

Multi-Language Support

The application is prepared for multi-language support using Django’s internationalization hooks `i18n`. The default language is English; a German translation of the entire task is included as an example. In which language the app is being displayed is determined by the `LANGUAGE_CODE` setting in `settings.py` of your *oTree* project directory.

Translations in Django templates use two different template tags. In order to provide any template file access to these tags, `{% load i18n %}` is put in the header of all `.html` files. The `{% trans %}` template tag translates either a constant string or variable content. The `{% blocktrans %}` tag allows to translate strings with variables at the same time. Any string literals marked by either of these tags can be translated within separate language files afterwards.

Note that the commands used by the internationalization for creating and compiling language files require commands from the gettext toolset. How to install gettext utilities on Windows is documented by the [Django Project](#).

To create (additional) translations, first, a so-called message file (indicated by the file ending `.po`) needs to be created. In the *oTree* Launcher, click on the console and change the directory to the respective app using `cd [appname]`.

To create or update a message file, create an empty directory named “locale” by running `mkdir locale` and then run `django-admin makemessages -l de` where `de` is the locale name for the message file to be created. The resulting `.po` files are plain text files that can be modified in any editor. However, gettext allows to translate the message files in special translation editors such as [Poedit](#). After your message files are completely translated, the translation needs to be compiled via the command `django-admin compilemessages`.

For detailed information on Django’s internationalization and translation hooks, refer to [Django’s Translation Docs](#).

Please note that any changes in the English version should not be conducted in the `.po`-file(s) but rather in `models.py`/`views.py` or the particular `.html`-templates, respectively.