



Boas Práticas de Mercado


Nesta aula, vamos explorar as Boas Práticas de Mercado no contexto de desenvolvimento de software, com foco em UX e Design Thinking. Nosso objetivo é compreender como práticas consolidadas, como versionamento de código, design responsivo, acessibilidade e manutenção de desempenho, são aplicadas para garantir a qualidade e a eficiência dos produtos digitais. Estas práticas são essenciais para criar soluções que não apenas atendam às necessidades técnicas, mas também proporcionem uma experiência de usuário otimizada, segura e acessível, alinhada às demandas do mercado atual.

Versionamento de Código

O versionamento de código é uma prática essencial na gestão e no desenvolvimento de software, permitindo o registro e controle das alterações feitas em um conjunto de arquivos ou repositórios ao longo do tempo. Essa prática é fundamental não apenas para o desenvolvimento web, mas para qualquer tipo de desenvolvimento de software. O versionamento possibilita a rastreabilidade das alterações, documentando quem fez cada modificação e em que data, o que é crucial para manter um histórico detalhado do projeto.

Ao versionar o código, você garante que todas as modificações sejam registradas, permitindo reverter para versões anteriores caso algo dê errado. Isso é especialmente útil quando uma alteração inesperada compromete o funcionamento do sistema. Com o versionamento, você pode facilmente identificar a origem do problema e restaurar uma versão estável.

Outro benefício é a colaboração eficiente. Em equipes de desenvolvimento, o versionamento de código permite que várias pessoas trabalhem simultaneamente

no mesmo projeto sem conflitos, pois cada modificação é registrada e pode ser integrada de forma controlada. Além disso, ele facilita a comunicação entre  membros da equipe, pois todos têm acesso ao histórico de alterações e podem compreender o contexto de cada mudança.


O controle de versões também é crucial para o lançamento de novas funcionalidades ou correções. Cada atualização significativa do sistema pode ser agrupada em uma nova versão, que é documentada e rastreada, garantindo que o processo de desenvolvimento seja organizado e transparente.

Por fim, o versionamento de código oferece segurança e facilita a recuperação de dados. Se uma versão do código apresentar problemas, é possível reverter para uma versão anterior que estava funcionando corretamente, garantindo a continuidade do desenvolvimento sem grandes prejuízos. Essa prática é indispensável para manter a qualidade e a integridade do software ao longo de seu ciclo de vida.

Boas Práticas no Desenvolvimento Web

No desenvolvimento web, a adoção de boas práticas é crucial para garantir a qualidade, eficiência e acessibilidade dos projetos. Uma das principais práticas é o design responsivo, que assegura que o site se adapte a diferentes tamanhos de tela, proporcionando uma experiência consistente em desktops, tablets e smartphones. Isso é fundamental para atender às necessidades dos usuários, independentemente do dispositivo que eles estejam utilizando.

Outra prática essencial é a acessibilidade. Garantir que o site seja acessível a pessoas com deficiência é uma responsabilidade de todo desenvolvedor. Isso inclui o uso de elementos que facilitam a navegação por leitores de tela, a inclusão de legendas em imagens e vídeos, e o cumprimento das diretrizes de acessibilidade. Essas medidas garantem que todos os usuários tenham acesso igualitário ao conteúdo do site.

A performance é outro aspecto vital. Um site deve ser otimizado para carregar rapidamente, mesmo em condições de alto tráfego. Isso envolve a redução  tempo de carregamento, minimização da largura de banda e otimização dos recursos utilizados. Em projetos de grande porte, como sites com milhões de acessos simultâneos, a performance pode ser um fator determinante para o sucesso ou fracasso da aplicação.

Manutenção e atualização também são práticas essenciais no desenvolvimento web. Manter o site atualizado com as últimas tecnologias e corrigir vulnerabilidades de segurança são tarefas contínuas que garantem a longevidade e a segurança da aplicação. Ferramentas e plugins devem ser constantemente revisados e atualizados para evitar brechas de segurança que possam comprometer o site.

Essas práticas, quando seguidas, não só melhoram a experiência do usuário, mas também facilitam a gestão e a evolução do projeto ao longo do tempo.

Conheça Frameworks Web

Frameworks são conjuntos de ferramentas, bibliotecas e padrões de codificação que facilitam o desenvolvimento de software, oferecendo uma estrutura básica para a criação de sites e aplicativos. Eles agilizam tarefas comuns, como manipulação de HTTP, gerenciamento de sessões e acesso a banco de dados, permitindo que os desenvolvedores se concentrem em aspectos mais complexos do projeto.

No contexto do desenvolvimento web, frameworks como React, Angular e Vue.js são amplamente utilizados para criar interfaces de usuário interativas e eficientes. Esses frameworks de JavaScript são essenciais para a construção de aplicações web modernas, oferecendo soluções robustas e escaláveis que simplificam o processo de desenvolvimento.

