

Link Github: <https://github.com/TonyOsman/GlobalSolution-TonyKhaledOsman553050-InaciaDosSantosSilva553401.git>

Integrantes da Dupla:

1. Tony Khaled Osman – RM553050
2. Inacia dos Santos Silva – RM553401

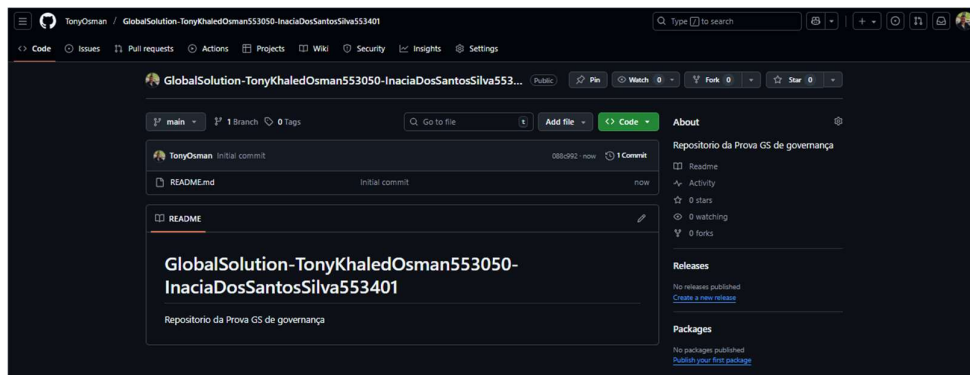
DISCIPLINA: GOVERNANÇA E MELHORES PRÁTICAS EM DESENVOLVIMENTO DE SISTEMAS

VOCÊ DEVERÁ EDITAR TODAS AS RESPOSTAS EM UM DOCUMENTO NO WORD OU OUTRO EDITOR DE TEXTOS E AO FINAL, GERAR UM PDF DO DOCUMENTO NO MESMO LOCAL ONDE PRODUZIU O ORIGINAL.

Considerando o escopo do seu projeto da Global Solution, faça:

A. REPOSITÓRIO GIT – parte 1 (peso 3, sendo 1 ponto para cada item):

a.1. Crie um repositório **público** no GITHUB.com para o seu projeto com Branch main e readme, com o nome dos integrantes do seu grupo no readme (tire print screen da tela do repositório criado com a Branch main e cole no documento de resposta da sua prova);



a.2. Usando o Git CMD ou BASH, clone o repositório em uma pasta local projeto (tire print screen da tela que mostra a execução do comando de clonagem e cole no documento de resposta da sua prova);

```
labsfiap@L2201MICR025 MINGW64 /d/tonyo/GlobalSolution-final
$ git clone https://github.com/TonyOsman/GlobalSolution-TonyKhaledOsman553050-InaciaDosSantosSilva553401.git
Cloning into 'GlobalSolution-TonyKhaledOsman553050-InaciaDosSantosSilva553401'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (3/3), done.
```

