**Curso Técnico e Superior de Desenvolvimento de Software**

**Ano letivo 2023/2024**

**Turma 1º ano**

**RELATÓRIO**

**PROJETO FINAL DAS**

**Criação de Repositório GitHub**

**Leonardo Oliveira e Paulo Almeida**

Março 2024

# **ÍNDICE**

# 

# 

[**ÍNDICE**](#_xgs2ctge3qw2) **2**

[**1. INTRODUÇÃO**](#_rrr87pengts6) **3**

[**2. CRIAÇÃO DO REPOSITÓRIO**](#_2u1gxa8pzect) **4**

[**3. DEMONSTRAÇÃO DAS OPERAÇÕES EFETUADAS**](#_qcy43jyjj2bu) **5**

[3.1 CRIAÇÃO DE BRANCHES](#_l3bktksu1kn3) 5

[3.2 NÍVEIS DE ACESSO NO GITHUB](#_qvc6eom5hf4) 5

[3.3 REVISÃO DE CÓDIGO](#_i5xpk8xc1oe) 5

[3.4 CONFIGURAÇÃO DO GITIGNORE](#_y2ovxilvbit2) 5

[3.5 CONTROLE DE VERSÕES DE RELATÓRIO](#_d44qtrwuptb8) 5

[**4. CONCLUSÃO**](#_9bl7rc82gw0n) **6**

[**5. REFERÊNCIAS**](#_2qbf3aiasuw1) **7**

# 

# **1. INTRODUÇÃO**

# 

Neste relatório, apresentamos o trabalho final desenvolvido na disciplina de Desenvolvimento Ágil de Software (DAS).

O projeto consistiu na criação de um repositório no GitHub utilizando o Git Bash e aplicando o modelo GitFlow para a gestão de branches.

O objetivo do projeto foi implementar as práticas ágeis aprendidas em sala de aula, incluindo controle de versões, revisão de código e gerenciamento de branches de forma eficiente.

# 

# **2. CRIAÇÃO DO REPOSITÓRIO**

O repositório foi criado utilizando o comando git flow init, seguindo o modelo GitFlow para a gestão de branches.

# 

# **3. DEMONSTRAÇÃO DAS OPERAÇÕES EFETUADAS**

A seguir, apresentamos as operações efetuadas no desenvolvimento do projeto:

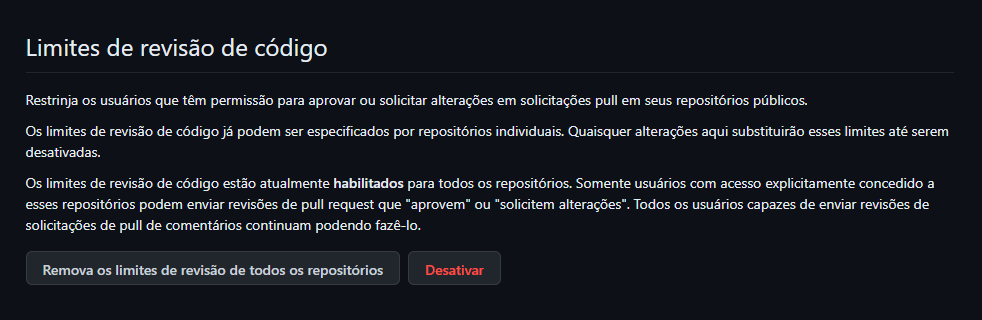
## **3.1 CRIAÇÃO DE BRANCHES**

Foram criados os branches necessários para a utilização do modelo GitFlow, incluindo develop, feature, release e hotfix

## **3.2 NÍVEIS DE ACESSO NO GITHUB**

Foram configurados dois níveis de acesso no GitHub: um para os desenvolvedores, permitindo a submissão de código, e outro para os proprietários do repositório, com permissão para alterar a visibilidade.

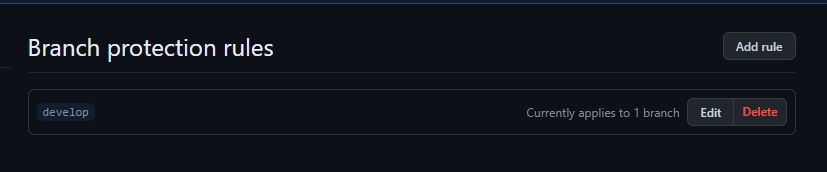




## 

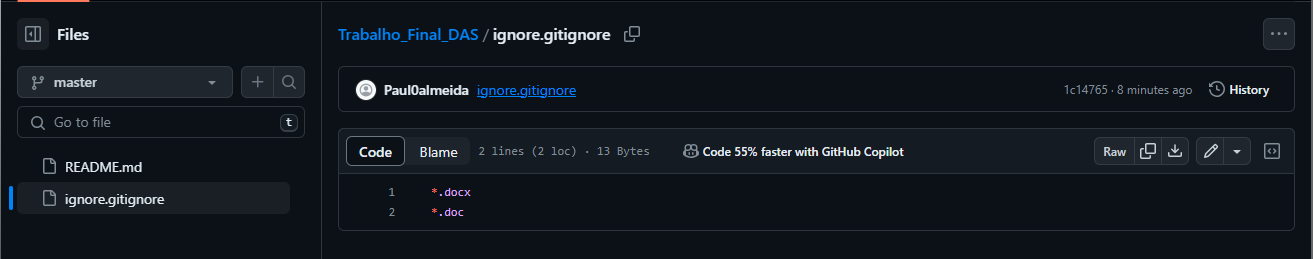
## **3.3 REVISÃO DE CÓDIGO**

Tornamos obrigatória a revisão de código antes da aprovação de um pull request, garantindo a qualidade do código submetido.



## **3.4 CONFIGURAÇÃO DO GITIGNORE**

Adicionamos um arquivo .gitignore para ignorar os arquivos .docx e .doc, garantindo que esses arquivos não sejam versionados no repositório.

****

## **3.5 CONTROLE DE VERSÕES DE RELATÓRIO**

Controlamos as versões do relatório utilizando o repositório, com as seguintes etapas:

a. Realizamos o carregamento inicial do arquivo no branch develop.

b. Efetuamos cinco alterações no próprio branch develop com os devidos comentários.

c. Contribuímos com uma release para o branch master.

d. Realizamos um hotfix utilizando a gestão de branches GitFlow.

## 

# **4. CONCLUSÃO**

Em conclusão, o projeto demonstrou a aplicação prática dos conceitos aprendidos na disciplina de Desenvolvimento Ágil de Software. Utilizamos o GitFlow para a gestão eficiente de branches, configuramos níveis de acesso no GitHub, implementamos revisão de código e controlamos as versões do relatório utilizando o repositório. Este trabalho proporcionou uma experiência valiosa no desenvolvimento de software ágil e na utilização de ferramentas de controle de versões.

# **5. REFERÊNCIAS**

<https://docs.github.com/en/account-and-profile/setting-up-and-managing-your-personal-account-on-github/managing-user-account-settings/permission-levels-for-a-personal-account-repository>

# 

## 

# 

# 

# 