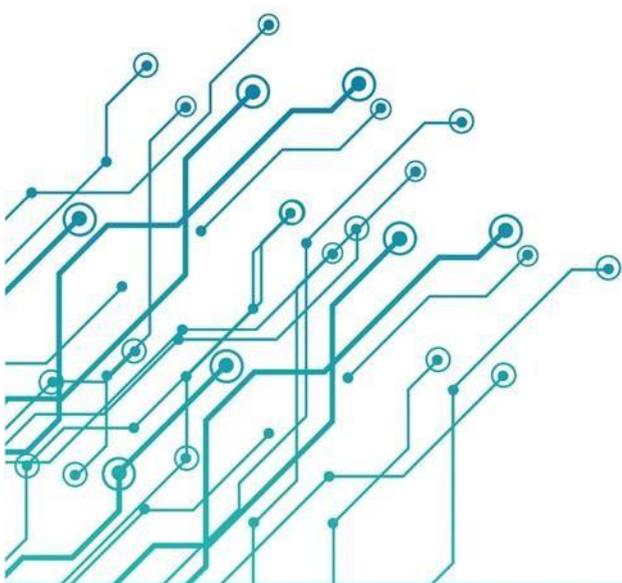


PADRÕES DE REPOSITÓRIO GITLAB - MEC

Sistema: [Boas práticas do Gitlab -
MEC]Versão:1.0



Sumário

1. Introdução	3
2. Objetivo do Documento	4
3. Gitlab	5
4. Política de acesso	7



MANUAL DE BOAS PRÁTICAS DO GITLAB - MEC

Controle de Versões			
Versão	Data	Autor	Descrição
0.1	18/02/2024	Wisley Alves couto	Criação do documento
1.0	18/02/2024	Shirlane Sousa	Atualização e validação

1 Introdução

Todo projeto deve possuir um repositório e uma estrutura padrão de pastas e subpastas que remetem ao alinhamento com a Metodologia de Desenvolvimento de Sistemas (MDS), mas ao mesmo tempo, é permitida a criação de outras pastas, principalmente relacionadas ao produto e/ou resultado gerado devido à singularidade que cada projeto possui.

Visando atingir essa organização, o repositório do MEC foi dividido em grupos de modo que possibilite uma classificação lógica dos repositórios existentes e um melhor gerenciamento da política de acesso a estes; foi definida a padronização de nomenclatura dos repositórios a serem criados e a estrutura de pastas para o armazenamento da documentação requisitada na MDS Agil.

Este documento visa descrever as boas práticas do uso da ferramenta gitlab em ambiente corporativo, bem como a padronização dos repositórios e a serem desenvolvidos e mantidos no ambiente da STIC do Ministério da Educação.

1.1 Definições, Acrônimos e Abreviações

Abreviação	Descrição
GITLAB	O GitLab é um gerenciador de repositório de software baseado em git
COMMIT	JSON Web Tokens (Chave de Conexão para Objetos Secretos)
BOARD	Representational State Transfer (Formato de Transferência de Dados pela Web)
DOC/DOC-SIS	Single Sign-On (Acesso Unico)
GRUPO	Design Pattern para transformar de maneira incremental uma aplicação monolítica em microsserviços, substituindo uma a uma as funcionalidades por novos microsserviços
PROJETO	Nome dado no gitlab para o projeto
HU	história de usuário
ISSUES	Use questões para colaborar em ideias, resolver problemas e planejar o trabalho. Compartilhe e discuta propostas com sua equipe e com colaboradores.



CODIGO-FONTE	Grupo e repositório destinado ao armazenamento do arquivos de código-fonte da aplicação
---------------------	-----------------------------------------------------------------------------------------

Tabela 1-Tabela de abreviações

1.2 Grupos de repositório

Nome do grupo	Descrição
arquitetura	Grupo destinado aos repositórios tanto de documentação quanto de código fonte de responsabilidade da arquitetura.
codigo	Grupo destinado aos repositórios de código-fonte dos sistemas.
doc-sis	Grupo destinado aos repositórios de documentação que envolva “desenvolvimento de sistemas”.

Tabela 2-grupo de repositório

2 Objetivo do Documento

O objetivo deste documento é apresentar e padronizar os sistemas que serão desenvolvidos por empresas parceiras do MEC, sejam elas universidades, institutos federais ou empresas regidas pelos contratos administrativos.

3 Gitlab

3.1. Visão Geral

O Git é um sistema open-source de controle de versão utilizado pela grande maioria dos desenvolvedores atualmente. Com ele podemos criar todo histórico de alterações no código do nosso projeto e facilmente voltar para qualquer ponto para saber como o código estava naquela data. No MEC em seu ativo institucional na STIC, tem em seu parque tecnológico uma versão *open source*, versão da comunidade, no entanto, a fabricante da ferramenta oferece versões por meio de licenciamento.