a.3. Inicialize o GIT FLOW projeto (tire print screen da tela que mostra a execução do comando e cole no documento de resposta da sua prova).

```

labsfiap@L2201MICR025 MINGW64 /d/tonyo/GlobaISolution-Final/GlobaISolution-TonyKhaledOsman553050-InaciaDosSantosSilva553401 (main)
$ git flow init

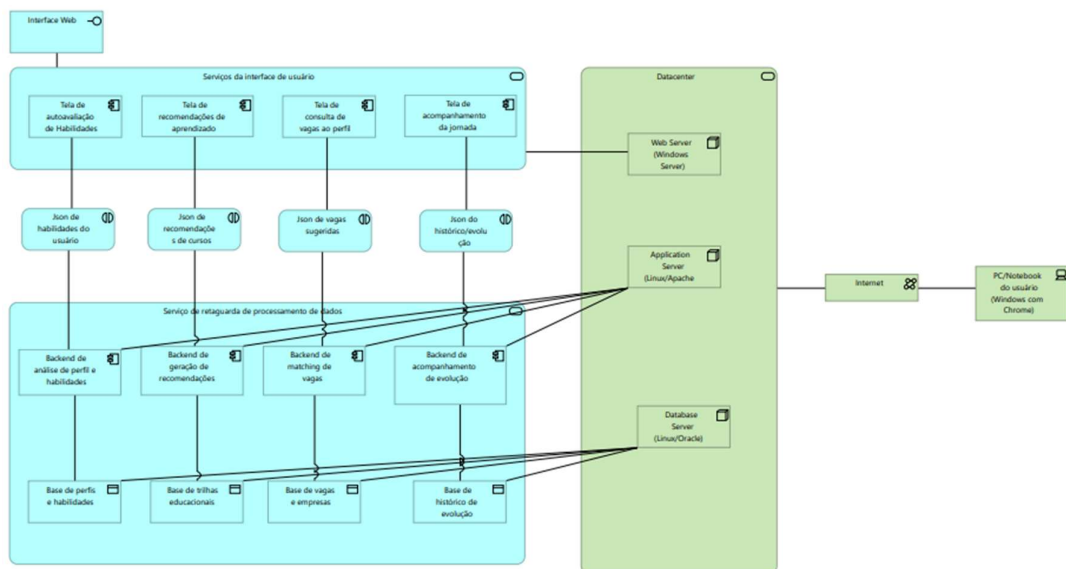
Which branch should be used for bringing forth production releases?
- main
Branch name for production releases: [main] main
Branch name for "next release" development: [develop] develop

How to name your supporting branch prefixes?
Feature branches? [feature/] feature
Bugfix branches? [bugfix/] bugfix
Release branches? [release/] release
Hotfix branches? [hotfix/] hotfix
Support branches? [support/] support
Version tag prefix? []
Hooks and filters directory? [D:/tonyo/GlobaISolution-Final/GlobaISolution-TonyKhaledOsman553050-InaciaDosSantosSilva553401/.git/hooks]
  
```

OBS: todos arquivos que você criar na solução da prova (WORD, PDF, ARCHI) precisam estar na pasta clonada do HUB. Use feature, develop, release, main, na sequência correta de uso do FLOW

B. Diagrama de arquitetura de sistema, usando a ferramenta ARCHI e o framework TOGAF (peso 4, sendo 1 ponto para cada item- copie e cole o desenho da arquitetura no documento de resposta da sua prova), representando:

- b.1. Camadas de serviços da solução e forma de acesso aos serviços pelo usuário;
- b.2. Componentes de aplicação;
- b.3. Componentes de dados;
- b.4. Conexões entre os itens/componentes da arquitetura.



OBS: desenhe somente a arquitetura da parte principal da sua solução. Ignore cadastros básicos de dados (CRUD). Não é necessário montar a visão de arquitetura, arquitetura de negócio e tecnologia. Esse ARCHI precisa ser criado na pasta do seu PC que é clone do HUB.

C. REPOSITÓRIO GIT – parte 2 (peso 3, sendo 1 ponto para cada item):

Termine a edição do arquivo de resposta da sua prova, colocando o nome e RM dos integrantes do seu grupo no alto da 1ª página (se tiver feito em grupo) e...

c.1. Usando o Git CMD ou BASH, faça o commit do documento na pasta GIT local (clonada) na Branch feature, depois atualize o develop (tire print screen da tela que mostra a execução dos comandos e cole no documento de resposta da sua prova);

```
labsfiap@L2201MICR025 MINGW64 /d/tony0/GlobalSolution-final/GlobalSolution-TonyKhaledOsman553050-InaciaDosSantosSilva553401 (main)
$ git add .
$ git commit -m "Commit do documento de respostas"
[main c575364] commit do documento de respostas
Committer: Laboratório FIAP <labsfiap@fiap.com.br>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly:

    git config --global user.name "Your Name"
    git config --global user.email you@example.com

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 3SI_GS_1oSemestre_2025_PROVA_PRESENCIAL.docx
```

