**Diferença entre aplicativo nativo, site e progressive web (PWA)**

Os aplicativos são ferramentas essenciais para empresas que desejam se conectar com seus clientes de forma mais direta, eficiente e personalizada. Portanto, ao considerar a criação de uma solução digital, como um site ou aplicativo, é crucial entender as diferenças entre as opções disponíveis. Cada um tem seus pontos fortes e fracos, e escolher a melhor solução depende das necessidades específicas do seu negócio.

**O que é um aplicativo?**

Um aplicativo ou “app” é um programa criado para ajudar a realizar tarefas específicas no seu dispositivo. Como o GPS que te ajuda a encontrar o caminho mais rápido, o app de e-mail, onde você confere suas mensagens, são exemplos clássicos de um aplicativo.

Foram desenvolvidos para resolver um problema específico ou tornar algo mais fácil no seu dia a dia.

Agora, vamos falar um pouco sobre o nome aplicativo. A palavra vem do latim applicare, que significa “colocar sobre” ou “aproximar”.

No mundo da tecnologia, isso faz sentido, já que os aplicativos são ferramentas que “aplicamos” para realizar certas tarefas no nosso computador ou celular.

**Para que serve um aplicativo?**

O grande objetivo dos aplicativos é automatizar tarefas, tornar as coisas mais rápidas e acessíveis, ou seja, os aplicativos existem para facilitar a nossa vida.

**Como funcionam os aplicativos**

Quando usamos um app, parece que tudo acontece magicamente, né?

Mas por trás disso, há muitos processos acontecendo. Cada aplicativo conversa com o seu celular e usa as ferramentas que o próprio aparelho tem para realizar tarefas.

Por exemplo, se você usa um app de mapas, ele vai precisar acessar o GPS do seu celular. O GPS (ou Global Positioning System) é a tecnologia que permite que o celular saiba onde você está no mundo, em tempo real. Então, o app usa essa informação para te mostrar o caminho certo ou te dar dicas de lugares por perto. Outro exemplo é quando você usa a câmera no Instagram ou no WhatsApp. O aplicativo se conecta à câmera do seu celular e permite que você tire fotos, grave vídeos e compartilhe com seus amigos(as). Isso tudo acontece porque o aplicativo sabe como “falar” com os diferentes recursos do celular.

**Tipos de aplicativos**

Os três tipos mais comuns são: nativo, web (ou PWA), e híbrido.

Um aplicativo nativo é aquele criado especificamente para um sistema operacional, como Android ou iOS. Ele é desenvolvido usando linguagens e ferramentas próprias desses sistemas, Os apps que você baixa diretamente na Play Store ou App Store, como jogos e redes sociais, geralmente são nativos.

Já o aplicativo web, também conhecido como PWA (Progressive Web App), é acessado diretamente pelo navegador, sem precisar ser baixado.

Esses aplicativos funcionam como sites, mas com a vantagem de terem recursos que lembram um app tradicional, como notificações e até a possibilidade de funcionar offline.

O aplicativo híbrido é desenvolvido com uma única base de código, que pode ser utilizada tanto para Android quanto para iOS, mas sem depender de tecnologias web como o PWA.

**Aplicativo nativo**

Esses são os apps feitos especialmente para uma única plataforma, ou seja, a pessoa desenvolvedora cria o app pensando nas necessidades de um sistema específico, como Android, iOS, ou até mesmo para uma TV ou relógio inteligente.

O aplicativo é construído sob medida para usar todas as funções e capacidades de um sistema operacional específico.

Então, no caso do iOS, as pessoas desenvolvedoras usam uma linguagem chamada Swift (ou Objective-C), que é exclusiva para os dispositivos da Apple, como iPhones, iPads e até o Apple Watch. Do outro lado, temos o Android, onde as pessoas desenvolvedoras usam Kotlin (ou Java) para criar apps feitos especialmente para os milhões de dispositivos com esse sistema, desde smartphones até tablets e relógios inteligentes que rodam o Android Wear.

Esse tipo de desenvolvimento garante que o aplicativo possa interagir com o aparelho de forma bem fluida.

**Vantagens do aplicativo nativo**

*1. Integração*

Um grande benefício de criar aplicativos nativos para iOS e Android é que esses aplicativos podem interagir com outros aplicativos nativos. Por exemplo, você pode permitir que os usuários do seu aplicativo se conectem facilmente ao Facebook.

