



GitHub Pages



lá, alunos e alunas! Sejam bem-vindos de volta à nossa disciplina. Aqui que aprenderemos a publicar nosso site ou portfólio na web. Para isso, utilizaremos o GitHub Pages.

O que é o GitHub?

Antes de prosseguir, vamos fazer uma breve revisão. No módulo anterior, discutimos o que é o GitHub. Para quem não se lembra, o GitHub é uma plataforma baseada em Git e é uma das mais utilizadas para a gestão de códigos e versionamento.

GitHub Pages: Uma Visão Geral

O GitHub oferece um serviço gratuito chamado GitHub Pages, que nos permite publicar sites estáticos na internet. Neste módulo, vamos explorar esse serviço em detalhes. A estrutura da aula será a seguinte:

1. Criação de uma conta no GitHub (esta aula)
2. Criação de um repositório e upload do projeto
3. Entendendo o GitHub Pages
4. Publicando o site ou portfólio no GitHub Pages

Passo a Passo: Como Criar uma Conta no GitHub



1. Acesse o GitHub

Abra seu navegador de preferência e digite “GitHub” no buscador. O primeiro resultado provavelmente será o site oficial do GitHub. Clique nele para acessar a plataforma.

2. Escolha entre Sign-in e Sign-up

Você verá duas opções: “Sign-in” (entrar) e “Sign-up” (cadastrar). Se você já tem uma conta, faça o login. Caso contrário, clique em “Sign-up” para criar uma nova conta.

3. Preencha seus Dados

Uma mensagem de boas-vindas aparecerá, e você será solicitado a inserir seu e-mail. Após inserir o e-mail, você será direcionado para criar uma senha. Lembre-se, um bom profissional de TI não usa a mesma senha para todos os serviços. Utilize um gerenciador de senhas para criar e armazenar senhas fortes.


4. Escolha um Nome de Usuário

O próximo passo é escolher um nome de usuário. Esse nome é único e será a forma como você será reconhecido na plataforma. Escolha com cuidado!

5. Configurações Adicionais e Verificação

Você terá a opção de receber e-mails de atualizações e anúncios. Após fazer sua escolha, você precisará resolver um pequeno puzzle para confirmar que não é um robô. Em seguida, um código de verificação será enviado ao seu e-mail. Insira o código para finalizar o processo de criação da conta.

6. Personalização (Opcional)

O GitHub oferecerá uma série de perguntas para personalizar sua experiência. Você pode optar por pular essa etapa ou responder às perguntas para receber sugestões mais direcionadas. 

E pronto! Agora você tem uma conta no GitHub e está pronto para a próxima aula, onde aprenderemos a criar um repositório e fazer o upload do nosso projeto.

Criando e Configurando um Repositório no GitHub


Agora, vamos avançar para o próximo passo: criar um repositório para o nosso projeto na plataforma GitHub.

Visão Geral da Conta no GitHub

Depois de fazer login no GitHub, você será direcionado para a página inicial da sua conta. Aqui, você pode personalizar seu perfil, adicionar uma foto de avatar e até mesmo contar um pouco mais sobre você. Há também uma linha do tempo e um gráfico que mostrarão sua atividade no GitHub à medida que você contribui para diferentes projetos.

Como Criar um Repositório

1. Navegue até a Aba “Repositories”: clique na aba “Repositories” e depois clique em “New” (representado geralmente pelo símbolo ‘+’).
2. Nome e Descrição: dê um nome ao seu repositório. No meu caso, vou chamar de “Portfolio”. Adicionar uma descrição é opcional, mas é uma boa prática.

3. Visibilidade do Repositório: você pode escolher entre tornar seu repositório público ou privado. Repositórios públicos são acessíveis por qualquer pessoa  enquanto repositórios privados requerem permissão para acesso.

4. Arquivos Iniciais: o GitHub oferece a opção de criar um arquivo README, que é o arquivo exibido na página inicial do seu repositório. Este arquivo deve conter informações sobre o projeto, como sua origem, autor e como contribuir. Como nosso projeto já tem um README, vamos pular essa etapa.

5. Git Ignore e Licença: o GitHub também sugere a criação de um arquivo .gitignore, que é útil para ignorar arquivos que você não quer enviar para o repositório (como senhas ou arquivos de configuração). Além disso, você pode escolher uma licença para o seu projeto, que define como outras pessoas podem usar, modificar ou distribuir seu código.

6. Crie o Repositório: finalmente, clique em “Create repository” para criar seu repositório.

Enviando o Projeto para o GitHub

Agora que o repositório foi criado, é hora de enviar (ou “empurrar”, no jargão do Git) o projeto do seu computador para o GitHub. Para isso, usaremos o comando `git push`.

1. Configurando o Repositório Remoto: abra o terminal no seu editor de código (como o VS Code) e use o comando `git remote add origin [URL do seu repositório no GitHub]` para indicar onde o repositório remoto está localizado.

2. Empurrando para o GitHub: use o comando `git push origin main` para enviar os arquivos para o GitHub. Você será solicitado a autenticar sua conta novamente, o que é uma medida de segurança.

3. Verificação: depois de empurrar o projeto, você pode voltar ao GitHub e ver que todos os arquivos foram enviados com sucesso.



Parabéns! Você criou seu repositório e enviou seu projeto para o GitHub.