c.2. Faça a atualização da Branch main após atualizar a develop, via processo de release (tire print screen da tela que mostra a execução dos comandos e cole no documento de resposta da sua prova);

```
labsfiap@L2201MICR025 MINGW64 /d/tony0/GlobalSolution-final/GlobalSolution-TonyKhaledOsman553050-InaciaDosSantosSilva553401 (develop)
$ git checkout -b release/v1.0 develop
Switched to a new branch 'release/v1.0'
```

```
labsfiap@L2201MICR025 MINGW64 /d/tony0/GlobalSolution-final/GlobalSolution-TonyKhaledOsman553050-InaciaDosSantosSilva553401 (release/v1.0)
$ git commit -m "entrega"
[release/v1.0 3114476] entrega
Committer: Laboratório FIAP <labsfiap@fiap.com.br>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly:

    git config --global user.name "Your Name"
    git config --global user.email you@example.com

After doing this, you may fix the identity used for this commit with:

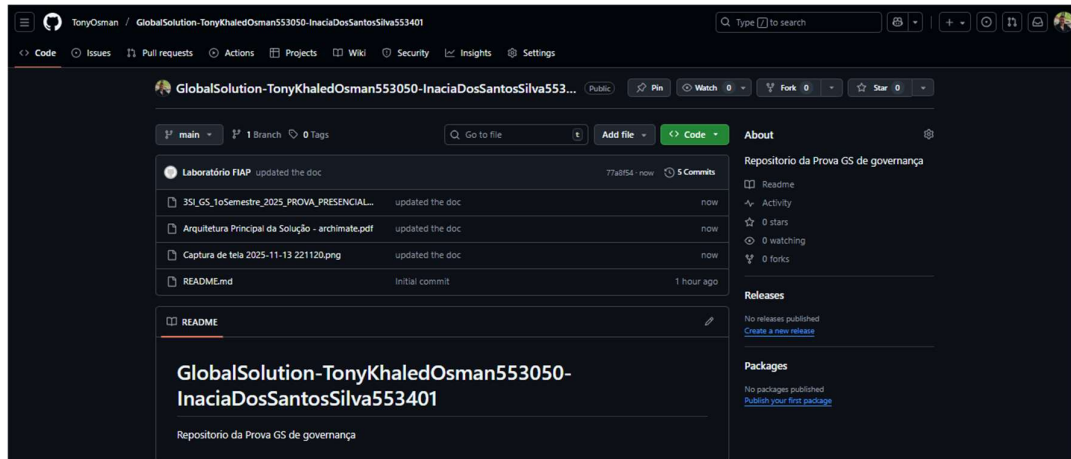
    git commit --amend --reset-author

1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 3SI_GS_1oSemestre_2025_PROVA_PRESENCIAL.docx
```

```
labsfiap@L2201MICR025 MINGW64 /d/tony0/GlobalSolution-final/GlobalSolution-TonyKhaledOsman553050-InaciaDosSantosSilva553401 (main)
$ git merge release/v1.0
Merge made by the 'ort' strategy.
```

c.3. Atualize a cópia remota, enviando as Branches develop e main para o GITHUB (tire print screen da tela que mostra a execução do comando de sincronização de dados e da tela do GITHUB com as branches atualizadas).

```
labsfiap@L2201MICR025 MINGW64 /d/tony0/GlobalSolution-final/GlobalSolution-TonyKhaledOsman553050-InaciaDosSantosSilva553401 (main)
$ git push origin main
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 8 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 185.06 KiB | 20.56 MiB/s, done.
Total 5 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/TonyOsman/GlobalSolution-TonyKhaledOsman553050-InaciaDosSantosSilva553401.git
088c992..9856d84  main -> main
```



OBSERVAÇÃO: a parte “C” da prova não estará na 1ª versão do documento de resposta de prova que você fez release e enviou ao HUB porque você precisa fazer o commit do documento fechado. Quando executar os passos da parte “C”, guarde os comandos e resultados em um documento à parte e depois, atualize o documento completo da resposta de prova e refaça a publicação do documento no HUB só que, esse segundo ciclo de execução de comandos de commit, término de feature, release e atualização do HUB não precisa ser documentado.

ENTREGUE O ARQUIVO PDF COM TODAS AS RESPOSTAS, O QUAL TAMBÉM CONTÉM O LINK DE ACESSO AO SEU GITHUB, NA ÁREA DE ENTREGA DE TRABALHOS DEFINIDA PELO PROFESSOR PARA ESTA PROVA.