Git e Github

Clarilene Silva

2023-12-06

# Oque é Git?

Git é um sistema de controle de versão distribuído, amplamente utilizado para o gerenciamento de código fonte durante o desenvolvimento de software. Ele foi criado por Linus Torvalds em 2005 para o desenvolvimento do kernel Linux, mas desde então tornou-se uma ferramenta essencial em muitos projetos de software.

# O que é GitHub?

O GitHub é uma plataforma de hospedagem de código-fonte baseada na web que utiliza o sistema de controle de versão Git. Ele oferece um ambiente colaborativo para desenvolvedores, permitindo que eles armazenem, compartilhem e colaborem em projetos de software.

# Como configurar bo git no PC?

Abra um terminal (no Windows, você pode usar o Git Bash que foi instalado junto com o Git) e execute os seguintes comandos, substituindo “Seu Nome” e “[seu@email.com](mailto:seu@email.com)” pelos seus dados reais:

git config –global user.name “Seu Nome” git config –global user.email “[seu@email.com](mailto:seu@email.com)”

# Conceitos basicos do GitHub

# Repositório

repositório no GitHub é um espaço onde os projetos de software são armazenados, gerenciados e compartilhados. Ele serve como um local centralizado para armazenar código-fonte, documentos, imagens, e outros arquivos relacionados a um projeto. Repositórios no GitHub são usados principalmente para controle de versão através do sistema Git, permitindo que várias pessoas colaborem em um projeto de maneira eficiente.

# Pull resquest

Um “pull request” (PR) no GitHub é uma solicitação feita por um colaborador de um repositório para que as alterações que ele fez em seu próprio fork desse repositório sejam consideradas para inclusão (mesclagem) no repositório original. Essa é uma prática comum em projetos de código aberto e em equipes de desenvolvimento colaborativo.

# Clone

O termo “clone” no contexto do GitHub refere-se à ação de criar uma cópia exata de um repositório, geralmente um repositório público, em sua própria conta do GitHub ou em sua máquina local. Isso permite que você tenha uma cópia do projeto para visualização, contribuição ou desenvolvimento sem afetar o repositório original.

Para fazer um clone de um repositório do GitHub, você pode usar o comando git clone. O formato geral do comando é o seguinte: git clone URL-do-Repositório git clone <https://github.com/nome-do-usuario/nome-do-repositorio.git>

Substitua “nome-do-usuario” pelo nome de usuário do proprietário do repositório e “nome-do-repositorio” pelo nome do repositório que você deseja clonar. Certifique-se de incluir “.git” no final da URL.

# Branch

branch (ramificação) é uma linha independente de desenvolvimento dentro de um repositório. Ela permite que os desenvolvedores trabalhem em funcionalidades, correções de bugs ou experimentos de maneira isolada, sem afetar diretamente o código na branch principal do projeto. O uso de branches facilita o desenvolvimento colaborativo e o gerenciamento de diferentes partes de um código-fonte.

# Commit

commit é uma operação que registra as alterações feitas nos arquivos do seu repositório. Um commit no Git é semelhante a tirar uma fotografia do estado atual dos seus arquivos e salvar essa imagem, incluindo uma mensagem que descreve as alterações realizadas.

# Pull

No GitHub, um “pull” se refere à ação de solicitar que as alterações feitas em um repositório remoto sejam incorporadas em seu repositório local. Esse processo é frequentemente chamado de “pull” porque você está puxando as alterações do repositório remoto para o seu ambiente local.

# Push

No contexto do GitHub, “push” se refere à ação de enviar as alterações que você fez no seu repositório local para o repositório remoto no GitHub. O comando git push é usado para enviar commits locais para o repositório remoto, mantendo as versões dos arquivos atualizadas em ambas as extremidades.

# Merge

O conceito de “merge” no GitHub refere-se à integração de alterações feitas em uma branch para outra branch. Geralmente, esse processo é usado para incorporar as alterações de uma branch de desenvolvimento (como uma feature branch) de volta à branch principal do projeto (como a branch “main” ou “master”).

# Status

O conceito de “status” no contexto do Git e GitHub refere-se ao estado atual dos arquivos no seu repositório em relação ao controle de versão. O comando git status é utilizado para verificar o estado do seu repositório local em relação ao último commit e às alterações pendentes.

# Checkout

conceito de “checkout” no GitHub e no Git refere-se à ação de mudar de branch ou de estado do repositório para um commit específico. O comando git checkout é usado para realizar essa operação, permitindo que você navegue entre diferentes branches, crie novas branches e até mesmo volte no tempo para um estado específico do histórico de commits.