Algumas das características essenciais do GIT são:

- Um sistema de controle de versão distribuído, o GIT segue uma abordagem peer to peer, contrário de outros como o Subversion (SVN) que segue um modelo baseado em cliente-servidor.
- GIT permite aos desenvolvedores ter uma infinidade de ramos de código completamente independente. Criação, exclusão e fusão desses ramos é simples e não leva tempo.
- No GIT, todas as operações são atômicas. Isso significa que uma ação pode ter sucesso ou falhar (sem fazer nenhuma alteração). Isso é importante porque em alguns sistemas de controle de versão (como





o CVS) onde as operações não são atômicas, se uma operação de repositório é suspensa, ela pode deixar o repositório em um estado instável.

- No GIT, tudo é armazenado dentro da pasta .git. Isso não é o mesmo em outros VCS como SVN e CVS onde os metadados de arquivos são armazenados em pastas ocultas (por exemplo, .cvs, .svn, etc.)
- GIT usa um modelo de dados que ajuda a garantir a integridade criptográfica de qualquer coisa presente dentro de um repositório. Cada vez que um arquivo é adicionado ou um commit é feito, suas somas de verificação são geradas. Da mesma forma, eles são recuperados através de suas somas de verificação também.
- Outra característica presente no GIT é sua área de teste ou índice. Na área de preparação, os desenvolvedores podem formatar commits e receber feedback antes de aplicá-los.

3.1 Estrutura e Ramificação do Gitlab

3.1.1 Ambiente código-fonte

No ambiente da STIC, há uma separação clara dos repositório de código-fonte para os repositórios de documentação.

Branchs no Git são formas de termos uma mesma versão do código sofrendo alterações e recebendo *commits* de diferentes fontes e inclusive por diferentes desenvolvedores.

Para os repositório de código-fonte o padrão de criação deve ser:

Nome do grupo/repositorio

Exemplo:

Grupo:

<https://gitlabbuilder.mec.gov.br/segmec>

Repositório:

<https://gitlabbuilder.mec.gov.br/segmec/backend>



Exemplo:

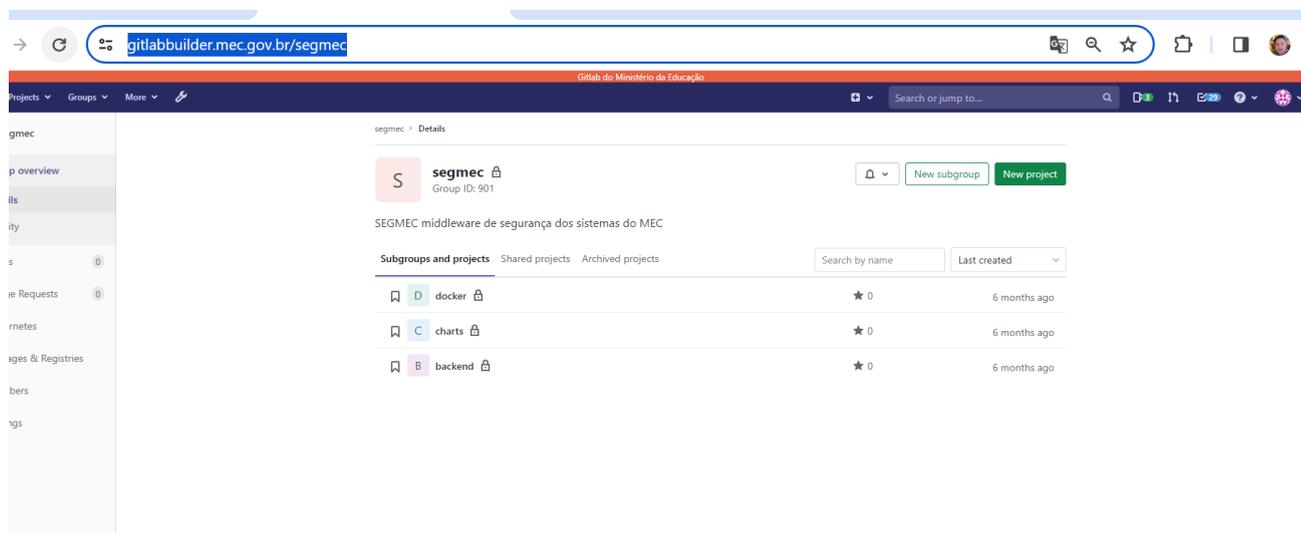


Imagem 01 - padrão repositório

Segue abaixo conceitos que são usados em ambiente de desenvolvimento:

Para repositório de código-fonte, por padrão as branches são: master, develop, release e hotfix que são as branches de produção (master), homologação e desenvolvimento.

