

Github Organisation d'un dépôt à trois fonctions

Structure générale des trois dépôts du projet

Type de dépôt	Rôle	État	Contenu typique	Interactions
1. Dépôt documentaire	Référentiel de travail et d'analyse	● semi-statique	Fichiers .docx, .pdf, cahiers, logs PDF	Humain (audit, lecture)
2. Dépôt statique	Snapshot gelé de configuration système	● figé	/etc/, /opt/, *.yml, *.json, scripts	Aucun push, backup uniquement
3. Dépôt dynamique	CI/CD vers GitHub & Infomaniak	● actif	.env, Makefile, scripts/, pipelines	Automatisation, push/clone

Détail par dépôt

1. Dépôt documentaire (jitsi-meet-docs)

- **Emplacement** : ~/Documents/Tercium_Stage/Documentation/
- **Contenu** :
 - README.md narratif
 - Annexes .docx, .pdf du dossier final
 - Journaux (log journalier, rapport technique, historique)
 - Captures : captures/, screenshots/

2. Dépôt statique (jitsi-meet-snapshot)

- **Emplacement** : ~/Documents/Tercium_Stage/Reserves_Statiques/
- **Structure** :
 - jitsi-meet-infra_snapshot_2025-07-29/
 - |— etc_jitsi/
 - |— opt_jitsi/
 - |— prometheus_config/
 - |— scripts/
 - |— dashboards/
 - |— metadata.json
- **Particularité** : jamais modifié, compressé, voire monté en lecture seule, ou figé par :
 - sudo chattr -R +i <snapshot_dir>

3. Dépôt dynamique (jitsi-meet-infra)

- **Emplacement** : ~/Documents/Tercium_Stage/Scripts/Dossier yml/jitsi-meet-infra
- **Rôle** : CI/CD, automatisation, push vers GitHub et Infomaniak
- **Contenu type** :
 - .
 - |— .env.template
 - |— .gitignore
 - |— Makefile
 - |— scripts/
 - |— deploy_to_infomaniak.sh
 - |— check_metrics.sh
 - |— generate_ssh_github.sh
 - |— prometheus/
 - |— prometheus.yml
 - |— blackbox.yml
 - |— alert.rules.yml
 - |— dashboards/
- **Workflow** :
 - Push vers GitHub (main, dev, preprod)
 - Script scp → Infomaniak (fichiers .yml, .json, etc.)
 - Tests via Makefile (validate, reload, install_exporters...)

Étapes suivantes proposées

1. **Créer les trois arborescences** :
 - jitsi-meet-docs
 - jitsi-meet-snapshot
 - jitsi-meet-infra (déjà commencé)
2. **Générer les scripts d'init automatisée** si tu veux faire un `init_project_structure.sh`
3. **Écrire le README.md racine** qui décrit ces trois modules comme une architecture documentaire, snapshot et active (GitOps).