*2. Ferramentas e estruturas suportadas*

Ao desenvolver um aplicativo nativo, você utilizará uma variedade de ferramentas com suporte do desenvolvedor em conjunto com o sistema operacional relevante. Em comparação, os PWAs não têm as mesmas opções para simplificar o desenvolvimento e agilizar o processo geral.

*3. Monetização*

Seu aplicativo nativo pode integrar o processamento de pagamentos à loja de aplicativos do usuário, facilitando as compras. Se você deseja monetizar um PWA, precisa integrar seu próprio sistema de pagamento.

*4. Visibilidade na App Store*

Embora tanto aplicativos nativos quanto PWAs possam ser publicados em lojas de aplicativos, esse processo é muito mais complicado para PWAs. A App Store da Apple exige que os desenvolvedores criem um wrapper nativo que registre os recursos nativos do seu aplicativo para iOS. Você também precisa fornecer um comprovante válido de que é uma empresa legal e registrada.

*5. Segurança*

É mais fácil implementar recursos de segurança robustos, como autenticação de dois fatores, porque o aplicativo tem acesso às informações necessárias do dispositivo. PWAs precisam de sua própria certificação de segurança, enquanto aplicativos nativos podem incorporar certificados TLS para garantir que altos padrões de segurança sejam atendidos.

*6. Consumo de bateria*

Os aplicativos nativos são eficientes quando se trata de uso de bateria simplesmente porque são escritos na linguagem nativa do hardware.

**Desvantagens dos aplicativos nativos**

*1. Desenvolvimento complexo*

Desenvolver um aplicativo nativo pode ser um processo complexo, caro e demorado. E quando você começa do zero, pode levar meses — ou até anos — para colocar o produto em funcionamento.

*2. Manutenção contínua*

Com atualizações do sistema operacional em constante mudança e necessidades dos usuários em constante evolução, os aplicativos nativos exigem manutenção mais contínua para garantir que seu aplicativo seja seguro, livre de bugs e atenda melhor aos seus usuários.

*3. Não é multiplataforma*

Se você quiser criar aplicativos para usuários de Android e iOS, precisará desenvolvê-los duas vezes: em Java (para Android) e em Objective C (para iOS). Isso pode aumentar o tempo e o dinheiro necessários para o lançamento.

**Exemplos de aplicativo nativo**

Agora que já falamos sobre o que torna os aplicativos nativos especiais, vamos dar uma olhada em alguns exemplos de apps famosos que foram desenvolvidos dessa forma e que aproveitam muito bem os recursos dos dispositivos.

O WhatsApp é um ótimo exemplo de aplicativo nativo. Ele foi criado para se integrar profundamente ao sistema do seu celular, usando recursos como a câmera, o microfone e até o sistema de notificações de uma maneira muito direta. Além disso, o WhatsApp usa o sistema de mensagens push do próprio celular para garantir que as notificações cheguem imediatamente, o que seria muito mais complicado de fazer em um app que não fosse nativo.

Outro exemplo interessante é o Spotify. Ele foi desenvolvido para tirar proveito do sistema de áudio do seu dispositivo, integrando-se ao controle de volume e até a reprodução em segundo plano, algo essencial para um app de música. Se você já colocou uma playlist para tocar enquanto faz outra coisa no celular, como mandar uma mensagem ou navegar na web, isso só é possível porque o Spotify consegue interagir diretamente com o sistema operacional.

**Aplicativo web**

O aplicativo web, também conhecido como PWA (Progressive Web App), é uma solução que roda diretamente no navegador, como o Chrome ou Safari, sem a necessidade de ser baixado e instalado a partir de uma loja de aplicativos, como acontece com os apps nativos.

Apesar disso, os PWAs se comportam de forma muito parecida com aplicativos tradicionais: possuem ícone na tela inicial, funcionam em tela cheia e podem até enviar notificações. Eles são desenvolvidos com tecnologias da web, como HTML, CSS e JavaScript.

Outro destaque é sua capacidade de funcionar offline, desde que algumas informações tenham sido carregadas anteriormente. Isso é possível porque o PWA armazena dados temporários no aparelho, permitindo o uso de certas funcionalidades mesmo sem conexão com a internet.

