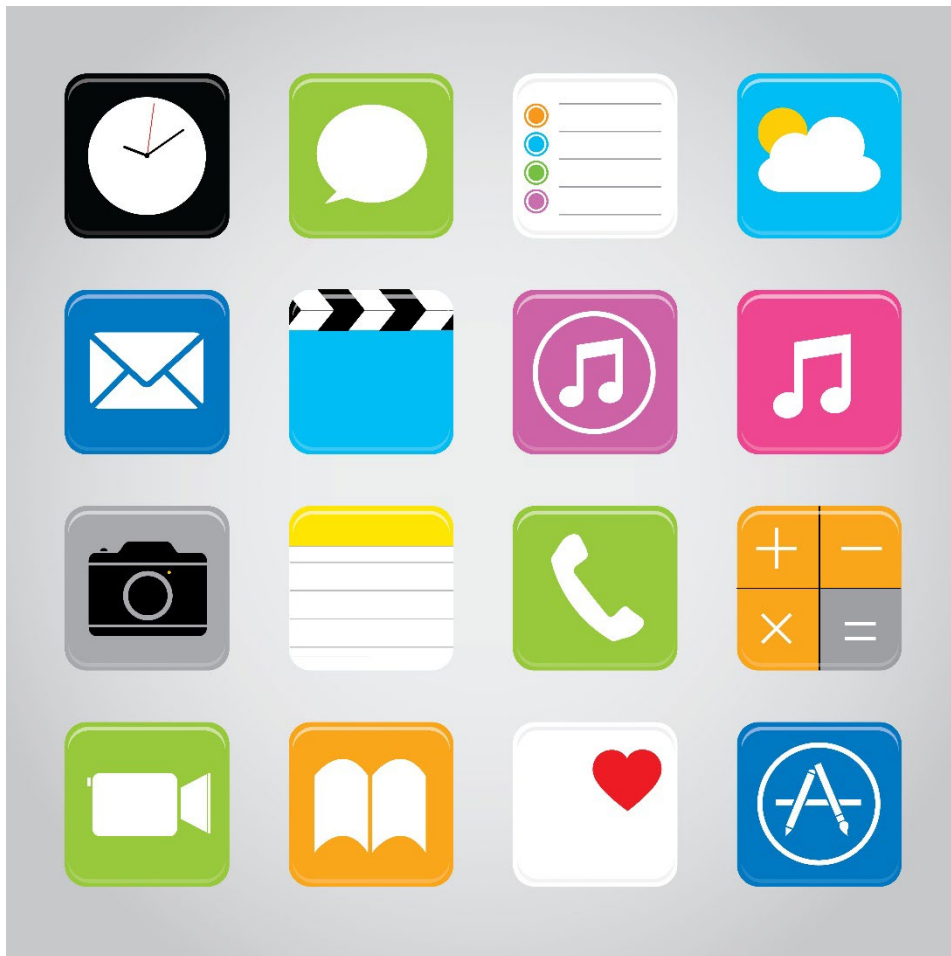
A microinteração consiste em um escopo focado em um elemento específico da interface. O ciclo de uma microinteração presume a alocação de uma microiteração, ou seja, um ciclo de escopo/ protótipo/ teste/ ajuste/ reteste/ desenvolvimento realizado com extrema rapidez. Caso contrário, a microinteração pode arrastar o cronograma de um projeto indefinidamente.

Para ser classificado como uma microinteração, o protótipo deve obedecer a alguns critérios:

- 1) Ser desenhado para um único caso de uso. Isso pressupõe que, ao narrar a ação, o usuário tem ao seu serviço apenas 1 atividade.
- 2) Sempre é disparada por um gatilho, seja ele humano, pelo uso da interface, seja um sistema pelo sistema.

São exemplos de microinterações:

- Os ícones dos apps no iOS que “tremem” quando o usuário segura o toque, para editar o posicionamento dos mesmos (gatilho de usuário);



Crédito: enjoyeverytime/Shutterstock

- As luzes coloridas que giram e alteram a cor no topo do assistente Google Home quando ele recebe o comando de voz “Hey Google” (gatilho de usuário);



Crédito: Pianodiaphragm/Shutterstock

- Notificações de nova mensagem (gatilho de sistema);
- Controle de volume em um vídeo player (gatilho de usuário). Notemos: o vídeo player em si não é uma microinteração, por possuir diversos casos de uso e gatilhos (*play*, *pause*, *stop*, próximo, *zoom* etc.).

TEMA 3 – FORMATOS DE PROTOTIPAÇÃO

Não há somente uma forma de antecipar uma solução. Abordaremos as duas formas de protótipos mais populares em projetos digitais.

3.1 Wireframes estáticos

Em tradução simples, *wireframe* é uma “estrutura de arame”. No digital, o *wireframe* funciona como um esqueleto visual. Um protótipo mais rudimentar que não deve, por definição, refletir o design e nem a interação com os elementos de interface. É uma referência mais abrangente, desenvolvida para que o cliente compreenda que componentes/funções existem em cada área de uma aplicação ou página de um site.

Para a construção de um *wireframe*, é necessário que o designer tenha uma boa noção da arquitetura de informação do site ou aplicação, as principais áreas, o menu, as conexões entre elas. De posse desse conjunto de informações e um software específico, como o Sketch, Figma, Adobe XD ou Invision, o designer trabalha em uma visão parecida com a que podemos ver a seguir:



Crédito: Chaosamran_Studio/Shutterstock

Por mais que ainda seja bastante usado em projetos mais simples, como *websites*, o *wireframe* estático está com os dias contados. Entre as razões para isso, podemos citar:

- É uma entrega exclusiva do designer, seu processo de desenvolvimento é fechado em um departamento apenas;
- O WF estático, por sua própria natureza estática, não prioriza ou explora o uso da solução. Comandos, animações, funcionalidades não são reproduzidas com os recursos usados neste tipo de protótipo;
- É necessário um grande tempo para ser construído e, normalmente, é 100% descartado na etapa de desenvolvimento;
- Por ser um “subprojeto” muito distante da solução final – não tem nenhum acabamento criativo, funcionalidades aparentes, conexões –, ele não garante a qualidade final do produto. E, considerando o tempo tomado por esta etapa no cronograma, é possível que ele surta o efeito contrário: ou “espreme” as fases seguintes de design e desenvolvimento, ou torna o projeto mais longo.