Para o front-end, o Bootstrap é um exemplo clássico de framework CSS, que facilita a criação de layouts responsivos e visualmente atraentes com pouco esforço. Além do Bootstrap, outros frameworks como Tailwind CSS e Materialize oferecem diferentes abordagens para a estilização de sites, permitindo aos desenvolvedores

escolherem a ferramenta que melhor se adapta às suas necessidades e ao projeto em questão.



O uso de frameworks não só acelera o desenvolvimento, mas também garante que as melhores práticas sejam seguidas, resultando em código mais organizado, reutilizável e fácil de manter. Ao adotar frameworks, os desenvolvedores podem se beneficiar de uma base sólida que suporta o crescimento e a evolução dos projetos, sem a necessidade de reinventar a roda a cada nova tarefa.

Portfólio

Um portfólio é uma ferramenta fundamental para qualquer desenvolvedor que deseja se destacar no mercado de trabalho. Ele serve como uma vitrine para suas habilidades, experiências e projetos realizados, permitindo que potenciais empregadores ou clientes vejam de forma clara e organizada o que você é capaz de fazer. Um portfólio bem elaborado e atualizado é essencial para mostrar seu progresso e conquistas na área de desenvolvimento.

Criar um portfólio pode ser simples e acessível. Plataformas como o GitHub Pages oferecem uma forma prática de hospedar um site de portfólio gratuitamente. Com o GitHub Pages, você pode criar e publicar um site usando HTML, CSS e JavaScript diretamente do seu repositório GitHub, tornando seu portfólio facilmente acessível para quem desejar.

O processo de criação envolve a configuração de um repositório, a personalização do site e a publicação do conteúdo. O portfólio deve destacar seus projetos mais relevantes, demonstrando suas habilidades técnicas e criativas. Além disso, é importante incluir uma seção sobre você, explicando sua trajetória profissional, interesses e objetivos.

Manter o portfólio atualizado é crucial. À medida que você adquire novas habilidades e completa novos projetos, essas atualizações devem ser refletidas em seu portfólio. Isso demonstra que você está continuamente aprendendo e se

desenvolvendo como profissional, o que é altamente valorizado no mercado de trabalho.



Um portfólio bem construído não apenas facilita a obtenção de oportunidades de emprego, mas também serve como uma plataforma para mostrar seu compromisso com a carreira e sua capacidade de entregar resultados.

Encerramento e Próximos Passos

Concluindo nossa jornada sobre boas práticas de mercado no desenvolvimento de software, é importante refletir sobre os próximos passos e áreas que podem ser exploradas para aprofundar ainda mais seu conhecimento. O estudo de frameworks front-end, como React, Angular e Vue.js, é uma excelente maneira de expandir suas habilidades e se preparar para os desafios do desenvolvimento web moderno.

Além dos frameworks, explorar bibliotecas e ferramentas adicionais pode aumentar sua produtividade e eficiência. Existem inúmeras bibliotecas para JavaScript, CSS e HTML que podem ser integradas aos seus projetos, oferecendo funcionalidades extras e facilitando tarefas repetitivas. Ferramentas de desenvolvimento, como plugins para IDEs, também podem otimizar seu fluxo de trabalho.

Outra área crucial a ser explorada é a integração com bancos de dados. Aprender a conectar aplicações front-end a bancos de dados relacionais e não relacionais é fundamental para criar sistemas completos e robustos. A persistência de dados é uma habilidade essencial em praticamente todos os tipos de aplicações, e o conhecimento sobre diferentes bancos de dados, como MySQL, SQL Server e MongoDB, pode abrir portas para diversas oportunidades no mercado.

Por fim, lembre-se de que o aprendizado contínuo é a chave para o sucesso na carreira de desenvolvimento. Mantenha-se atualizado com as últimas tendências e tecnologias, participe de comunidades de desenvolvedores e esteja sempre disposto a explorar novas áreas e desafios. Assim, você estará preparado para trilhar um caminho sólido e promissor no desenvolvimento de software.

Conteúdo Bônus



Para uma leitura complementar gratuita e acessível, sugiro a Documentação do Git e a Documentação do Mozilla Developer Network (MDN):

1. Documentação do Git: O Git é uma ferramenta essencial para o versionamento de código, e sua documentação oficial é um excelente recurso para entender boas práticas de controle de versões.

2. Mozilla Developer Network (MDN): O MDN oferece uma vasta gama de recursos gratuitos sobre desenvolvimento web, incluindo boas práticas de acessibilidade, performance, e design responsivo. É uma leitura essencial para quem quer se aprofundar em boas práticas de desenvolvimento web.

