

### ☐ **Software**

#### ☐ **Allgemein**

- ☐ Aktualisierung in UpdateLibrary.cpp bearbeiten
- ☐ Versionsnummer aktualisieren
  - ☐ enterprise.sql
  - ☐ Configuration.h
    - ☐ Warehouse
    - ☐ Warehouse Admin
    - ☐ LicenceGenerator (optional)
  - ☐ Handbuch
    - ☐ Generieren
    - ☐ Nach src/modules/warehouse/doc/Handbook.pdf kopiert
  - ☐ Changelog anpassen

#### ☐ **FreeBSD**

- ☐ git pull
- ☐ Aktuelle DSLib
- ☐ LicenceGenerator kompilieren (*make -f Makefile.FreeBSD clean all*)
- ☐ Warehouse Admin kompilieren (*make -f Makefile.FreeBSD clean all*)
- ☐ Warehouse kompilieren (*make -f Makefile.FreeBSD clean all*)
- ☐ Reports kompilieren (mvn package)
- ☐ Installation bauen (resources/installation/freebsd -> *./mkarchive X.Y.Z*)

#### ☐ **Ubuntu**

- ☐ git pull
- ☐ Aktuelle DSLib
- ☐ LicenceGenerator kompilieren (*make -f Makefile.Ubuntu clean all*)
- ☐ Warehouse Admin kompilieren (*make -f Makefile.Ubuntu clean all*)
- ☐ Warehouse kompilieren (*make -f Makefile.Ubuntu clean all*)
- ☐ Reports kompilieren (mvn package)
- ☐ Installation bauen (resources/installation/ubuntu -> *./mkarchive X.Y.Z*)

#### ☐ **Mac OS X**

- ☐ git pull
- ☐ Aktuelle DSLib
- ☐ LicenceGenerator kompilieren (*make -f Makefile.MacOSX clean all*)
- ☐ Warehouse Admin kompilieren (*make -f Makefile.MacOSX clean all*)
- ☐ Warehouse kompilieren (*make -f Makefile.MacOSX clean all*)
- ☐ Reports kompilieren (mvn package)
- ☐ Apps bauen/DMG erstellen
  - ☐ resources/installation/macosx -> *./mkdmg.bash*
  - ☐ Libraries in allen Paketen austauschen (mit den vorhandenen)
  - ☐ DMG erstellen (Warehouse, Warehouse Admin, LicenceGenerator, PostgreSQL, Handbuch)

#### ☐ **Windows**

- ☐ git pull
- ☐ Aktuelle DSLib
- ☐ LicenceGenerator kompilieren (*mingw32-make -f Makefile.Windows clean all*)
- ☐ Warehouse Admin kompilieren (*mingw32-make -f Makefile.Windows clean all*)
- ☐ Warehouse kompilieren (*mingw32-make -f Makefile.Windows clean all*)
- ☐ Reports kompilieren (mvn package)
- ☐ InnoSetup Setup-Routinen erstellen

#### ☐ **Abschluss**

- ☐ Tag erstellen und pushen
  - ☐ *git tag -a vX.Y.Z -m "Version X.Y.Z"*
  - ☐ *git push origin vX.Y.Z*
- ☐ Alle Pakete korrekt benennen
  - ☐ GuGIT-Warehouse-X.Y.Z\_Windows\_i386.exe
  - ☐ GuGIT-Warehouse-X.Y.Z\_OSX\_64bit.dmg
  - ☐ GuGIT-Warehouse-X.Y.Z\_Ubuntu\_1404LTS\_amd64.tar.bz2
  - ☐ GuGIT-Warehouse-X.Y.Z\_FreeBSD\_amd64.tar.bz2
  - ☐ GuGIT-Warehouse-X.Y.Z\_Handbook.pdf
- ☐ Alle Pakete zusammen in bzip2 packen (*tar cjvf GuGIT-Warehouse-X.Y.Z.tar.bz2 GuGIT-Warehouse-X.Y.Z*)

### ☐ **Website**

- ☐ Gepacktes Paket hochladen und entpacken nach /server/web/gug-it/www/files/warehouse
- ☐ Auf Website bei „Kostenlos herunterladen“ alles tauschen (auf Downloadgröße achten)
- ☐ Im „Kundencenter“ die Änderungen von „Kostenlos herunterladen“ hineinkopieren
- ☐ version.conf anpassen (s. Changelog)