**Prática 1**

1. Criação de Repositório no GitHub.

a) Abra seu navegador e acesse o site do GitHub (<https://github.com/>).

b) Faça login na sua conta do GitHub. Se ainda não tiver uma conta, crie uma seguindo as instruções fornecidas.

c) Após o login, clique no ícone "+" no canto superior direito da página inicial do GitHub.

d) No menu suspenso, escolha a opção "New repository".

e) Na página de criação de repositório, preencha o nome do repositório (ex.: MeuProjeto), escolha a visibilidade pública, selecione a opção "Add a README file" para inicializar o repositório com um arquivo README e selecione Node como template para adicionar o arquivo .gitignore. Role para baixo e clique no botão verde "Create repository".

f) Copie o link do repositório. Na parte superior da página, clique no botão verde "Code" e copie a URL fornecida.

2. Clonagem do Repositório Remoto.

a) Abra o Visual Studio Code (VSCode) no seu computador.

b) No canto superior esquerdo, clique em "File" (Arquivo) e selecione "Open Folder" (Abrir Pasta).

c) Escolha a pasta onde deseja clonar o repositório. Clique em "Select Folder" (Selecionar Pasta).

d) Abra o terminal no VSCode. Você pode fazer isso clicando em "View" (Visualizar) no menu superior e escolhendo "Terminal".

e) No terminal do VSCode, use o comando git para clonar a partir da URL do repositório remoto.

f) No canto superior esquerdo, clique em "File" (Arquivo) e selecione "Open Folder" (Abrir Pasta).

g) Escolha a pasta do repositório clonado. Clique em "Select Folder" (Selecionar Pasta).

h) Abra outro terminal no VSCode.

i) Execute o comando git para configurar localmente seu nome de usuário no Git.

j) Execute o comando git para configurar localmente seu e-mail de usuário no Git.

k) Execute o comando git config para verificar as configurações realizadas.

3. Adição e Commit de Alterações Locais.

a) Crie um arquivo "package.json" executando comando npm init -y

b) Execute o comando git para verificar o status do seu repositório local.

c) Execute o comando git para adicionar o arquivo "package.json" ao stage.

d) Execute novamente o comando git para verificar o status do seu repositório local.

e) Faça o commit das alterações com uma mensagem descritiva "Adicionando arquivo package.json".

4. Envio de Alterações Locais para Repositório Remoto.

a) Execute o comando git para verificar o status do seu repositório local.

b) Execute o comando git para listar as configurações dos repositórios remotos.

c) Execute o comando git para atualizar o repositório local com as mudanças na branch main do repositório remoto .

d) Execute o comando git para enviar as alterações na branch "main” para o repositório remoto.

e) Atualize a página do repositório remoto no navegador para verificar se as alterações foram enviadas.

5. Ramificação e Mesclagem Local.

a) Execute o comando git para criar uma nova branch chamada "develop".

b) Execute o comando git para mudar para a nova branch.

c) Execute o comando npm install nodemon --save-dev para instalar o pacote "nodemon" como dependência de desenvolvimento.

d) Execute o comando git para verificar o status do seu repositório local. A pasta node\_modules foi exibida como uma alteração no repositório local?

e) Execute o comando git para adicionar as alterações ao stage.

f) Faça o commit das alterações na branch "develop" com uma mensagem descritiva "Adicionando uma dependência no package.json".

g) Execute o comando git para enviar as alterações na branch "develop" para o repositório remoto.

h) Atualize a página do repositório remoto no navegador para verificar se as alterações foram enviadas.

i) Execute o comando git para mudar para a branch "main".

j) Execute o comando git para mesclar a branch "develop" na branch "main".

k) Execute o comando git para enviar as alterações na branch "main" para o repositório remoto.

l) Atualize a página do repositório remoto no navegador para verificar se as alterações foram enviadas.

6. Resolução de Conflitos na Mesclagem Local.

a) Execute o comando git para mudar para a branch "develop".

b) Faça uma alteração no arquivo "package.json" nos valores dos campos "name" e "version".

c) Faça o commit das alterações na branch "develop" com uma mensagem descritiva "Ajustando nome do projeto e versão no package.json".

d) Execute o comando git para mudar para a branch "main".

e) Faça uma alteração no arquivo "package.json" nos valores dos campos "name" e "version" de modo que sejam diferentes da branch "develop".

f) Faça o commit das alterações na branch "main" com uma mensagem descritiva "Ajustando nome do projeto e versão no package.json".

g) Execute o comando git para mesclar a branch "develop" na branch "main".

h) Abra o arquivo "package.json" no VSCode e localize as seções marcadas como "<<<<<<< HEAD", "=======" e ">>>>>>> develop".

i) Aceite as mudanças vindas da branch "develop" da clicando em "Accept incomming".

j) Execute o comando git para adicionar as alterações ao stage.

k) Faça o commit das alterações na branch "main" com uma mensagem descritiva "Corrigindo conflitos no package.json".

l) Execute o comando git para enviar as alterações na branch "main" para o repositório remoto.

7. Colaboração em Equipe.

a) Vá para a página inicial do seu repositório no GitHub.

b) Clique em "Settings" (Configurações) e, em seguida, em "Manage access" (Gerenciar acesso).

c) Convide um colega para colaborar, fornecendo o nome de usuário do GitHub ou o endereço de e-mail associado à conta dele. Conceda acesso de escrita ao seu repositório.

d) Acesse seu e-mail para aceitar o convite enviado pelo GitHub.

e) No menu principal do VSCode, clique em "File" (Arquivo) e selecione "Open Folder" (Abrir Pasta).

f) Escolha a pasta onde o repositório do seu colega será clonado. Clique em "Select Folder" (Selecionar Pasta).

g) No terminal do VSCode, execute o comando git para clonar a partir da URL do repositório remoto do colega.

h) No menu principal do VSCode, clique em "File" (Arquivo) e selecione "Open Folder" (Abrir Pasta).

i) Escolha a pasta do repositório clonado. Clique em "Select Folder" (Selecionar Pasta).

j) No terminal do VSCode, execute o comando git para mudar para a branch "develop".

k) Crie um arquivo index.js contendo a instrução console.log("Olá").

l) Execute o comando git para verificar o status do seu repositório local.

m) Execute o comando git para adicionar as alterações ao stage.

n) Faça o commit das alterações na branch "develop" com uma mensagem descritiva "Adicionando arquivo index.js".

o) Execute o comando git para enviar as alterações na branch "develop" para o repositório remoto do seu colega.

p) Vá para a página inicial do repositório do seu colega no GitHub e selecione a aba "Pull Request". Clique no botão "New Pull Request".

q) Selecione a branch "main" como base e a branch "develop" como comparação. Clique no botão "Create pull resquest".

r) Informe um título "Aplicar alterações da develop na main" para o pull request. Clique no botão "Create pull resquest".

s) Vá para a página inicial do seu repositório no GitHub e selecione a aba "Pull Request". Clique no título do pull request enviado.

t) Se não houver conflitos no pull request, clique no botão "Merge pull request” e depois no botão "Confirm merge”.

u) Execute o comando git para atualizar seu repositório local com as mudanças na branch main do GitHub.