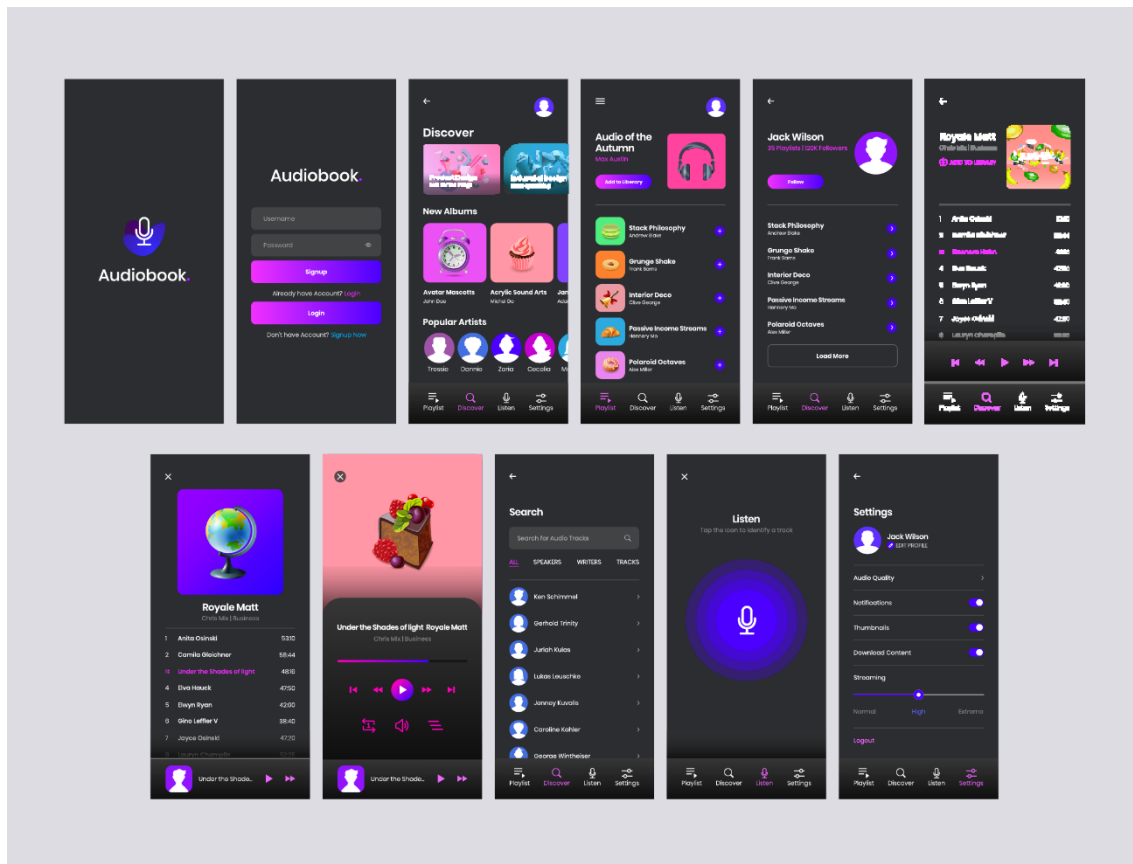
3.2 Protótipos ou *wireframes* interativos

São versões mais simples, porém muito mais fiéis ao produto final. O designer trabalha em sua aparência já pensando na proposta final. Cores, formatos, posições, interações (microinterações também, se for o caso) são construídas de forma que, para um visualizador leigo, até podem passar por uma aplicação já final.

Em um protótipo, é possível simular o uso efetivo da solução. Cliques, toques, funções, caminhos, tudo é simulado não em um nível “*pixel perfect*”, ou perfeito em detalhes visuais, mas já com boa parte da bagagem criativa. Os protótipos podem ser exibidos em vídeos, com roteiros que simulam o uso da solução, e também em situações reais de uso, como um protótipo de um aplicativo móvel que pode ser testado utilizando um smartphone – normalmente de um tamanho específico, ainda sem os recursos de responsividade.

Sendo o estágio mais atual da prototipação, é neste tipo de trabalho que estamos mais limitados por softwares. Conforme estes evoluem, os protótipos também o fazem. Lembremos do futuro da prototipação: quanto mais eficaz, colaborativa e veloz é a fase de prototipação, maior será a garantia de entrega de valor de um projeto.

Abaixo, temos a visualização de um software moderno de prototipação:



Crédito: Usman/Adobe Stock.

Observem o apuro visual do projeto quando comparado com o *wireframe* estático. Os toques/cliques também são visualmente configurados, bem como os estados de botão, animações, transições entre páginas etc.

TEMA 4 – DESIGN SYSTEM

Avançando na cadeia de desenvolvimento de projetos, passada a fase de protótipo, chegamos à fase de desenvolvimento do design final da solução.

Por mais que as ferramentas evoluam, a fase de design é uma fase que tende a ser bastante trabalhosa. Envolve muitos detalhes, e sua extensão é proporcional à complexidade das interfaces que serão criadas, quantidade de casos de uso, especificidades do público usuário.

