
kitovu

Release 0.1.0

Mar 07, 2019

Contents

1	Overview	1
1.1	Installation	1
1.2	Documentation	1
1.3	Development	1
2	Installation	3
3	Usage	5
3.1	Verwendung über die Kommandozeile	5
3.2	Verwendung der grafischen Oberfläche	6
3.3	Konfiguration von kitovu	6
3.4	Synchronisation	7
3.5	Der FileCache	8
4	Create a Plugin	9
4.1	Plugin Class	9
4.2	User Output	10
5	Reference	11
5.1	Abstract Sync Plugin	11
5.2	Errors	11
5.3	SMB Plugin	11
6	Contributing	13
6.1	Bug reports	13
6.2	Code contributions	13
6.3	Documentation	13
7	Authors	15
8	Changelog	17
8.1	0.0.0 (2018-03-13)	17
9	Indices and tables	19

CHAPTER 1

Overview

docs	
tests	
package	

OpenHSR Connect 2

- Free software: GNU GPL v3

1.1 Installation

```
pip install kitovu
```

1.2 Documentation

<https://kitovu.readthedocs.io/>

1.3 Development

To run the all tests run:

tox

CHAPTER 2

Installation

At the command line:

```
pip install kitovu
```

Important: Kitovu addresses a primarily German-speaking audience, so we decided to write the following usage instructions in German first. An English translation will follow.

3.1 Verwendung über die Kommandozeile

Du kannst kitovu sowohl mit einer grafischen Oberfläche verwenden als auch über die Kommandozeile. Bevor du kitovu verwenden kannst, musst du jedoch die Konfigurationsdatei erstellen, siehe weiter unten.

Die Kommandozeile bietet folgende Optionen:

- `kitovu --help` zeigt alle verfügbaren Optionen an
- `kitovu [command] --help` zeigt die für den spezifischen Befehl alle verfügbaren Optionen an
- `kitovu gui` startet die grafische Oberfläche
- `kitovu sync` startet die Synchronisation mit der von dir gewählten Konfiguration.
- `kitovu validate` prüft, ob deine gewählte Konfiguration korrekt ist.
- `kitovu fileinfo` sagt dir, wo kitovu zwei wichtige Dateien speichert, die Konfigurationsdatei und der FileCache.
- `kitovu edit` öffnet die Konfigurationsdatei in einem Editor. Dieser kann mit `--editor [EDITOR_NAME]` oder über die Umgebungsvariable `EDITOR` angegeben werden. Ansonsten sucht kitovu nach einem gängigen Editor.
- `kitovu docs` öffnet die Dokumentation von Kitovu in einem Webbrowser.

3.2 Verwendung der grafischen Oberfläche

Unter “Konfiguration verwalten” stellst du ein, wo und wie kitovu Dateien synchronisiert. Wie die Konfiguration genau aufgebaut ist, liest du weiter unten.

Mit einem Klick auf “Dateien synchronisieren” lädst du alle benötigten Dateien herunter.

Important: Dateien können sich zwischen zwei Synchronisationszeitpunkten ändern, etwa, weil du noch selber Notizen hinzugefügt hast. In diesem Fall werden sie beim derzeitigen Entwicklungsstandpunkt von kitovu mit der Server-Version überschrieben.

3.3 Konfiguration von kitovu

Um überhaupt Dateien synchronisieren zu können, musst du zuerst festlegen, über welche Verbindungen das geschieht. Dies legst du in der Konfigurationsdatei fest. Je nach Betriebssystem liegt diese an einem anderen Ort. Du kannst diese Datei direkt mit `kitovu edit` bearbeiten. Alternativ siehst du, wo diese Datei gespeichert ist, indem du `kitovu fileinfo` auf der Kommandozeile eingibst. Die Datei kannst du in jedem beliebigen Texteditor konfigurieren.

Important: Achte auf die korrekten Einrückungen, ansonsten ist die Konfiguration fehlerhaft!

So sieht eine Beispielkonfiguration aus:

```
root-dir: ~/kitovu
connections:
  - name: skripte
    plugin: smb
    username: legger
  - name: moodleskripte
    plugin: moodle

subjects:
  - name: EPJ
    sources:
      - connection: skripte
        remote-dir: Informatik/Fachbereich/Engineering-Projekt/EPJ
  - name: Betriebssysteme2
    sources:
      - connection: moodleskripte
        remote-dir: "Betriebssysteme 2 FS2016"
```

Es sind keine Leerzeichen in Namen erlaubt. Möchtest du Leerzeichen verwenden, etwa für name, musst du den Namen in Gänsefüßchen setzen: `Betriebssysteme2` oder `"Betriebssysteme 2"`.

