

Tópicos Essenciais à Programação

Prof. Me. Nisston Moraes Tavares de Melo
prof2279@iesp.edu.br



Conceito de Fork

- O **Fork** é a criação de uma cópia completa de um repositório no GitHub na conta do usuário. Isso permite ao usuário fazer alterações livremente sem afetar o repositório original.
- Objetivo:
 - Permitir que desenvolvedores trabalhem em um projeto independente do repositório original.
- Passos para **Forkar** um Repositório:
 - Navegue até o repositório que deseja forkar.
 - Clique no botão "Fork" no canto superior direito.
 - O repositório será copiado para a sua conta do GitHub.

Trabalhando com Forks

- Após forkar um repositório, você pode clonar o fork para o seu computador local, fazer alterações, e depois enviar (push) essas alterações de volta para o seu repositório no GitHub.
- Passos para Clonar o Repositório Forkado:
 - No GitHub, vá para a página do seu repositório forkado. • Clique no botão "Code" e copie a URL.
 - No seu terminal, execute ``git clone <URL>`` para clonar o repositório.

Conceito de Pull Request (PR)

- Um **Pull Request** (PR) é uma solicitação para mesclar suas alterações no repositório original. É uma maneira de contribuir com projetos de código aberto e colaborar com outros desenvolvedores.

- Objetivo:
 - Facilitar a revisão e integração de contribuições de vários desenvolvedores. •

Passos para Criar um Pull Request:

- Faça suas alterações no repositório forkado.
- Suba (push) suas alterações para o GitHub.
- Navegue até a página do repositório original.
- Clique em "New Pull Request".
- Compare suas alterações e clique em "Create Pull Request".
- Adicione um título e uma descrição, e envie.

Revisão de Pull Requests

- Após criar um Pull Request, outros colaboradores do projeto podem revisar seu código, fornecer feedback e sugerir mudanças.
- Passos na Revisão de um PR:

- Analisar o código proposto.
- Deixar comentários e solicitar mudanças, se necessário.
- Aprovar o PR se o código estiver correto e funcional.
- Mesclar o PR ao repositório principal.

Resolvendo Conflitos de Merge

- Conflitos de merge ocorrem quando duas ou mais alterações são feitas nas mesmas linhas de um arquivo e Git não consegue mesclar automaticamente.
- Como Resolver:
 - Identifique os conflitos no seu editor de texto.
 - Edite manualmente os arquivos conflitantes para resolver os conflitos.
 - Faça commit das alterações resolvidas.

- Continue com o processo de merge.

Boas Práticas para Forks e Pull Requests

- Adotar práticas que facilitam a colaboração e mantêm a integridade do código.
- Boas Práticas:
 - Sempre crie branches para novas funcionalidades ou correções de bugs antes de começar a trabalhar.
 - Escreva descrições claras e detalhadas para seus Pull Requests. • Revise cuidadosamente o código antes de solicitar um Pull Request. • Use commits atômicos, ou seja, commits que representam uma única mudança lógica.
 - Mantenha seu fork sincronizado com o repositório original.

Ferramentas e Recursos Úteis

- Recursos adicionais que podem auxiliar no gerenciamento de forks e pull requests.
- Ferramentas:
 - GitHub Desktop: Interface gráfica para gerenciar repositórios no GitHub.
 - GitKraken: Ferramenta gráfica para git com suporte a GitHub.
 - GitHub CLI: Interface de linha de comando para interagir com GitHub.

Exemplos Práticos

- Demonstrações ao vivo ou exemplos práticos para reforçar os conceitos discutidos.

- Exemplo 1: Forkar um repositório de exemplo, clonar localmente, fazer uma pequena alteração, e criar um Pull Request.
- Exemplo 2: Revisar um Pull Request, fazer comentários e sugerir mudanças.