O *design system* é um sistema criado para otimização desta fase. Sua lógica é a “componentização” do trabalho, a divisão em partes segundo sua utilidade e natureza dentro da aplicação à qual servirá. Por mais que seja uma forma de documentação, não é uma frente que “fere” os princípios do desenvolvimento ágil, muito pelo contrário: seu objetivo é agilizar e padronizar o desenvolvimento, conferindo, ao mesmo tempo, a velocidade e a qualidade do

produto final. É também por meio do *design system* de um projeto que um novo funcionário pode dar continuidade a um trabalho, diminuindo bastante o tempo gasto para aprendizado.

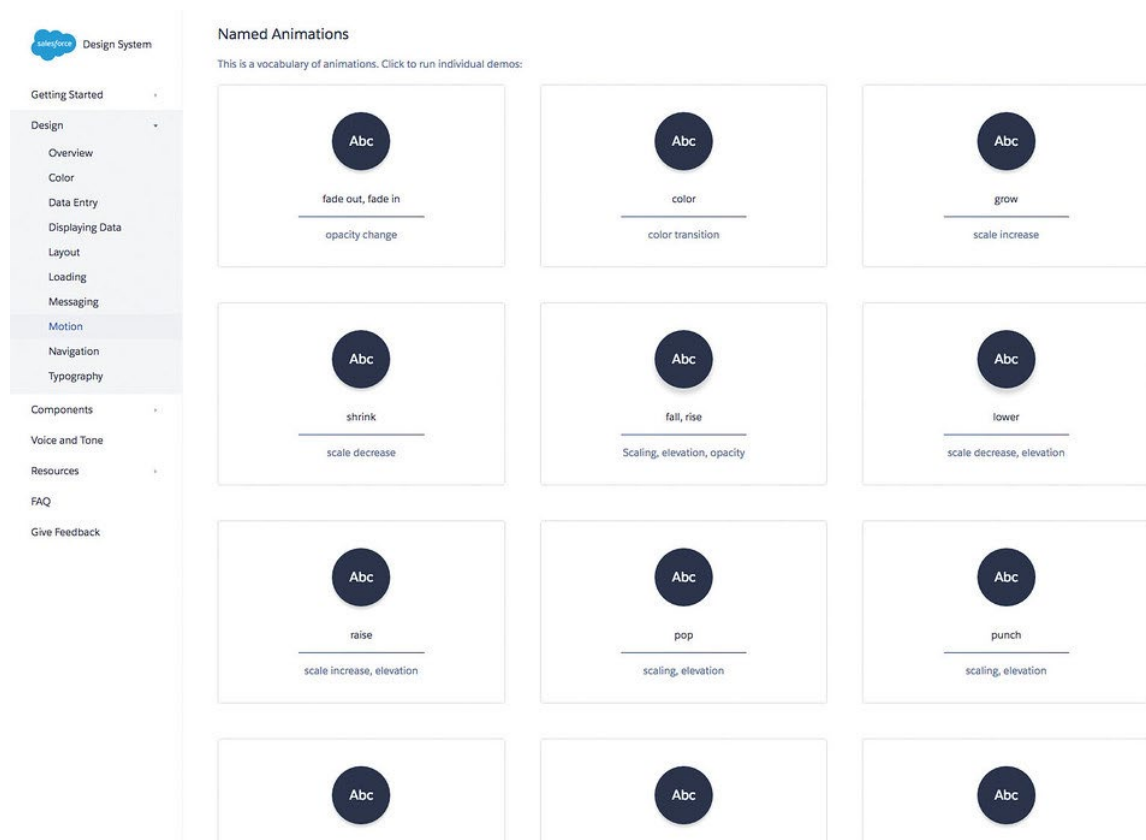
Os *design systems* chegam a um nível tão aprimorado de especificação que se tornam um produto em si, servindo a um outro produto.

Os capítulos de um design system são:

- **Conceitos** / tutoriais: capítulo que contém os fundamentos mais teóricos do design daquela solução ou plataforma;
- **Estilos**: regras para o uso de cores, ícones, tipografia;
- **Componentes**: todos os padrões de elementos reutilizáveis, como botões, barras e menus de navegação, barras de *scroll* etc.

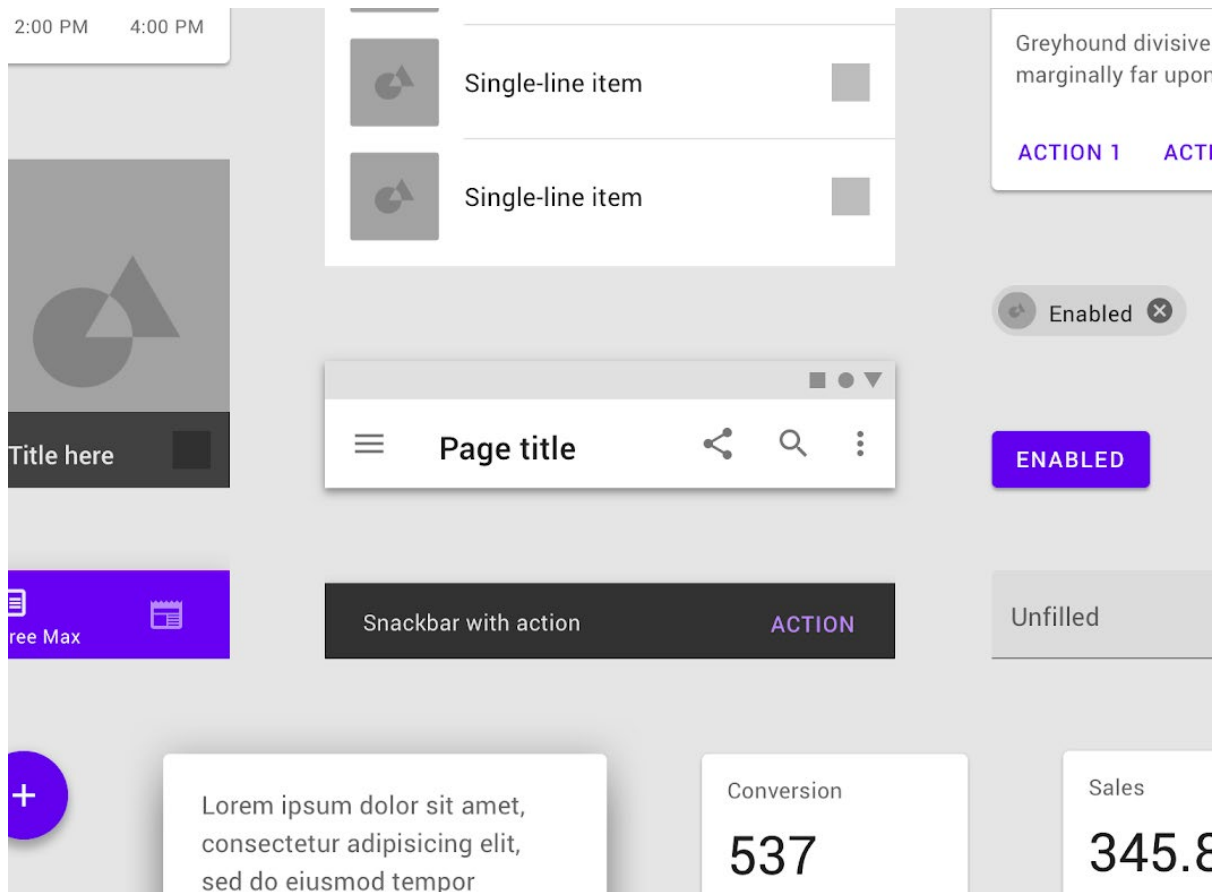
Design systems famosos ganharam até seus próprios nomes, o que reforça a expressão de “um produto servindo a outro produto”.