Introdução ao GitHub Pages

Já aprendemos como criar um repositório no GitHub e enviar nosso projeto para lá. Agora, vamos nos aprofundar em um serviço muito útil oferecido pelo GitHub: o GitHub Pages.

O que é GitHub Pages?

O GitHub Pages é um serviço de hospedagem de sites estáticos oferecido gratuitamente pelo GitHub. Mas o que significa “site estático”? São sites que não possuem back-end, ou seja, são compostos apenas por HTML, CSS e JavaScript no front-end. Esse serviço permite que você hospede esses sites diretamente de um repositório do GitHub, tornando o processo de colocar um site no ar muito mais simples e acessível.

Por que usar o GitHub Pages?

Hospedar um site geralmente requer um servidor, o que pode envolver custos mensais ou a necessidade de configurar um servidor em sua própria casa, o que pode ser complexo. O GitHub Pages surge como uma alternativa eficaz e largamente utilizada para hospedar sites estáticos de forma gratuita.

Documentação do GitHub Pages



Para quem deseja entender mais profundamente, a documentação oficial é um excelente recurso. Ela está disponível em diferentes idiomas, incluindo o português, embora a versão em inglês geralmente esteja mais atualizada. Você pode acessar a documentação em [indox.github.com](https://docs.github.com).

Exemplos de Uso

Um exemplo notável de site hospedado no GitHub Pages é o site do projeto Bootstrap, que estudamos na seção de Bootstrap deste curso. Isso demonstra o poder e a flexibilidade deste serviço.


Outras Opções: Jekyll

O GitHub também oferece a possibilidade de criar um site a partir de templates de Jekyll, um framework popular para sites estáticos. Isso pode ser feito diretamente através do site pages.github.com. Se você está começando um novo projeto, essa pode ser uma opção interessante a considerar.

Resumindo

O GitHub Pages é uma ferramenta poderosa para hospedar sites estáticos diretamente de um repositório GitHub. Ele é gratuito, fácil de usar e oferece várias opções para personalizar seu site.

Publicando Seu Site com GitHub Pages

Chegamos à quarta e última parte desta aula sobre GitHub Pages. Agora, vamos aprender como publicar nosso site na internet utilizando essa poderosa ferramenta  do GitHub.

Navegando pelo GitHub

Antes de começarmos, é importante notar que o GitHub é uma plataforma robusta com uma variedade de funcionalidades, como gestão de projetos, Wiki e Insights. Vale a pena explorar essas opções para tirar o máximo proveito da plataforma. Mas, por hoje, vamos nos concentrar em como publicar nosso site.

Passo a Passo para Publicar com GitHub Pages

1. **Acessar as Configurações do Repositório:** vá até o repositório onde está o seu projeto e clique na aba “Settings”.
2. **Localizar a Seção GitHub Pages:** dentro das configurações, você encontrará um menu à esquerda. Clique em “Pages” para acessar as opções de GitHub Pages.
3. **Escolher a Branch:** aqui, você terá a opção de publicar seu site a partir de diferentes branches. No nosso caso, como só temos a branch “main”, vamos selecioná-la.
4. **Definir o Diretório Raiz:** você também pode escolher publicar a partir de uma subpasta específica, mas vamos publicar a partir da raiz do projeto. Isso significa que o GitHub Pages procurará pelo arquivo `index.html` para exibir como página inicial.
5. **Salvar e Aguardar:** depois de fazer essas seleções, clique em “Save” e aguarde alguns minutos para que o site seja publicado.

URL Personalizada



Se você desejar, pode comprar um domínio personalizado e configurá-lo no GitHub Pages. Isso pode ser feito na seção “Custom domain” dentro das configurações de GitHub Pages. No Brasil, uma opção para comprar domínios é o [NIC.br](#), que é responsável pelos domínios que terminam com “.br”.

Comitê Gestor da Internet no Brasil ([CGI.br](#))

Para aqueles interessados em seguir as atualizações e regulamentações sobre a internet no Brasil, o [CGI.br](#) é uma entidade que vale a pena acompanhar. Eles são responsáveis por regular a internet no Brasil e fazem parte de diversas iniciativas, pesquisas e eventos.

Concluindo

E é isso! Seu site agora está publicado e acessível para qualquer pessoa na internet. Você pode compartilhar a URL fornecida pelo GitHub ou, se preferir, utilizar um domínio personalizado.

Este é o fim da nossa série sobre GitHub Pages. Agradeço a todos pela atenção e participação ao longo deste curso. Espero que vocês tenham aprendido algo valioso que possa ser aplicado em seus futuros projetos.

Conteúdo Bônus

Agora, gostaria de apresentar um conteúdo bônus para aqueles que desejam ir além do que foi ensinado em sala de aula.

- Visitação de Site



- Nome do Site: MDN Web Docs
- Autor: Mozilla
- Plataforma para Pesquisa: Basta pesquisar “MDN Web Docs” em seu mecanismo de busca favorito.

O MDN Web Docs é um recurso incrível para qualquer desenvolvedor web. Ele oferece documentação extensa, tutoriais e guias sobre quase todos os aspectos do desenvolvimento web.

Referências Bibliográficas

SOMMERVILLE, Ian. **Engenharia de software**. 10. ed. São Paulo: Pearson, 2018.

PFLEEGER, Shari Lawrence. **Engenharia de software: teoria e prática**. 2. ed. São Paulo: Pearson, 2004.

Ir para exercício