Assim, os PWAs combinam praticidade, leveza e desempenho em uma solução acessível e moderna.

**Vantagens do aplicativo web**

*1. Velocidade de carregamento*

Os PWAs utilizam service workers para gerenciar solicitações, cache e armazenamento de dados do shell. Como resultado, o shell do aplicativo carrega muito mais rápido do que um aplicativo nativo. A velocidade de carregamento tem um grande impacto na retenção e no engajamento.

*2. Experiência do usuário*

Os usuários têm uma experiência unificada, com a mesma interface tanto no navegador quanto no aplicativo instalado no dispositivo. Isso elimina a necessidade de o usuário aprender mais de uma interface.

*3. Custo de desenvolvimento*

Construir um PWA que possa operar em diversas plataformas e sistemas operacionais é econômico e reduzirá drasticamente sua carga de trabalho inicial. Você também economizará tempo e dinheiro no desenvolvimento de recursos e atualizações específicos para cada sistema operacional.

*4. Visibilidade online*

Mencionamos anteriormente que aplicativos nativos são mais fáceis de colocar em lojas de aplicativos, mas os PWAs ainda têm maior visibilidade devido à natureza de sua construção. Como os PWAs são compostos por páginas da web que imitam aplicativos, os usuários podem encontrar seu aplicativo online — não apenas na loja de aplicativos. Os usuários podem compartilhar a URL de qualquer página do aplicativo com facilidade, aumentando as oportunidades de crescimento orgânico e engajamento.

**Desvantagens dos aplicativos web progressivos (PWAs)**

*1. Capacidade offline limitada*

Muitos aplicativos nativos podem ser executados mesmo quando o usuário não está conectado à internet. No entanto, os PWAs dependem da conexão com o navegador para serem executados e, portanto, exigem conectividade com a internet. Se o modo offline for importante para seus usuários, um PWA provavelmente não é a escolha certa.

*2. Menos conveniente*

Os PWAs têm funcionalidade limitada em alguns sistemas operacionais e rápido consumo de bateria, e

pode não ter acesso total a muitos recursos do dispositivo.

**Exemplos de aplicativo web**

Um dos melhores exemplos de aplicativo web é o Google Docs. Se você já precisou escrever um trabalho ou colaborar com colegas em tempo real, sabe que ele permite editar documentos diretamente no navegador, sem precisar baixar nada.

O mais interessante é que, mesmo sendo um app web, ele te dá quase todas as funcionalidades de um editor de texto tradicional, como salvar automaticamente tudo o que você escreve. Você pode acessar seus documentos de qualquer dispositivo com internet, o que facilita muito o trabalho colaborativo.

Outro exemplo é o Netflix, que, além de ser famoso pelo seu app de celular, funciona perfeitamente no navegador. Na versão web, você pode assistir a filmes e séries com a mesma qualidade que teria no app, sem precisar baixar nada.

Isso é ótimo quando você está usando um computador que não tem o app instalado ou até em smart TVs que usam o navegador para rodar a plataforma. A navegação continua intuitiva e os vídeos carregam rápido, o que mostra como o Netflix trabalha bem a experiência de quem acessa pelo navegador.

**Aplicativo híbrido**

Os aplicativos híbridos combinam características dos apps nativos e dos apps web, permitindo o uso de um único código para Android e iOS. Essa abordagem economiza tempo e esforço no desenvolvimento. Existem diferentes tecnologias híbridas, como **Flutter**, **React Native**, **Ionic** e **Kotlin Multiplatform (KMP)**.

O **Flutter**, criado pelo Google, “desenha” toda a interface do app do zero usando o canvas do dispositivo, o que garante visual padronizado entre plataformas. No entanto, não usa componentes nativos e depende de plugins para acessar funções como GPS ou câmera.

Já o **React Native**, do Facebook, utiliza componentes nativos, o que dá ao app uma aparência mais integrada ao sistema operacional. Também usa JavaScript e plugins mantidos pela comunidade.

O **Ionic** tem uma abordagem mais próxima da web, usando HTML, CSS e JavaScript em uma WebView. Ele é ideal para quem já tem experiência em desenvolvimento web.