- Lightning Design System, da Salesforce.



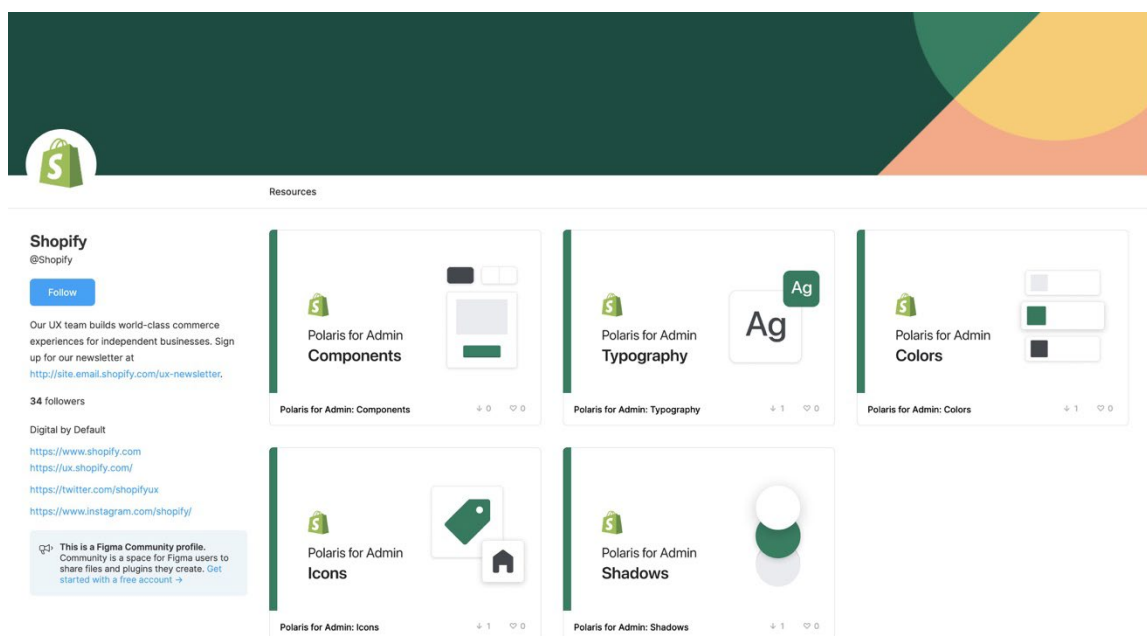
Fonte: Lightning Design System.

- Material Design do Google.



Fonte: Material Design.

- Polaris, do Shopify



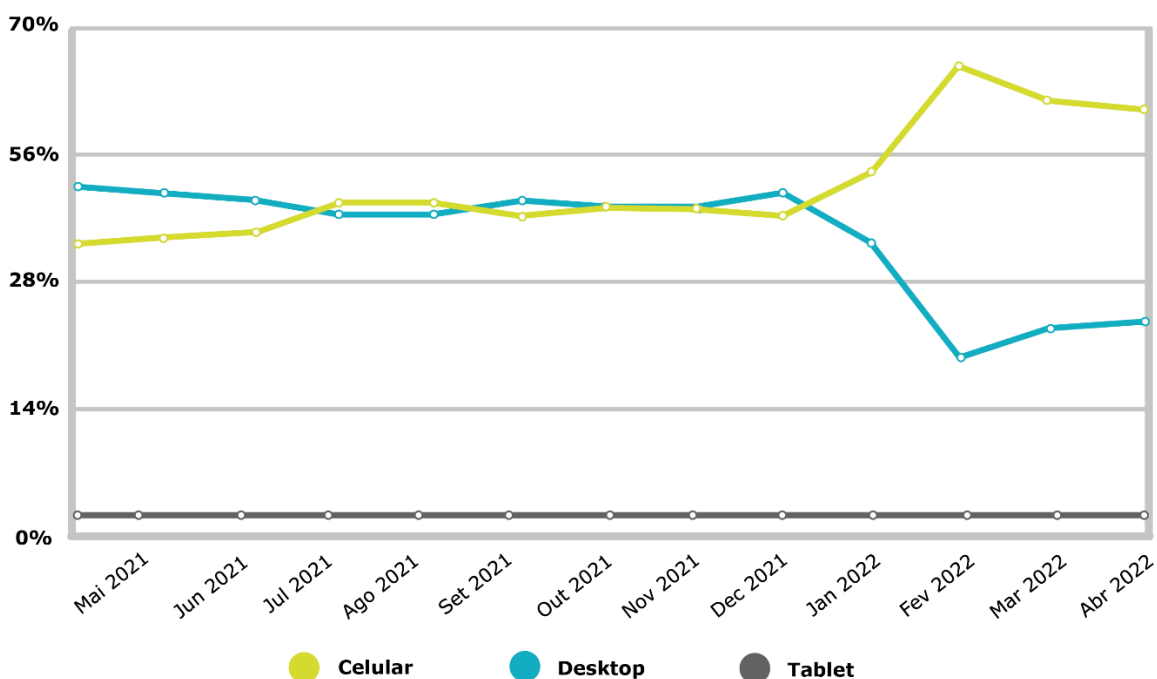
Fonte: Polaris.