Referência Bibliográfica

ASCENCIO, A. F. G.; CAMPOS, E. A. V. de. Fundamentos da programação de computadores. 3.ed. Pearson: 2012.

BRAGA, P. H. Teste de software. Pearson: 2016.

GALLOTTI, G. M. A. (Org.). Arquitetura de software. Pearson: 2017.

GALLOTTI, G. M. A. Qualidade de software. Pearson: 2015.

MEDEIROS, E. Desenvolvendo software com UML 2.0 definitivo. Pearson: 2004.

PFLEEGER, S. L. Engenharia de software: teoria e prática. 2.ed. Pearson: 2003.

SOMMERVILLE, I. Engenharia de software. 10.ed. Pearson: 2019.

Prática Integradora Desenvolvimento de Software

Atividade Prática 16 – Aplicando Boas Práticas no Desenvolvimento de Software



Objetivos:

- Aplicar conceitos de versionamento de código em um projeto de software.
- Implementar boas práticas de desenvolvimento web, incluindo design responsivo e acessibilidade.
- Utilizar frameworks e ferramentas para criar um projeto web funcional e de qualidade.


Materiais, Métodos e Ferramentas:

- **Materiais:** Computador com acesso à internet, editor de código (como VS Code), e navegador web.
- **Métodos:** Desenvolvimento prático de um projeto web com uso de boas práticas e frameworks.
- **Ferramentas:** Git para versionamento de código, Bootstrap para design responsivo, e um framework JavaScript (como React, Angular, ou Vue.js) para construção de interfaces interativas.

Atividade Prática

Primeiramente, leia atentamente o texto a seguir:

Você foi designado para criar um pequeno projeto web aplicando boas práticas de desenvolvimento de software. O projeto deve incluir uma página inicial que demonstre a aplicação das seguintes práticas:

- **Versionamento de Código:** Configure um repositório Git para o projeto, documente suas alterações e faça commits regulares. 
- **Design Responsivo e Acessibilidade:** Desenvolva a página utilizando Bootstrap para garantir um design responsivo e implemente práticas de acessibilidade.
- **Utilização de Frameworks:** Utilize um framework JavaScript para criar uma interface interativa e inclua um componente de sua escolha (por exemplo, um formulário ou uma lista de itens).

Agora, vamos praticar!

PASSO A PASSO DETALHADO DA ATIVIDADE:

- **Configuração do Projeto e Versionamento de Código:**
 - Crie um novo repositório Git no GitHub para o seu projeto.
 - Clone o repositório para seu computador e configure o ambiente de desenvolvimento.
 - Crie um arquivo **README.md** e documente o propósito do projeto.
 - Faça o primeiro commit com a configuração inicial do projeto.
- **Desenvolvimento da Página Inicial com Design Responsivo:**
 - Crie um arquivo `index.html` e inclua os links para o CSS e JS do Bootstrap.
 - Desenvolva a estrutura básica da página com um container, um grid de três colunas, e um menu de navegação.



- Utilize classes do Bootstrap para garantir que o layout seja responsivo em diferentes tamanhos de tela.
- Adicione elementos de acessibilidade, como atributos alt em imagens e roles apropriados em componentes interativos.
- **Implementação de um Framework JavaScript:**
 - Escolha um framework JavaScript (React, Angular, ou Vue.js) e configure-o no seu projeto.
 - Crie um componente interativo (por exemplo, um formulário de contato ou uma lista de produtos) utilizando o framework escolhido.
 - Certifique-se de que o componente esteja integrado corretamente com o layout da página e que siga boas práticas de desenvolvimento.
- **Testes e Versionamento Contínuo:**
 - Teste o projeto em diferentes dispositivos e navegadores para garantir que o design responsivo e a acessibilidade estejam funcionando corretamente.
 - Continue fazendo commits regulares com descrições detalhadas sobre as mudanças feitas.
 - Adicione um tag de versão para a conclusão da atividade prática.
- **Documentação e Finalização:**
 - Atualize o arquivo [README.md](#) com informações sobre as boas práticas aplicadas no projeto.

- Faça um commit final e envie o projeto para o GitHub.



- Prepare uma breve apresentação ou descrição do projeto, destacando como as boas práticas foram aplicadas.

Gabarito Esperado

- **Versionamento de Código:** o repositório Git deve estar corretamente configurado, com commits regulares e uma documentação clara sobre o projeto.
- **Design Responsivo e Acessibilidade:** a página deve estar responsiva e funcional em diferentes dispositivos, com práticas de acessibilidade corretamente implementadas.
- **Utilização de Frameworks:** o framework JavaScript deve ser integrado de forma eficaz, e o componente criado deve ser funcional e bem implementado.

Espero que essa atividade prática ajude a aplicar e consolidar as boas práticas de desenvolvimento de software!

Ir para exercício