3.3.1 Aufbau der Konfigurationsdatei

`root-dir`: Das Installationsverzeichnis von kitovu.

Abschnitt connections

`connections`: Die Verbindungen zu den Plattformen, von denen du deine Unterrichtsmaterialien synchronisierst, in der Regel ist das ein SMB-Skripteserver oder Moodle.

`name`: Ein von dir frei wählbarer Name für die jeweilige Plattform.

`plugin`: Der Name der Plattform, der kitovu intern verwendet, damit eine Verbindung zustande kommt. Derzeit gibt es die fixen Bezeichnungen `smb` oder `moodle`.

`username`: Dein Login-Name, womit du dich auch andernorts an der Schule einloggst, bestehend aus Vor- und Nachname. Moodle benötigt keinen Usernamen.

Abschnitt subjects

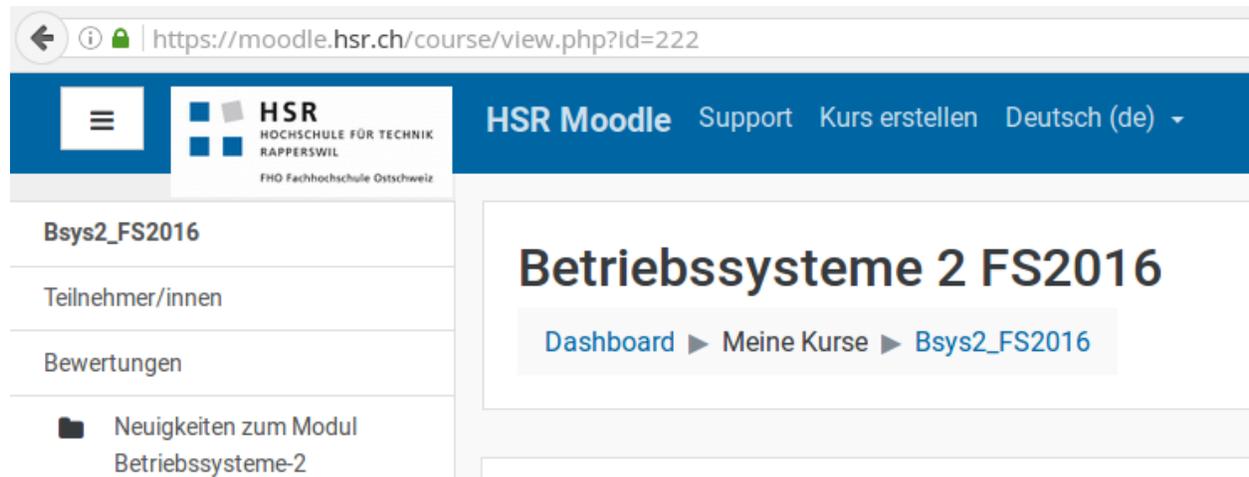
`subjects` bezeichnet die Unterrichtsmodule, die du an der Schule belegst, beispielsweise “Betriebssysteme 1” oder “Objektorientierte Programmierung in Java”.

`name`: Ein von dir frei wählbarer Name für das jeweilige Unterrichtsmodul.

`sources`: Die Plattformen, von denen du die Unterrichtsmaterialien synchronisierst. Gewisse Unterrichtsmodule sind sowohl auf Moodle als auch auf dem Skripteserver zu finden - hier kannst du für jede Plattform einen separaten Eintrag erstellen.

`connection`: Der Name der Plattform, den du weiter oben unter `connections` bzw. dort als `name` festgelegt hast.