TEMA 5 – DESIGN RESPONSIVO

Há alguns anos o termo “*mobile first*”, que significa prioridade para o acesso móvel, começou a ser usado nos projetos digitais. O motivo é simples: a maior parte do tráfego de diversas plataformas on-line começou a vir de dispositivos móveis – sendo quase todos smartphones.

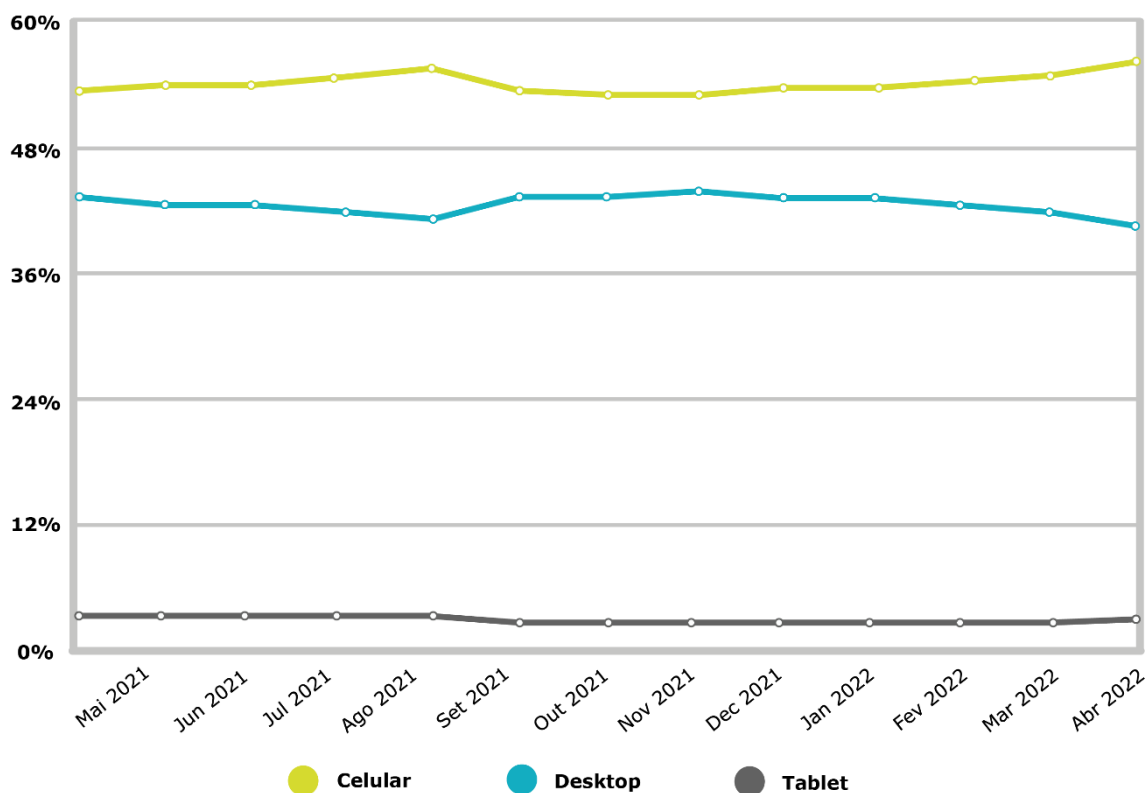
No Brasil, a escalada do acesso móvel veio com a liberação da banda 3G, em 2007. De lá pra cá, os aparelhos vêm se tornando cada vez mais velozes, com telas de maior resolução e maior velocidade de conexão. Frequentemente, é mais simples executar tarefas e acessar informações ou serviços em um celular do que usando o computador. Prova disso são os gráficos abaixo. No Brasil (figura 1), o celular assumiu a dianteira neste ano, depois de ter passado um ano praticamente empatado. E, observando os dados mundiais (figura 2), fica fácil inferir que esta dianteira do celular vai se manter. Ou seja, é mandatório que os projetos digitais, salvo em casos muito específicos de aplicações *desktop*, precisam levar em conta como será a interface nos celulares antes mesmo de pensar no desktop.

Figura 1 – Participação do uso de celulares, desktops e tablets no Brasil nos últimos 12 meses



Fonte: Statcounter, 2022.

Figura 2 – Participação mundial do uso de celulares, desktops e tablets nos últimos 12 meses



Fonte: Statcounter, 2022.

