**DESENVOLVIMENTO DE UM APLICATIVO WEB PARA EXIBIÇÃO DE IMAGENS E CONTEÚDOS: ETAPAS E TECNOLOGIAS**

**Luis Henrique Pessoa**  
**Gabriel Tomazine**  
**Matheus Costa**

**Henrik Borsoti**

**São Paulo**

**2025**

**ETEC Professor Camargo Aranha**  
  
  
  
  
  
  
**DESENVOLVIMENTO DE UM APLICATIVO WEB PARA EXIBIÇÃO DE IMAGENS E CONTEÚDOS: ETAPAS E TECNOLOGIAS**

Trabalho de Desenvolvimento Web, da mateira Programação de Aplicativos Mobile, sob orientação da Professora Najara Gabriela**.**

**São Paulo**

**2025**

## **1. INTRODUÇÃO**

Hoje em dia, com a popularização dos smartphones, tablets e computadores, as pessoas acessam conteúdos online o tempo todo. Aplicativos web — ou seja, aqueles que funcionam direto do navegador, sem precisar instalar nada — têm ganhado cada vez mais espaço. Eles são usados para todo tipo de tarefa: estudar, comprar, se informar ou se entreter.

Entre os aplicativos mais comuns estão os que permitem visualizar imagens e outros tipos de conteúdo, como textos e vídeos. Seja para mostrar um portfólio de trabalhos, ensinar algo ou divulgar produtos, a criação desse tipo de app precisa seguir alguns passos importantes, desde a ideia inicial até o funcionamento completo.

Este trabalho mostra, de forma organizada, como é o processo de desenvolvimento de um aplicativo web que funcione bem em diferentes dispositivos e permita exibir imagens e exemplos de conteúdo.

## **2. OBJETIVO**

O objetivo aqui é apresentar os principais passos para desenvolver um aplicativo web moderno e responsivo, voltado para a exibição de imagens e outros tipos de conteúdo, como textos e vídeos. A ideia é entender desde o planejamento até o desenvolvimento técnico e a publicação do aplicativo, usando boas práticas e ferramentas atuais.

## **3. ETAPAS DO DESENVOLVIMENTO**

### **3.1 Planejamento e Requisitos**

Antes de começar a programar, é importante entender o que o aplicativo vai fazer. Nessa etapa, são definidos:

* Quem vai usar o aplicativo;
* Quais funções ele precisa ter (por exemplo: cadastrar imagens, montar galerias, exibir descrições);
* Quais requisitos extras precisam ser considerados (como segurança, velocidade, aparência, acessibilidade etc.);
* Desenhos iniciais da estrutura do sistema (fluxogramas, casos de uso etc.).

### **3.2 Escolha das Tecnologias**

Depois do planejamento, é hora de decidir quais ferramentas serão usadas. Algumas opções modernas incluem:

* **Frontend**: React.js, Vue.js ou Angular, que ajudam a criar interfaces rápidas e interativas;
* **Backend**: Node.js (JavaScript), Django (Python) ou Laravel (PHP), para cuidar da lógica do sistema e banco de dados;
* **Banco de dados**: Firebase (em tempo real), MongoDB (NoSQL) ou MySQL (relacional);
* **Hospedagem**: Vercel, Netlify, Firebase Hosting ou Heroku, que permitem colocar o app no ar com rapidez;
* Comunicação entre partes do sistema pode ser feita com **API REST** ou **GraphQL**.

### **3.3 Prototipagem e Design**

Antes de começar a programar de fato, é bom criar esboços (wireframes) e protótipos da interface, usando ferramentas como **Figma** ou **Adobe XD**. Isso ajuda a visualizar como o app vai funcionar e facilita ajustes antes do código.

Também é importante pensar em design responsivo — ou seja, o app precisa funcionar bem tanto no celular quanto no computador.

### **3.4 Desenvolvimento**

Com tudo planejado, começa o desenvolvimento. Ele pode ser dividido em partes, com entregas aos poucos (usando metodologias como **Scrum** ou **Kanban**). Algumas boas práticas incluem:

* Criar componentes reutilizáveis;
* Usar imagens em formatos leves como **WebP**;
* Implementar carregamento inteligente (**lazy loading**) para melhorar o desempenho.

### **3.5 Testes**

Os testes garantem que tudo está funcionando como deveria. Podem ser feitos com ferramentas como:

* **Jest** ou **Mocha** para testes de código;
* **Cypress** para testar o funcionamento do sistema como um todo;
* **Lighthouse** e **PageSpeed Insights** para medir desempenho e responsividade.

### **3.6 Publicação e Monitoramento**

Depois de tudo pronto e testado, é hora de colocar o app no ar. Ferramentas como **Vercel** ou **Firebase Hosting** facilitam isso. É importante também monitorar o uso com ferramentas como **Google Analytics** ou **Sentry**, que ajudam a identificar problemas e entender o comportamento dos usuários.

### **3.7 Manutenção**

Por fim, o trabalho não termina com a publicação. É importante corrigir erros que possam surgir e atualizar o app conforme o feedback dos usuários. Essa manutenção garante que o sistema continue útil e eficiente com o tempo.

## **4. CONSIDERAÇÕES FINAIS**

Desenvolver um aplicativo web para mostrar imagens e conteúdos exige planejamento, atenção aos detalhes e escolha de boas ferramentas. Usar tecnologias modernas, manter o foco na experiência do usuário e garantir que o app funcione bem em diferentes dispositivos são pontos essenciais para um bom resultado.

Seguindo etapas bem definidas — como análise de requisitos, prototipagem, desenvolvimento, testes e manutenção —, é possível criar um aplicativo funcional, bonito e fácil de usar.

## **5. REFERÊNCIAS BIBLIOGRÁFICAS**

ALMEIDA, M. E. B. de; MORAN, J. M.; VALENTE, J. A. *Tecnologias na educação: ensinando e aprendendo com as tecnologias*. São Paulo: Edições Loyola, 2011.

BEZERRA, E. *Desenvolvimento de aplicações web com HTML, CSS, JavaScript, jQuery, Bootstrap, AngularJS, PHP e MySQL*. 3. ed. São Paulo: Novatec, 2020.

GASPAR, H. R. *Engenharia de Software: uma abordagem baseada em projetos*. 2. ed. São Paulo: Pearson Education do Brasil, 2015.

LERMAN, J.; SHARP, J. *Desenvolvimento Web com HTML, CSS, JavaScript e jQuery*. São Paulo: Novatec, 2019.

PRESSMAN, R. S.; MAXIM, B. R. *Engenharia de software: uma abordagem profissional*. 8. ed. Porto Alegre: AMGH, 2016.

REACTJS. *Documentação oficial do React*. Disponível em: <https://react.dev/>. Acesso em: 20 maio 2025.

MDN WEB DOCS. *Image loading best practices*. Mozilla Developer Network. Disponível em: <https://developer.mozilla.org/en-US/docs/Web/Media/Formats/Image_types>. Acesso em: 20 maio 2025.

GOOGLE DEVELOPERS. *Web performance optimization*. Disponível em: [https://web.dev](https://web.dev/). Acesso em: 20 maio 2025.