

Tutorial de Fluxo de Trabalho Git para o Projeto IonicNews

Este documento descreve o fluxo de trabalho Git e os comandos essenciais a serem utilizados pela equipe do projeto IonicNews.

1. Estratégia de Branches

Adotaremos um modelo simplificado baseado no Gitflow:

- **`main`**: Branch principal. Contém o código de produção estável (ou a versão mais estável para entrega). Ninguém faz push direto para **`main`**. Alterações entram via Pull Requests a partir do branch **`develop`**.

- **`develop`**: Branch de desenvolvimento. Contém as últimas funcionalidades desenvolvidas e integradas. É o branch base para criar novas **`feature branches`**.

- **`feature/<nome-da-feature>`**: Branches criados a partir de **`develop`** para desenvolver novas funcionalidades. Ex: **`feature/currency-conversion`**, **`feature/history-page`**. Após a conclusão, são mesclados de volta em **`develop`** via Pull Request.

- **`fix/<nome-da-correcao>`**: Branches criados a partir de **`develop`** (ou **`main`** em casos críticos) para corrigir bugs. Ex: **`fix/api-error-handling`**. Mesclados de volta ao branch de origem via Pull Request.

2. Configuração Inicial

1- Clonar o repositório (apenas uma vez):

```
```bash
git clone
https://github.com/renanmiranda99/IonicNews.git
cd IonicNews
```

#### 2- Configure a API Key:

Copie sua chave da ExchangeRate-API para os arquivos:

- \* **`ionicnewsapp/currency-converter/src/environments/environment.ts`**
- \* **`ionicnewsapp/currency-converter/src/environments/environment.prod.ts`**

Exemplo (**`environment.ts`**):

```
```typescript
export const environment = {
  production: false,
  exchangeRateApiKey: 'SUA_CHAVE_API_AQUI'
};
```
```

**3- Navegue até a pasta do aplicativo Ionic e instale as dependências:**

```
```bash
cd ionicnewsapp/currency-converter
npm install
```
```

**4- Execute o aplicativo no navegador (para desenvolvimento):**

```
```bash
ionic serve
```
```

O aplicativo estará disponível em `http://localhost:8100`.