`remote-dir`: Das Verzeichnis, das synchronisiert werden soll. In Moodle entspricht dies dem Titel, den der Kurs im Browser trägt. Im folgenden Beispiel ist das “Betriebssysteme 2 FS2016”:



The screenshot shows a web browser window with the URL `https://moodle.hsr.ch/course/view.php?id=222`. The page header includes the HSR logo (Hochschule für Technik Rapperswil, FHO Fachhochschule Ostschweiz) and navigation links for 'Support', 'Kurs erstellen', and 'Deutsch (de)'. The main content area displays the course title 'Betriebssysteme 2 FS2016' and a breadcrumb trail: 'Dashboard ► Meine Kurse ► Bsys2_FS2016'. On the left sidebar, there are links for 'Bsys2_FS2016', 'Teilnehmer/innen', 'Bewertungen', and 'Neuigkeiten zum Modul Betriebssysteme-2'.

3.4 Synchronisation

Bei der ersten Synchronisation fragt dich kitovu nach allen benötigten Passwörtern. Für den Skripteserver ist dies das Passwort, das du für alle anderen Dienste an der Schule verwendest.

Moodle hingegen ist ein Spezialfall. Um dich erfolgreich mit Moodle verbinden zu können, fragt dich kitovu bei der Erstverwendung nach dem “Sicherheitsschlüssel”. Den findest du folgendermassen:

1. Logge dich via Browser auf Moodle ein.

2. Navigiere zu den Einstellungen: klicke auf das Profilfoto oben rechts und wähle Einstellungen, dann Sicherheitsschlüssel.
3. Kopiere den Eintrag unter “Schlüssel”, der mit dem Servie “Moodle mobile web service” beschrieben wird. Das ist eine lange Kette von Zeichen und Zahlen. **Gib auf keinen Fall diesen Sicherheitsschlüssel weiter und behalte ihn für dich!**
4. Füge den Sicherheitsschlüssel in kitovu ein, wenn dich das Programm danach fragt.

3.5 Der FileCache

Wenn du Dateien synchronisierst, hält kitovu das in einer Datei fest. Nach Ende eines Semesters bzw. nach Prüfungsende kannst du diese Datei wieder löschen - also wenn keines der Unterrichtsmodule des vergangenen Semesters mehr synchronisiert werden sollte. Du siehst, wo diese Datei gespeichert ist, indem du `kitovu fileinfo` auf der Kommandozeile eingibst.

4.1 Plugin Class

4.1.1 Implementation

To create a plugin that is able to connect to another service you have to first create a plugin class. This custom plugin class needs to inherit from `kitovu.sync.syncplugin.AbstractSyncPlugin`.

For an example implementation see `kitovu.sync.plugin.smb`.

4.1.2 Registration

The plugin class needs to be registered with `stevedore`.

You need to create a `setup.py` file like this:

```
from setuptools import setup

setup(
    # other setup parameters

    entry_points={
        'kitovu.sync.plugin': [
            'my-example = my.plugin:ExamplePlugin',
        ],
    },

    # other setup parameters
)
```

By setting `entry_points` you can list all plugins you implement in this package.

my-example This is the name of the plugin used in the kitovu configuration.

my.plugin:ExamplePlugin This is the namespace and class of the plugin to use.

For further information see the stevedore documentation for [creating a plugin](#).

4.2 User Output

4.2.1 Errors

For errors happening in your plugin, you can raise `kitovu.utils.PluginOperationError`. If raised during `configure`, `connect` or `list_path`, the GUI/CLI shows an error and plugin is skipped entirely.

When raised in `create_remote_digest`, `create_local_digest` or `retrieve_file`, only the affected file is skipped.

Any other exception will lead to the kitovu application to terminate.

4.2.2 Warnings

To print warnings to the user without aborting the current operation, use Python's [logging framework](#). It's recommended to use `logger = logging.getLogger(__name__)` to get an unique logger for your plugin.

5.1 Abstract Sync Plugin

5.2 Errors

5.3 SMB Plugin

6.1 Bug reports

When reporting a bug please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

6.2 Code contributions

Since kitovu is currently developed as an university project, we're unable to accept any outside code contributions. This will change around July 2018.

6.3 Documentation

If you want to generate the entire documentation of the project you can create it with:

```
sphinx-apidoc --separate --no-toc --force -o docs/reference/complete src
tox -e docs
```


CHAPTER 7

Authors

- Méline Sieber, Nicolas Ganz, Florian Bruhin - <https://github.com/kitovu-bot/kitovu>

CHAPTER 8

Changelog

8.1 0.0.0 (2018-03-13)

- First release on PyPI. Does not contain any code yet.

CHAPTER 9

Indices and tables

- [genindex](#)
- [modindex](#)
- [search](#)