Temos também as branches: release/xxxx e hotfix/xxxxx que são branches que trabalham em uma determinada versão que será publicada em homologação para caso de evolução e branch de hotfix/bugfix que são branches para correção de incidentes ou de sustentação.

Conforme exemplo abaixo:

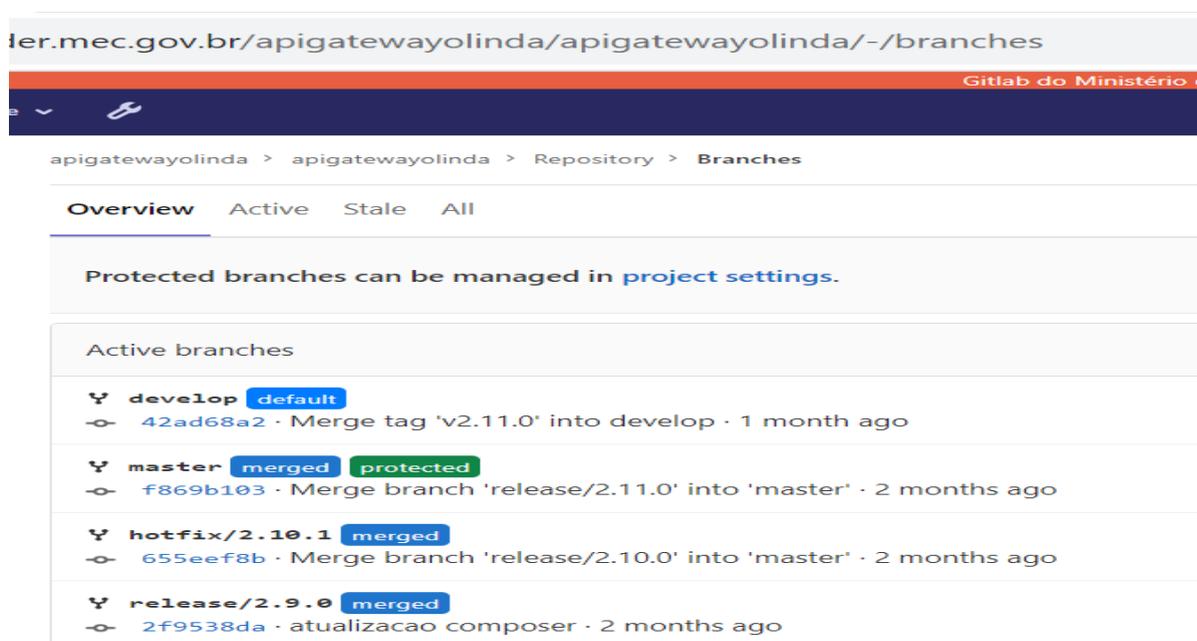


Imagem - 02 estrutura de branches do gitlab do MEC

Na imagem abaixo podemos ver um exemplo de trabalho com vários ramos e *commits* aplicados. Veja que em alguns pontos da história os ramos são unidos para que as alterações de um ramo sejam aplicadas a outro.



Imagem - 03 ramificação gitlab source de <https://leanpub.com/git-flow/read>

3.1.2 Ambiente documentação

No ambiente da STIC, há uma separação clara dos repositórios de **codigo-fonte** para os repositórios de **documentação**, somente será permitido criação de repositórios e grupo que sigam os padrões definidos nesse documento.

Para os repositório de documentação, inclui-se todos os projetos de documentação, contendo, quadros, wike e issues bem como qualquer outra forma de trabalho em consonância com MDS Agil do MEC, todos os itens citados devem estar agrupados dentro do grupo doc-sis e no seguinte padrão conforme exemplo:



gitlabbuilder.mec.gov.br/doc-sis

Nome do Repositório	Descrição	Estrelas	Atualizado há
PNEERQ		0	6 days ago
pba-saldos		0	1 week ago
documentacao-simec	documentacao do sistema SIMEC	0	1 week ago
conferencia		1	1 week ago
Documentacao-EducacaoESaudeMaisMedico	Repositório de documentação do Educação em Saúde (antes chamado Mais Médicos) =...	0	3 months ago
documentacao-simec_projovem_campo_2021	Repositório de documentação do simec projovem campo 2021	0	3 months ago
documentacao-SIMEC_Projovem_Urbano_2021		0	3 months ago
documentacao-prounigestao	Repositório de documentação do novo prouni gestão	0	3 months ago
documentacao-simec-par4-questionarios		0	3 months ago
documentacao-isfgestao	documentacao-isfgestao migrada do SVN	0	5 months ago
documentacao-segmec	Projeto de documentação do sistema de segurança de usuários de sistemas do MEC.	0	6 months ago
PDDE-Interativo - Escola Acessível 2.0	Projeto de documentação "Escola Acessível"	0	3 months ago
SIMEC - Escola da Terra Especialização	Projeto de documentação para o módulo do SIMEC, Escola da Terra Especialização	0	6 months ago
documentacao-redeintegra		0	6 months ago
documentacao-novosimec	Repositório destinado ao armazenamento da documentação do projeto/sistema novo SI...	0	2 months ago

