

# Create your own openITCOCKPIT translation



This documentation is related to openITCOCKPIT 4

Don't know how to setup an openITCOCKPIT development system? Read our guide: [Creating a openITCOCKPIT Development System](#)

- 1. Fork the repository
  - IMPORTANT! Please make sure to clone the development branch!
- 2. Go to the locales folder
- 3. Choose a language you want to translate to
  - Step 3.1: The language folder does not exist yet
  - Step 3.2: The language folder already exists
- 4. Translate
  - If you haven't done an openITCOCKPIT translation yet, please read the Important notes section!
- 5. Commit, push & create PR
- Important notes
- Need help?

## 1. Fork the repository

Fork our repository into your personal GitHub account by pressing on *Fork* in the top right corner of our github repository page

<https://github.com/it-novum/openITCOCKPIT>

Clone the repository (replace **YOUR\_ACCOUNT\_NAME**)

```
git clone -b development https://github.com/YOUR_ACCOUNT_NAME/openITCOCKPIT
cd openITCOCKPIT/
```

**IMPORTANT! Please make sure to clone the *development* branch!**

## 2. Go to the *locales* folder

```
cd resources/locales
```

## 3. Choose a language you want to translate to

### Step 3.1: The language folder does not exist yet

Copy the template folder *en\_US* and rename it like your chosen language.

e.g.

```
cp -r en_US de_DE
```

If it is done continue with step 3.2

### Step 3.2: The language folder already exists

e.g.

```
cd de_DE
```

## 4. Translate

Translate the content of *\*.po* files (e.g. *default.po*)

The translation must be done line by line.

The *msgid* contains the original english content and should not be touched!

The translated message goes to *msgstr*.

**If you haven't done an openITCOCKPIT translation yet, please read the [Important notes](#) section!**

Example:

```
msgid "Week"
msgstr "Woche"
```

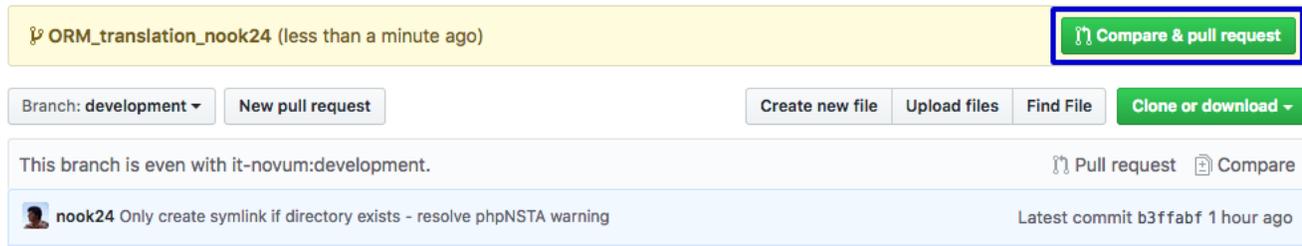
## 5. Commit, push & create PR

Replace **YOUR\_ACCOUNT\_NAME** with your GitHub account name.

```
git checkout -b development_translation_YOUR_ACCOUNT_NAME
git add .
git commit -m "Added translation for de_DE"
git push origin development_translation_YOUR_ACCOUNT_NAME
```

To contribute your changes, you need to send a pull request:

Your recently pushed branches:



The screenshot shows the GitHub interface for a pull request. At the top, a yellow bar highlights the branch `ORM_translation_nook24` (less than a minute ago) with a green button labeled "Compare & pull request". Below this, there are buttons for "Branch: development", "New pull request", "Create new file", "Upload files", "Find File", and "Clone or download". A message states "This branch is even with it-novum:development." with links for "Pull request" and "Compare". At the bottom, a commit message is shown: "nook24 Only create symlink if directory exists - resolve phpNSTA warning" with the latest commit hash `b3ffabf` from 1 hour ago.

Please notice: You need to select **development** as base branch while you create your pull request.

## Important notes

Don't translate `en_US`, that's our source language :)

- Special words are not to translate
  - `openITCOCKPIT`, `Statusengine`, `CakePHP`, `Gearman`, `GearmanWorker`
  - `LDAP`, `OpenLDAP`, `PHP`, `SSO`, `NRPE`, `NDO`, `UUID`, `ARO`, `ACO`, `ACL`,
  - `Nagios`, `Naemon`, `NPCE`, `NDOUtils`
  - .....
- Don't touch placeholders, escape characters, classes, values or html (style) code
  - `%` prefixed characters are used for replacements. (e.g. `%s` replaces a string or `%1$s`)
  - `\` escapes "
  - `CrudAuthorize::authorize()` is an example for an internal class. (Needed for error messages)
  - `$` prefixed words are variables (e.g. `$PATH`, `$USER1$`, `$ARG1$`)
  - use `<b>This is bold</b>` to generate bold characters
  - use `<success>Green text</success>` to generate green characters
- Keep length of strings that are using blank characters to get an uniform length

```
msgid "User:          %s"
msgstr "Benutzer:     %s"
```

# Need help?

Join [#openitcockpit](#) on [freenode.net](#) for any questions