O **Kotlin Multiplatform** permite compartilhar a lógica de negócios entre plataformas, mas mantém as interfaces e recursos específicos separados, dando flexibilidade para usar o que há de melhor em cada sistema.

Apesar das vantagens, apps híbridos dependem fortemente de plugins, que são mantidos por comunidades e podem variar em qualidade e atualização.

**Vantagens do aplicativo híbrido**

***1. Código único para múltiplas plataformas***  
Permite reutilizar uma única base de código para Android e iOS, reduzindo retrabalho e acelerando o desenvolvimento.

*2.* ***Redução de custos e tempo***  
Evita desenvolver dois apps separados, o que diminui os custos de produção e acelera a entrega do produto.

*3.* ***Facilidade de manutenção***  
Atualizações e correções podem ser feitas em um só lugar e refletidas em todos os sistemas, tornando a manutenção mais simples e eficiente.

*4.* ***Flexibilidade no desenvolvimento***  
Mesmo com código compartilhado, é possível personalizar partes específicas do app para cada sistema operacional, se necessário.

*5.* ***Mais rapidez para lançar melhorias***  
A centralização do código permite implementar novas funcionalidades e lançá-las rapidamente em todas as plataformas.

**Desvantagens aplicativo híbrido**

*1.* ***Dependência de plugins de terceiros***  
Recursos nativos (como câmera ou GPS) dependem de plugins, que muitas vezes são mantidos pela comunidade e podem não ser atualizados rapidamente.

*2.* ***Risco de incompatibilidade com atualizações de sistema***  
Mudanças em versões novas do Android ou iOS podem quebrar funcionalidades se os plugins usados não forem atualizados a tempo.

*3.* ***Diferenças sutis na experiência do usuário***  
Mesmo com otimizações, apps híbridos podem apresentar comportamentos ligeiramente diferentes dos nativos, especialmente em dispositivos variados.

*4.* ***Desempenho inferior em apps complexos***  
Para aplicações que exigem alta performance (como jogos ou apps com realidade aumentada), o desempenho pode ser inferior ao de apps nativos.

**Exemplos de aplicativo híbrido**

O Instagram, foi inicialmente desenvolvido como um app nativo, mas, à medida que cresceu e passou a incluir mais funcionalidades e versões para várias plataformas, adotou parte da abordagem híbrida, especialmente com o React Native.

O interessante aqui é que, mesmo sendo híbrido, o Instagram mantém uma navegação fluida e a integração com recursos nativos do celular, como a câmera e o sistema de notificações. Essa combinação de tecnologias permite que a equipe de desenvolvimento implemente novos recursos com rapidez, sem precisar reescrever tudo do zero para Android e iOS.

E o Microsoft Teams é outro exemplo de um app que utiliza a abordagem híbrida, especialmente para garantir que o app funcione bem em diferentes dispositivos e sistemas. O Teams precisa estar disponível tanto em computadores quanto em celulares, e a abordagem híbrida ajuda a garantir que a experiência seja parecida em todos os dispositivos, facilitando o uso para quem precisa alternar entre plataformas durante o dia.

**Referências:**

**1. Adjust. Native app vs progressive web app.**  
Adjust, 18 nov. 2022. Disponível em: <https://www.adjust.com/blog/native-app-vs-progressive-web-app/>. Acesso em: 18 ago. 2025.

**2. Sulivam Softwares. Progressive Web App (PWA): o que é, vantagens e desvantagens para seu negócio.**  
Sulivam, 17 jun. 2025. Disponível em: <https://sulivam.com.br/progressive-web-app-pwa/>. Acesso em: 18 ago. 2025.

**3. Inova e-Business. Entendendo as Diferentes Tipologias de Sites e Apps.**Inova e-Business, 15 mai. 2024. Disponível em: <https://www.inovaebiz.com.br/post/entendendo-as-diferentes-tipologias-de-sites-e-apps>. Acesso em: 18 ago. 2025.

**4. ALURA. Tipos de aplicativos: o que são apps nativos, web e híbridos?**  
Alura, 6 out. 2024. Disponível em: <https://www.alura.com.br/artigos/tipos-de-aplicativos?srsltid=AfmBOorwhkDQWgnRLOwM0p1dQ_KZ6qD6sqS-KR2hF1WbFl22jSmxx5GN>. Acesso em: 18 ago. 2025.