

Git Structure Rules

Purpose: Standard git hygiene for the multi-tenant tree. Read BEFORE `git init`, `gh repo create`, or any commit decision.

1. Repo placement matrix

Repo type	Canonical home	GitHub org/owner	Visibility default
ALAI product (Bilko, Drop, Tok)	<code>~/business/ALAI-Holding-AS/products/<product>/</code>	<code>johnatbasicas/<product></code>	private until launch
ALAI internal CLI / SDK / library	<code>~/projects/<repo-name>/</code>	<code>johnatbasicas/<repo></code>	public if open-source
ALAI infra workspace	<code>~/aisystem/</code>	n/a	n/a
ALAI Tech DOO (RS subsidiary)	<code>~/business/ALAI-Tech-D00/</code>	<code>alai-tech-doo/<repo></code>	private
Client-owned repo	<code>~/clients-external/<client>/</code>	client's GitHub org	client's choice
Personal scholarly project	<code>~/personal/scholarly/<topic>/</code>	<code>johnatbasicas/<topic></code>	public (transparency)
System orchestration	<code>~/system/</code>	<code>johnatbasicas/alai-system</code>	private

2. Required files per repo type

File	When required
<code>README.md</code>	Always
<code>CLAUDE.md</code>	Every repo where agents will be dispatched
<code>BUILD-BLUEPRINT.md</code>	Repos that build/deploy (per ZAKON P12)
<code>DEPLOY-MAP.md</code>	Repos with live deploys (CF Pages, Cloud Run, App Runner, etc.)
<code>.gitignore</code>	Always — see Section 4
<code>.github/workflows/deploy*.yml</code>	Repos with CI/CD
<code>LICENSE</code>	Public repos (default MIT for ALAI products)

3. Commit conventions — Conventional Commits (mandatory)

```
<type>(<scope>): <subject> (MC #<id>)  
  
<body – what changed and why>  
  
Co-Authored-By: <persona-name> <noreply@alai.no>
```

type ∈ feat | fix | chore | docs | style | refactor | perf | test | ci | build

4. .gitignore standards

Universal (every repo)

```
# OS  
.DS_Store  
Thumbs.db  
  
# Editor / IDE  
.vscode/  
.idea/  
*.swp  
.aider*  
.claude-scratch/  
  
# Secrets – NEVER COMMIT  

```

```
# Logs
*.log
logs/

# Build outputs
dist/
build/
out/
target/
.next/
.nuxt/
.output/
.turbo/
```

Node-specific

```
node_modules/
npm-debug.log*
yarn-debug.log*
yarn-error.log*
```

Terraform / IaC (CRITICAL after Parisa finding)

```
*.tfstate
*.tfstate.*
*.tfplan
.terraform/
.terraform.lock.hcl
crash.log
```

5. Anti-patterns — explicitly forbidden

Anti-pattern	Why it's bad	Correct alternative
<code>git add .</code> without reviewing diff	Commits unwanted files (.env, tfstate, secrets, IDE configs)	<code>git add -p</code> or explicit paths
Committing <code>.env</code> "for now"	Secrets in git history forever	Bitwarden / <code>.env.example</code> pattern

Anti-pattern	Why it's bad	Correct alternative
Force-push to shared/master	Destroys teammates' work	PRs only; force-push only on private feat branches
Cross-tenant repos	Tenant contamination, IP confusion	Split into 2 repos, separate orgs ideally
Tags pushed before work is verified live	Tag points to broken state	Tag AFTER <code>curl 200</code> + Playwright pass

6. References

- ADR-023: `~/system/architecture/decisions/ADR-023-anvil-tenant-restructure-2026-05-07.md`
- Tree blueprint: `~/system/specs/anvil-tree-blueprint-2026-05-07.md`
- Canonical registry: `~/system/specs/canonical-registry.md`
- ZAKON PI2 (deploy verification): `~/system/rules/zakon-pi2-deploy-verification.md`
- Conventional Commits spec: <https://www.conventionalcommits.org/en/v1.0.0/>

Revision #2

Created 2026-05-07 20:31:46 UTC by John

Updated 2026-06-14 20:02:34 UTC by John