<https://gitlabbuilder.mec.gov.br/doc-sis/documentacao-sistec/>

Para os repositórios de documentação, quadros do gitlab(kanban) devem ser criados seguintes os critérios definidos na MDS AGIL. No entanto cada projeto deve ter seus devidos quadros, com ISSUES e HU e outras informações necessárias para o desenvolvimento do projeto.

- **Para documentação do projeto:**

documentacao + "-" + <sigla do projeto/sistema>

Exemplo: documentacao-sistec

Parte da Nomenclatura	Descrição
documentacao	Texto fixo para os repositórios que forem destinados à documentação do projeto/sistema, tornando o nome deste repositório diferente do repositório destinado ao código-fonte (quando houver).

4.0 Política de Acesso

Como os repositórios são divididos em grupos, a política de acesso dos repositórios do MEC determina algumas





diretrizes para as permissões de **leitura**¹ e **escrita**² aos grupos existentes, conforme descrito abaixo:

- **arquitetura:**
 - O acesso ao grupo arquitetura é permitido somente para membros da equipe de arquitetura, que por sua vez possuem o perfil **Owner**³ para todo o repositório do MEC, tendo permissão para realizar qualquer ação (leitura, escrita e configuração) não somente nos repositórios que compõem o grupo “arquitetura”, mas também nos demais repositórios do MEC.
- **Código / Api:**
 - O acesso ao grupo codigo é prioritário para os desenvolvedores que atuam nos projetos/sistemas do MEC, portanto, estes possuem o perfil **Developer**⁴, tendo permissão de escrita a quaisquer repositórios que compõem o grupo. Caso necessário, o gerente do projeto/sistema pode possuir o perfil **Reporter**⁵ apenas para o(s) repositório(s) referente(s) ao projeto/sistema de sua responsabilidade.
- **/ doc-sis:**
 - O acesso aos grupos doc-sis é permitido para qualquer usuário que necessita trabalhar com documentação geral ou documentação de sistemas. Portanto, o usuário que tiver permissão de escrita a um determinado repositório do grupo doc-sis, deve possuir o perfil **Master**⁶ para este repositório específico e o perfil **Reporter** para o grupo em questão. Ou seja, este pode visualizar todos os repositórios que compõem o mesmo grupo, porém, pode atualizar somente repositório(s) que sejam de sua necessidade.
 - Exemplo: Um gerente de operação do sistema SIMEC deve ser inserido no grupo doc-sis como *Reporter* e receber o perfil *Master* para o repositório ../doc-sis/documentacao-simec.

Informações uteis:

MGDS Ágil do MEC:

Boas praticas de criação de banco de dados e tabelas

https://portalstic.mec.gov.br/images/pdf/normativos/CGGD_Anexo_Nomenclatura_Objeto_Banco_de_Dados.pdf

5.0 Criação de repositório

A equipe de arquitetura é responsável pelo gerenciamento dos repositórios no MEC. Neste caso, ao necessitar de um novo repositório para armazenamento de documentação do projeto ou outra necessidade, somente o gerente poderá abrir uma requisição por meio da ferramenta IT SM solicitando a criação do repositório. Na requisição deverá ser descrita a necessidade e quais usuários deverão ter permissão de escrita ao repositório.

A equipe de arquitetura deverá analisar a requisição devidamente e solicitar demais informações caso a descrição seja insuficiente. Com o objetivo de criar o repositório solicitado dentro do grupo correspondente à necessidade apresentada; com o nome conforme padronização existente e liberar o acesso aos usuários relacionados de acordo com a política de acesso documentada. Não será permitido criação de código-fonte ou de repositório de documentação fora do padrão conforme definido nesse documento

¹ Permissão para somente visualizar os arquivos em um repositório;

² Permissão para visualizar/criar/alterar/excluir arquivos em um repositório;

³ Owner (Perfil no Git que dá acesso de administrador) – somente a equipe de arquitetura pode ter este perfil no MEC;

⁴ Developer (Perfil no Git que dá acesso de escrita a código-fonte);

⁵ Reporter (Perfil no Git que dá acesso somente de leitura);

⁶ Master (Perfil no Git que dá acesso de escrita a documentação).



