

Repozitář: Kde se kód ukládá

Zatímco [Git](#) je nástroj, který provádí samotné verzování, **repozitář** je fyzické nebo cloudové místo, kde jsou tato data (včetně celé historie změn) uložena. Repozitář není jen složka se soubory; je to databáze všech stavů, kterými projekt kdy prošel.

1. Lokální vs. Vzdálený repozitář

V distribuovaných systémech (jako je Git) rozlišujeme dva typy:

- **Lokální repozitář (Local Repo):** Nachází se přímo v počítači vývojáře (ve skryté složce `.git`). Zde vývojář provádí commity, tvoří větve a experimentuje bez potřeby internetu.
- **Vzdálený repozitář (Remote Repo):** Umístěný na serveru (např. GitHub, GitLab). Slouží jako „hlavní kopie“ a synchronizační bod pro celý tým.

2. Struktura moderního repozitáře

Kromě samotného zdrojového kódu by měl každý profesionální repozitář obsahovat:

- **README.md:** Základní dokumentace (k čemu projekt slouží, jak ho spustit).
- **.gitignore:** Seznam souborů, které Git nemá sledovat (např. dočasné soubory, hesla, složky s knihovnami jako `node_modules`).
- **LICENSE:** Právní podmínky, za kterých lze kód používat.
- **Konfigurační soubory:** Nastavení pro [CI/CD](#) (např. `.github/workflows`), Dockerfile nebo nastavení linterů.

3. Populární platformy pro hosting (Remote Repos)

Ačkoliv můžete mít repozitář na vlastním serveru, většina světa využívá specializované služby:

Platforma	Charakteristika
—	—
GitHub	Největší na světě, domov open-source projektů. Vlastněn Microsoftem.
GitLab	Zaměřen na DevOps a CI/CD. Nabízí skvělou možnost hostování na vlastních serverech (Self-hosted).
Bitbucket	Od firmy Atlassian, úzce integrovaný s nástroji jako Jira a Confluence.

4. Workflow v repozitáři: Pull Requesty

Vzdálené repozitáře přinášejí zásadní funkci pro kvalitu kódu: **Pull Request (PR)** nebo **Merge Request (MR)**.

1. Vývojář nahraje (push) svou větev do vzdáleného repozitáře.
2. Požádá o sloučení do hlavní větve (PR).
3. Ostatní kolegové kód zkontrolují (**Code Review**), napíší připomínky a teprve po schválení se kód sloučí.

5. Mono-repo vs. Poly-repo

Při návrhu architektury se firmy rozhodují mezi dvěma přístupy:

- **Mono-repo:** Všechny projekty a mikroslužby firmy jsou v jednom obřím repozitáři (používá např. Google nebo Meta). Usnadňuje sdílení kódu, ale vyžaduje extrémně výkonné nástroje.
- **Poly-repo (Multi-repo):** Každá služba má svůj vlastní repozitář. Je to přehlednější pro menší týmy, ale složitější na správu verzí napříč projekty.

Související články:

- [Git: Jak se s repozitářem pracuje](#)
- [CI/CD: Automatické nasazování z repozitáře](#)
- [DevOps: Kultura spolupráce a automatizace](#)

Tagy: it repository git github gitlab development infrastructure

From:

<https://www.serviceit.cz/> - IT ENCYKLOPEDIE

Permanent link:

<https://www.serviceit.cz/doku.php?id=it:sw:repozitar>

Last update: **2026/01/02 20:29**

