

Release process of Vivo release

1. Update local branches (fetch/pull)
 - a. develop
 - b. master
2. Bump version
 - a. Create new issue in VIVO JIRA project (<https://jira.lnd.bz/projects/VIVO>) called "Release/Hotfix x.x.xx"
 - b. Create new branch issue/VIVO-xxx from master branch (git checkout -b issue/VIVO-xxx)
 - c. Bump version in file VERSION, update text in README.md, commit this change to branch issue/VIVO-xxx and push it
 - d. Create pull-request from issue/VIVO-xxx to preview and merge it
3. Create release (hotfix)
 - a. git checkout master
 - b. git flow release/hotfix start x.y.z
 - c. merge all branches to release/hotfix branch (git merge issue/VIVO-xxx, last merge should be the issue with version bump)
 - d. git flow release/hotfix finish x.y.z
4. Push
 - a. master
 - b. develop
 - c. tags
5. Control git graph (e.g. on [bitbucket](#))
 - a. if merge commits look O.K. (compare two last releases/hotfixes)
 - b. if tags have been pushed
6. Update dependencies to Vivo on projects [vivodemo](#), [vivoproject](#), if necessary and **test new release**
7. Create Confluence page with issue list like [Vivo 2.2.2 change log](#) or [Vivo 2.2.3](#)
8. Create/update release in VIVO JIRA project (tag all issues, fill in release date)
9. Update change log on Vivo web - <http://vivo.lundegaard.eu/cs/vivo-cms/change-log/>
10. Update Vivo documentation for developers and editors - [Vivo](#), [Vivo Documentation](#)
11. Let people know about new version wherever you can 🗣️

- [Release process of Vivo release](#)
- [getSentry Service | nastveni modulu raven](#)
- [Specifický title stránky](#)
- [Nasazení Vivo 2 na NGINX+HHVM](#)
- [Zprovoznění Vivo pod open_basedir](#)
- [Vivo 2 migration pack](#)
- [Rozšíření logon - přihlašování a přesměrování](#)
- [Vlastní výběr souborů pro TinyMCE 4.x](#)