5.1 Desafios da responsividade

Os modelos de smartphone se proliferam a cada dia. A corrida tecnológica entre as marcas acarreta lançamentos frequentes, resoluções crescentes, capacidades cada vez maiores. Ou seja: ao desenhar as interfaces para celulares, o designer precisa sempre ter em mente algumas importantes questões:

- Quantidade de telas e resoluções disponíveis: diferente do desktop, que possui poucas resoluções de tela, representando a grande maioria dos acessos, nos celulares, cada marca possui dezenas de modelos em operação, cada uma com um tamanho e resolução de tela;
- Variações de sistemas operacionais, capacidade de processamento e memória de armazenamento: muitos celulares, hoje em dia, têm capacidades de armazenamento e processamento maiores que muitos computadores pessoais.

Na tabela abaixo, é possível ter uma amostra da complexidade do design para dispositivos móveis. São as variações que um software de testes que utilizo, o LambdaTest, tem à disposição:

MOBILE - ANDROID	SAMSUNG	MOTOROLA	XIAOMI	OUTROS
	21	2	4	10

MOBILE -APPLE	IPHONE	IPAD
iOS 14 versões +	20	29

Além das variações de telas, os projetos ainda podem optar pela forma de uso dos aparelhos: via websites, aplicativos móveis ou, ainda, PWAs (*progressive web apps*), que são uma mescla de aplicativo com website.

Saiba mais

Neste site, vocês poderão compreender um pouco melhor as diferenças entre cada tecnologia: <<https://umbo.net.br/blog/pwa-vs-apps-nativos-como-escolher-sua-tecnologia/>>. Acesso em: 23 jun. 2022.

5.2 Benefícios do design responsivo

- Base de visitas: quanto mais compatível, mais acessível.
- SEO: Google considera fortemente o item “*mobile-friendly*” para ranquear um site em sua busca. Do contrário, seus projetos dependerão demais de mídia paga para conseguir acessos.

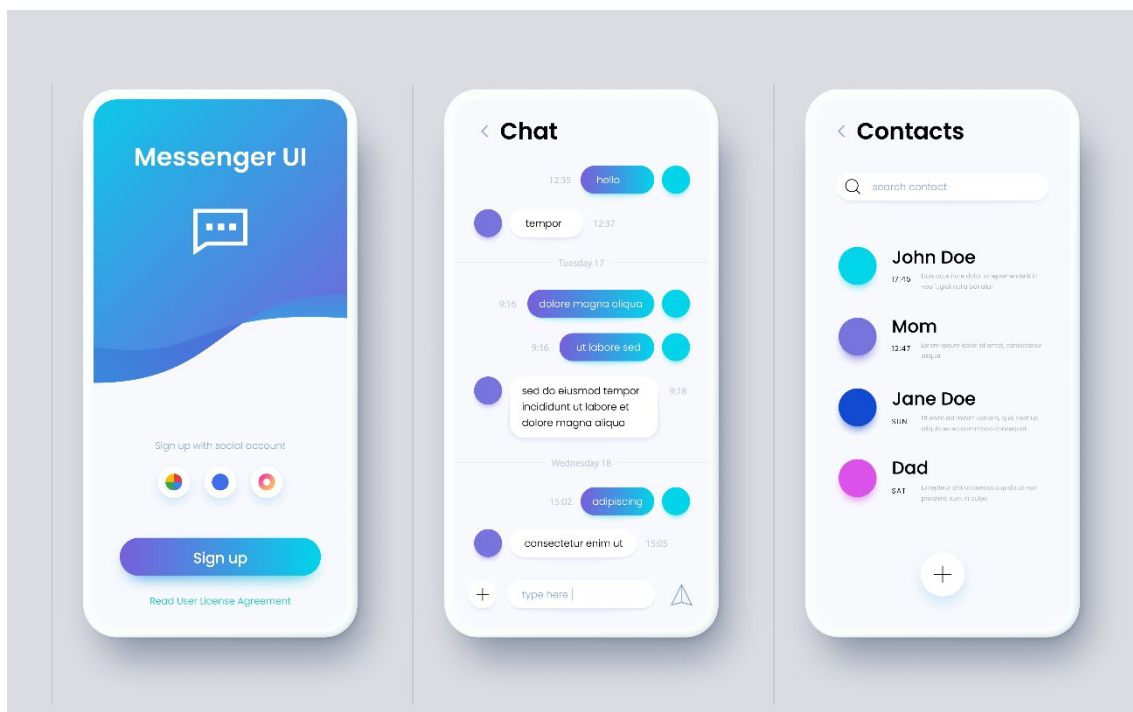
5.3 Princípios do design responsivo

- Comandos sempre acessíveis: as telas precisam ter seus comandos sempre ao alcance dos olhos e dos dedos. Observem a barra de URL (onde se coloca o endereço do site) no *browser* Safari, em um iPhone. Ela fica abaixo do conteúdo, ao alcance de onde geralmente está o polegar, dedo mais usado para operar os celulares.



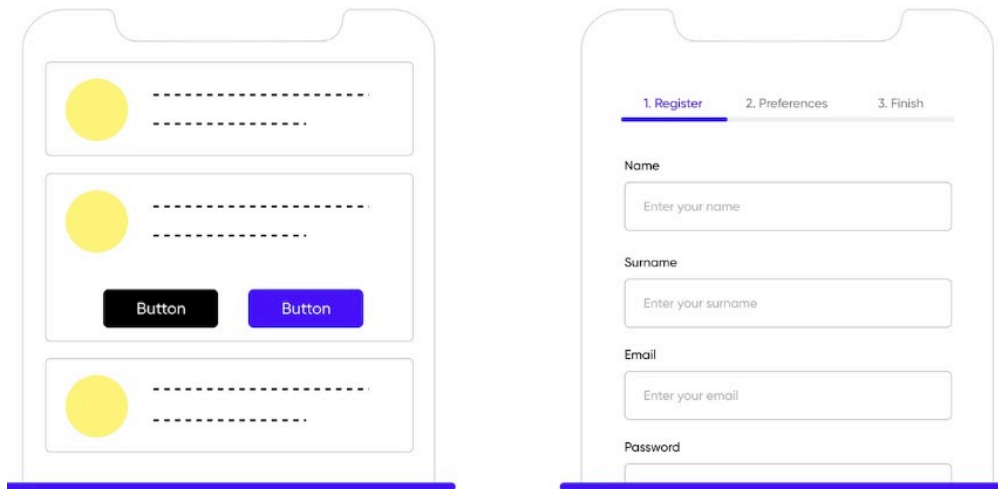
Crédito: Sbo.ow-j/Adobe Stock.

