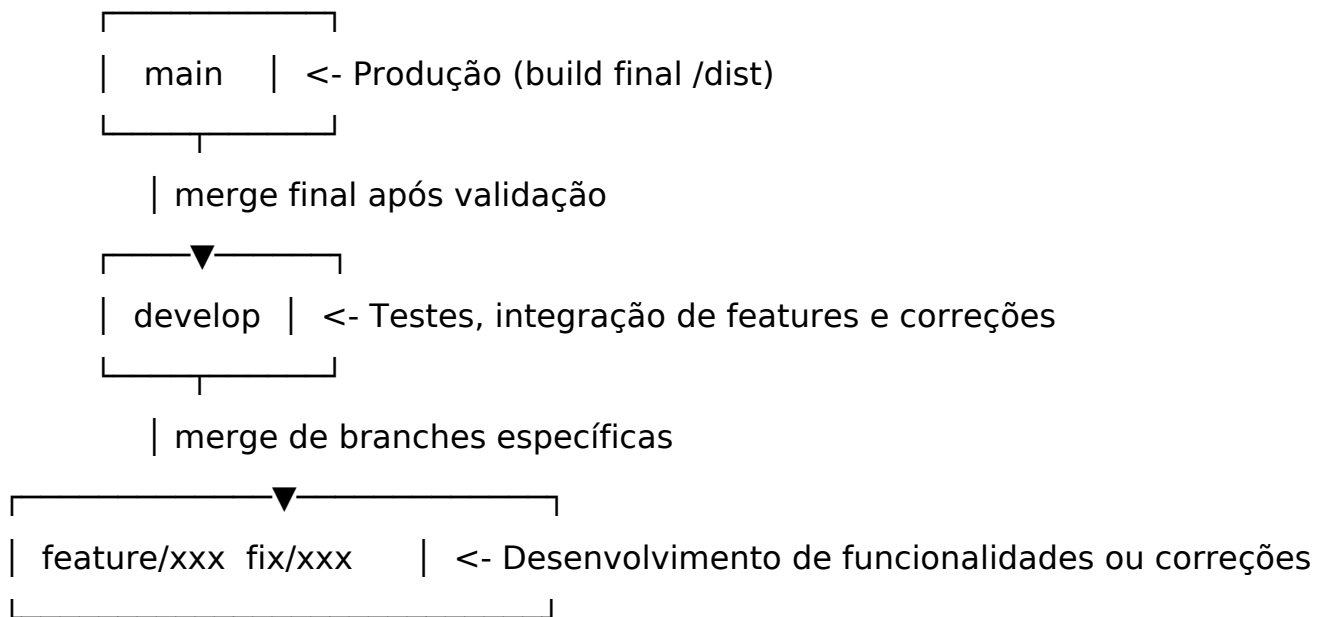


## Guia Git + Deploy no Windows para seu projeto

### Fluxo de Branches



### Comandos Git por cenário

#### Criar branch de feature ou correção

```
git checkout -b feature/nova-funcionalidade
```

# ou

```
git checkout -b fix/correcao
```

#### Commit e push da branch

```
git add .
```

```
git commit -m "Descrição da alteração"
```

```
git push origin feature/nova-funcionalidade
```

#### Integrar feature/correção ao develop

```
git checkout develop
```

```
git pull origin develop # garantir que está atualizado
```

```
git merge feature/nova-funcionalidade
```

```
git push origin develop
```

Validar build local

npm install

npm run build

# testar a versão no /dist

Merge para main (produção)

git checkout main

git pull origin main

git merge develop

npm run build

git add -f dist # força adicionar dist mesmo se estiver no .gitignore

git commit -m "Build final versão X.Y"

git push origin main

Marcar release com tag

git tag -a v1.0.0 -m "Release versão 1.0.0"

git push origin v1.0.0

Dicas

- Use branches separadas para cada feature ou correção.
- Teste sempre no develop antes de enviar para main.
- Utilize tags para marcar cada release.
- O -f no git add -f dist é necessário se a pasta /dist estiver no .gitignore.