

# Estrutura e Automações

## Estrutura do repositório

A estrutura do repositório segue o seguinte formato:

```
infra/
├─ clusters/      # Um diretório por cluster Kubernetes
│  └─ admin/      # Cluster admin (Terraform + Terragrunt)
├─ modules/       # Módulos Terraform reutilizáveis
│  └─ vsphere-vm/ # Provisionamento de VM no vSphere
│  └─ k3s-cluster/ # Cluster k3s completo
│  └─ k3s-server/ # Nó servidor k3s
│  └─ k3s-agent/  # Nó agente k3s
├─ ansible/       # Playbooks de pós-configuração
│  └─ site.yml    # Configura clusters k8s (addons, discos, etc)
│  └─ edge.yml    # Configura VMs de borda (nginx, firewall)
└─ .gitlab-ci.yml # Pipeline de CI
```

Cada diretório em `clusters/` é um módulo Terragrunt que referencia os módulos reutilizáveis de `modules/`. O estado do terraform está sendo salvo no próprio GitLab. Desse modo, todos os comandos do terraform devem ser executados diretamente pelo GitLab CI ou utilizando o estado remoto do GitLab.

## Nomenclatura de branches neste repositório

Prefixo	Exemplo
<code>feat/</code>	<code>feat/cluster-portal</code>
<code>upgrade/</code>	<code>upgrade/k3s-1.32</code> , <code>upgrade/terraform-1.10</code>
<code>fix/</code>	<code>fix/rede-admin</code> , <code>fix/disco-node1</code>

Prefixo	Exemplo
resize/	resize/admin-node1-memoria, resize/portal-node2-cpu
docs/	docs/tutorial-novo-cluster

# Pipeline de CI

O pipeline possui quatro stages executados em sequência:

validate → plan → apply → configure

Stage	Quando roda	O que faz
validate	Todo push e todo MR	Valida a sintaxe dos arquivos Terraform
plan	Todo push e todo MR	Mostra o que seria criado, alterado ou destruído
apply	Somente na <code>main</code> , <b>manual</b>	Aplica as mudanças de infraestrutura no vSphere
configure	Somente na <code>main</code> , automático após apply	Roda Ansible para configurar k8s e VMs de borda

O `apply` **nunca roda sozinho** — exige um clique manual de quem aprovou o MR. O `configure` roda automaticamente em seguida, se o `apply` for bem-sucedido.

# Variáveis de CI necessárias

Configuradas no grupo `dicom` do GitLab (Settings → CI/CD → Variables):

Variável	Descrição
<code>VSPHERE_PASSWORD</code>	Senha do usuário de serviço no vSphere
<code>ANSIBLE_SSH_PUBLIC_KEY</code>	Chave pública SSH injetada nas VMs pelo Terraform
<code>ANSIBLE_SSH_PRIVATE_KEY</code>	Chave privada SSH usada pelo Ansible (em base64)
<code>CERT_MANAGER_TSIG_SECRET</code>	Chave TSIG para desafios DNS01 do cert-manager (Obsoleto: está sendo movido para repositório <code>stacks</code> )