- Legibilidade: textos com tamanho suficiente para leitura e contraste suficiente com a cor de fundo.



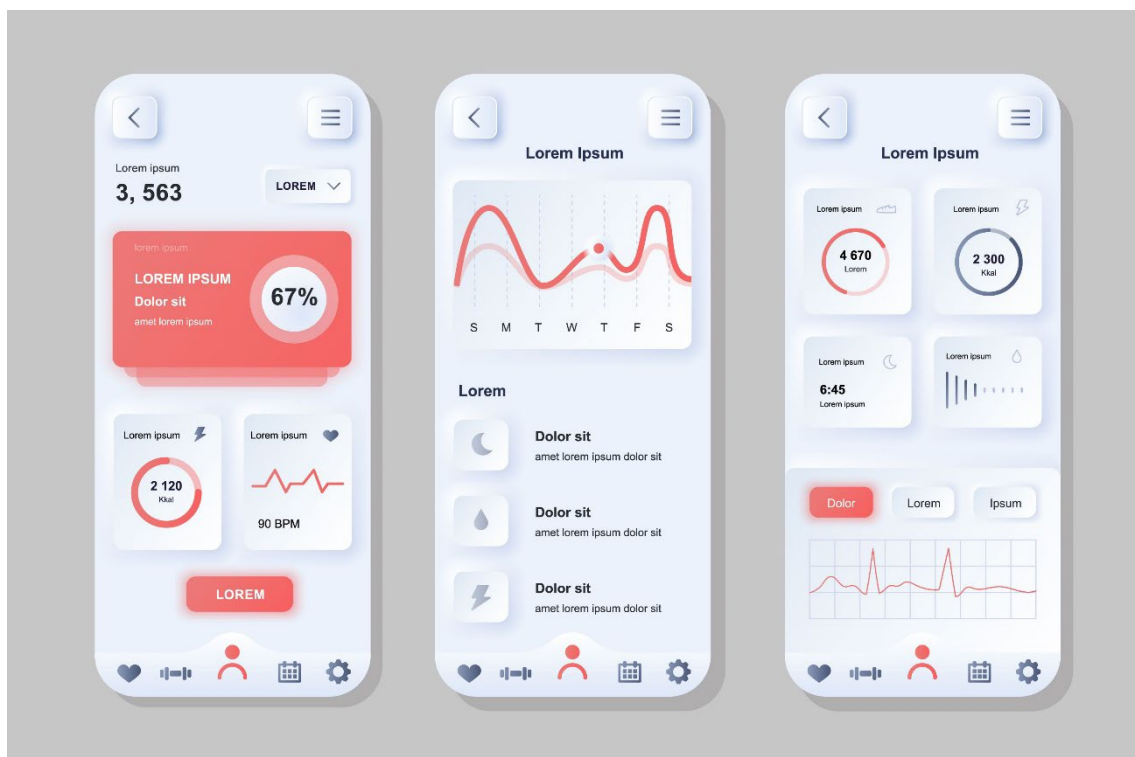
Crédito: SpicyTruffel/Adobe Stock

- Evite o *scroll* longo: o agrupamento de informações e a utilização de “abas” de navegação são importantes recursos para que o usuário não se perca em uma página muito alta/longa.



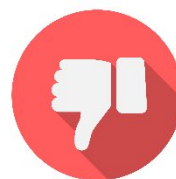
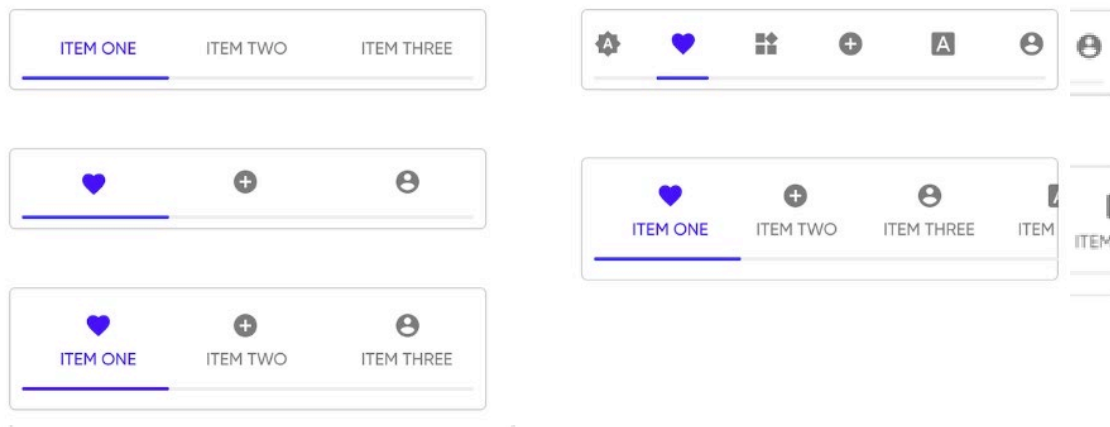
Fonte: Fleck, 2020.

- Dê espaço para os olhos e, principalmente, para os dedos: áreas de toque, como botões e links/títulos, não podem ficar “espremidos” entre outros elementos. Do contrário, um toque pode acionar mais de um gatilho ao mesmo tempo, ou, ainda, deixar o usuário desconfortável para usar a interface.



