



ISTEC - Instituto Superior de Tecnologias Avançadas do Porto

Projeto Final Gitflow

DAS - Desenvolvimento Ágil de Software

Aníbal Freire

Curso: Desenvolvimento de Software

March 8, 2025

Contents

1	Título do Projeto	2
2	Introdução	2
3	Membros da Equipa e Tarefas	2
4	Link para o repositório	3
5	Código git usado	3
6	Ferramentas e Tecnologias	4
6.1	Overleaf	4
6.2	Git	5
6.3	MinGW	5
6.4	Github	6
6.5	GIMP	7
7	Webografia	8
8	Considerações Finais	8

1 Título do Projeto

Projeto Final Gitflow.

2 Introdução

O objetivo deste projeto é criar um repositório no GitHub, usando o GitFlow, no qual devemos implementar boas práticas de controle de versão e gestão de branches.

3 Membros da Equipe e Tarefas

Aníbal Freire

- Criação do Github
- Criação do relatório

Diogo Lima

- Auxílio com pesquisa
- Realização de testes das permissões do Github

4 Link para o repositório

<https://github.com/zewo2/das-final>

5 Código git usado

Para a criação do repositório local e uso do git flow para a manipulação e gestão deste foram usados os seguintes comandos em git bash:

```
git clone https://github.com/zewo2/das-final
cd das-final
git flow init
```

Depois, para a subsequente manipulação do repositório remoto, upload das alterações e controlo de versão foram usados os seguintes comandos:

```
git remote add origin https://github.com/zewo2/das-final
```

Após a adição da origem remota, foi realizada a base inicial do relatório e colocada no repositório local, sendo estas alterações depois enviadas para o remoto:

```
git status
git add .
git commit -m "mensagem de commit"
git push origin develop
```

Após concluir o relatório e fazer as cinco (5) alterações e commits necessários, foi criada uma release:

```
git flow release start Rel-1.0 develop
git flow release finish Rel-1.0 -m "v1"
git push origin develop --tags
```

Como foram denotados alguns erros no relatório, foram usados os comandos referentes às funções de hotfix para corrigir os mesmos:

```
git flow hotfix start Rel-1.1 develop
git flow hotfix finish Rel-1.1
git push origin develop
```

Após isto, foi realizada uma pull request para fazer merge da branch develop à main, de modo a finalizar este projeto.

6 Ferramentas e Tecnologias

Para realizar este projeto usufrui das seguintes ferramentas:

- Overleaf - Para escrever este relatório
- Git - Controlar repositórios locais e remotos
- MinGW - Para interagir com o Git.
- Github - Para controlar a versão final e certificar que esta estava sincronizada através de todos os dispositivos e permitir a colaboração remota com o resto da equipa.
- GIMP - Para editar algumas imagens usadas no relatório

6.1 Overleaf



O Overleaf [1] é um editor de Latex que não requer qualquer instalação, por ser online, cujo permite a fácil colaboração entre vários membros de um projeto.

Este foi projetado para tornar o processo de escrita e criação de relatórios técnicos mais fácil e rápido, através da utilização de templates e dos seus intuitivos editores Visual e de Código, permitindo utilizadores com diferentes competências realizarem o seu projeto de forma intuitiva.

O Overleaf permite ainda a utilização de gestores bibliográficos para a inserção de citações e referências bibliográficas, sendo alguns exemplos deste tipo de ferramentas o Zotero e o Mendeley.

No âmbito deste projeto, utilizei o Overleaf para me permitir escrever este relatório.

6.2 Git



O Git [2] é um sistema de controle de versões distribuído, usado principalmente no desenvolvimento de software, mas pode ser usado para registrar o histórico de edições de qualquer tipo de arquivo (Exemplo: alguns livros digitais são disponibilizados no GitHub e escrito aos poucos publicamente).

O Git foi inicialmente projetado e desenvolvido por Linus Torvalds para o desenvolvimento do kernel Linux, mas foi adotado por muitos outros projetos.

Cada diretório de trabalho do Git é um repositório com um histórico completo e habilidade total de acompanhamento das revisões, não dependente de acesso a uma rede ou a um servidor central.