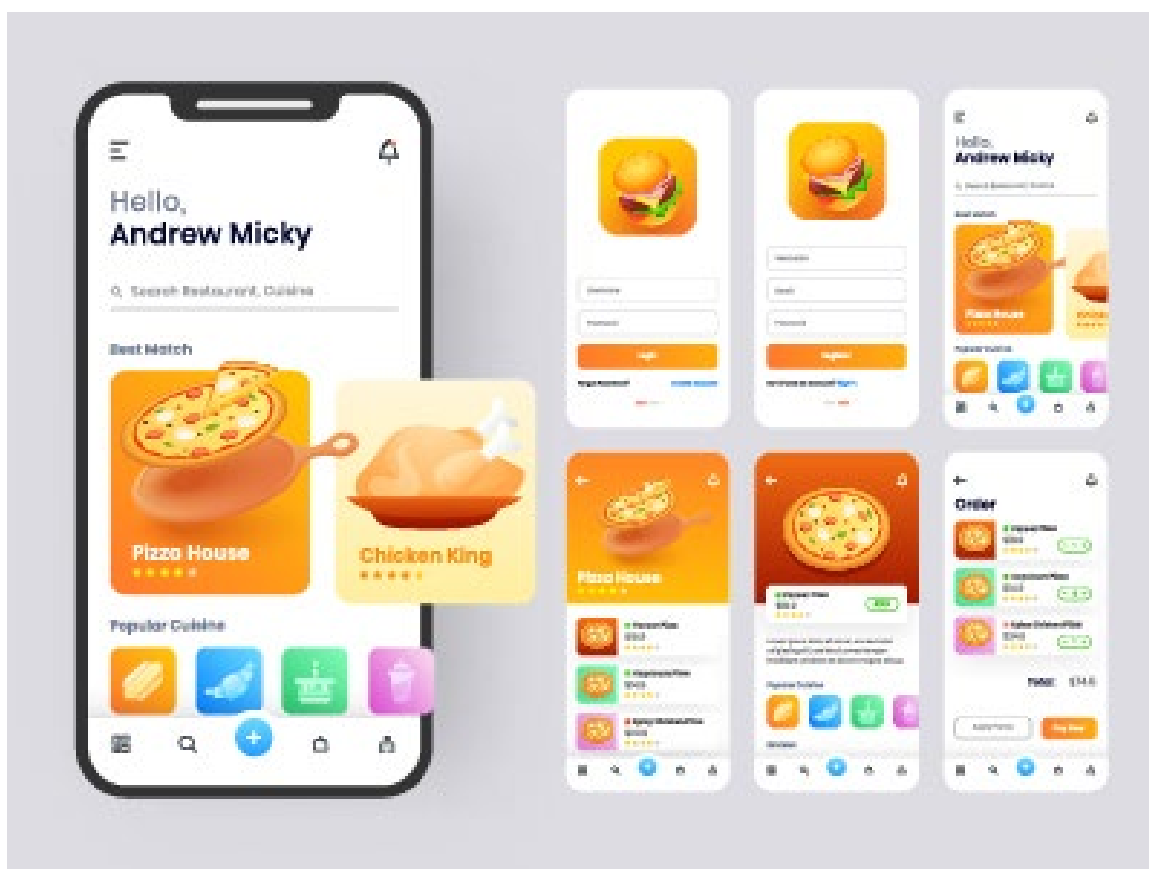
Crédito: Alexdndz/Shutterstock.

- Mantenha as barras de navegação com espaço e clareza: não acumule muitos ícones, nem use rolagens horizontais ou itens ocultos na tela.



Crédito: Noindex/Shutterstock.

- Use barras de navegação familiares: nestes e em muitos outros casos, o melhor é usar o padrão e não ser inovador ou extremamente criativo. Quanto mais rapidamente reconhecida, mais usável é a barra.



Crédito: Abdul Qaiyoom/Adobe Stock.

REFERÊNCIAS

CAO, J. Wireframing & Prototyping: The Past, Present, and Future. **Designmodo**, 5 fev. 2015. Disponível em: <<https://designmodo.com/wireframing-prototyping/>>. Acesso em: 23 jun. 2022.

DESIGN System. **Wikipedia**. Disponível em: <https://en.wikipedia.org/wiki/Design_system>. Acesso em: 23 jun. 2022.

FLECK, R. 8 mobile UX best practices every designer should consider. **Dribbble**, 12 mai. 2020. Disponível em: <<https://dribbble.com/stories/2020/05/12/8-mobile-design-tips>>. Acesso em: 23 jun. 2022.

PORTMAN, H. A Project Manager's Guide to 42 Agile Methodologies. **The Digital Project Manager**. Disponível em: <<https://thedigitalprojectmanager.com/agile-methodologies/>>. Acesso em: 23 jun. 2022.

PREMIER AGILE. **Top 7 Agile Frameworks**. Disponível em: <<https://premieragile.com/types-of-agile-frameworks/>>. Acesso em: 23 jun. 2022.

SILVA JÚNIOR, C. O Conceito de The Heart Of Agile. **LinkedIn**, 29 mai. 2020. Disponível em: <<https://www.linkedin.com/pulse/o-conceito-do-heart-agile-carlos-silva-j%C3%BAnior-/?originalSubdomain=pt>>. Acesso em: 23 jun. 2022.

STATCOUNTER. **Desktop vs Mobile vs Tablet Market Share Brazil**. 2022. Disponível em: <<https://gs.statcounter.com/platform-market-share/desktop-mobile-tablet/brazil>>. Acesso em: 23 jun. 2022.

_____. **Desktop vs Mobile vs Tablet Market Share Worldwide**. 2022. Disponível em: <<https://gs.statcounter.com/platform-market-share/desktop-mobile-tablet>>. Acesso em: 23 jun. 2022.