No âmbito deste projeto, utilizamos o Git para trabalhar com repositório locais e remotos com sincronização e controle de versão.

6.3 MinGW



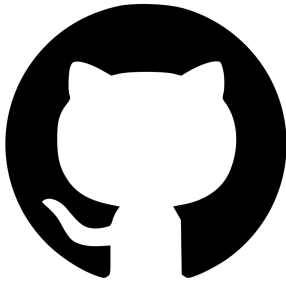
O MinGW [3] (Minimalist GNU for Windows) é uma versão portada para Microsoft Windows do conjunto de ferramentas GNU.

Este software inclui um conjunto de arquivos cabeçalho para a API do Windows que permite aos programadores usarem o GCC para criarem programas nativos em Windows sem precisarem contar com uma emulação em tempo real de um sistema Unix-like.

Este era originalmente uma ramificação do Cygwin, que fornece um suporte Unix-like maior para Windows. O MinGW é software livre, estando os seus cabeçalhos Win32 disponíveis em domínio público enquanto as ferramentas estão em GNU General Public License.

No âmbito deste projeto, utilizamos o MinGW como interface para poder interagir com o Git.

6.4 Github



O GitHub [4] é uma plataforma de hospedagem de código-fonte e arquivos com controlo de versão utilizado o Git. Ele permite que programadores, utilitários ou qualquer utilizador registado na plataforma contribuam em projetos privados e/ou Open Source de qualquer lugar do mundo.

Este é amplamente utilizado por programadores para a divulgação dos seus trabalhos ou para que outros programadores contribuam para o seu projeto, para além de promover fácil comunicação através de recursos que relatam problemas ou misturam repositórios remotos (issues, pull request).

O GitHub é mundialmente usado e chega a ter mais de 36 milhões de utilizadores ativos mundialmente contribuindo em projetos comerciais ou pessoais. Hoje o GitHub abriga mais de 100 milhões de projetos, alguns deles sendo conhecidos mundialmente, como o WordPress, GNU/Linux, Atom e Electron.

No âmbito deste projeto, usamos o Github para controlar as versões atuais do projeto, podermos colaborar remotamente e corrigir erros sem haver necessidade de processos complexos e demorados.

6.5 GIMP



O GIMP (GNU Image Manipulation Program) (originalmente (General Image Manipulation Program) [5], ou traduzido em Português (Programa de Manipulação de Imagem do GNU) é um programa de software "open-source" focado na criação e edição de imagens bitmap, e desenhos vetoriais.

O GIMP foi criado por estudantes como uma alternativa livre e gratuita ao Adobe Photoshop. Foi um projeto universitário que amadureceu bastante e hoje alcança expressiva popularidade, sendo utilizado por artistas amadores e profissionais do ramo.

O GIMP foi criado para a criação ou manipulação de imagens e fotografias, podendo também criar gráficos, logótipos, redimensionar fotos, alterar cores, combinar imagens utilizando camadas, remover partes indesejadas e converter arquivos entre diferentes formatos de imagem digital.

No âmbito deste projeto, utilizei o GIMP para redimensionar e alterar algumas imagens.

7 Webografia

- 1 Overleaf, "<https://overleaf.com/about/features-overview>", 2025
- 2 Git, "<https://pt.wikipedia.org/wiki/Git>", 2025
- 3 MinGW, "<https://pt.wikipedia.org/wiki/MinGW>", 2025
- 4 GITHUB, "<https://docs.github.com/pt/get-started/using-git/about-git>", 2025
- 5 GIMP, "<https://docs.gimp.org/2.8/pt-BR/introduction.html>", 2025

8 Considerações Finais

A implementação deste repositório demonstrou o passo a passo de como o fizemos através do modelo GitFlow, garantindo um fluxo de trabalho mais seguro e eficiente.

Além disso, o controle de versões aplicado permitiu-nos manter um histórico detalhado das alterações realizadas, facilitando futuras manutenções e melhorias.

Por fim, a realização deste projeto final, permitiu-nos melhorar ainda mais as nossas competências e familiaridade com o Git e o modelo GitFlow.