

Requirements

You need Java 8 installed on your computer. You can download it [here](#).

Launching in UI mode

UI mode is set as default. Just double click on the Gipter.jar and that's it. If you have any problems then use the right click option and pick 'Open with' then choose 'Java(TM) Platform SE binary'.

Launching in command line (CLI) mode

Open PowerShell, go to Gipter home and the use this command:

```
java -jar Gipter.jar useUI=N
```

or create a file with extension *.cmd and copy paste the above instruction to it. Place the file in the Gipter home directory and double-click it. You will launch the command window in which Gipter will be executed.

Description of UI

All parameters are described [here](#).

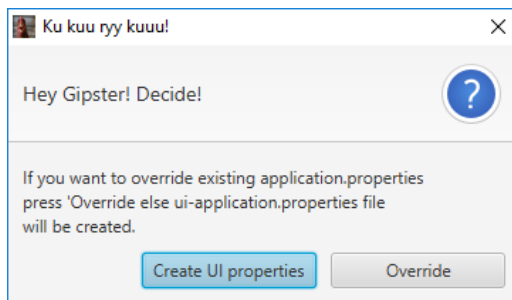
The screenshot shows the Gipter v3.0 application window. It is divided into several sections for configuration:

- CSV details:** Includes fields for Authors (Pawel Gawedzki, PreCyz), Committer email, Git author, Mercurial author, SVN author, Skip remote (checkbox), and Code protection (dropdown set to NONE).
- Launch details:** Includes Preferred arg source (dropdown set to UI), Confirmation window (checkbox checked), Use UI (checkbox checked), Tray active (checkbox checked), and a Save current configuration button.
- Time frame:** Includes Start date (2019-03-03), End date (2019-03-10), and Period in days (7).
- Toolkit details:** Includes Username (PAWG), Password (masked with dots), Domain (NCDMZ), List name (WorkItems), Url (https://goto.netcompany.com/cases/GTE106/NCSCOPY), WS Url (https://goto.netcompany.com/cases/GTE106/NCSCOPY/_vti_bin/lists.asmx), and User folder (https://goto.netcompany.com/cases/GTE106/NCSCOPY/Lists/WorkItems/PAWG).
- Paths details:** Includes File name prefix, Project paths (C:\Work\workspace\GitDiffGenerator), and Item path (c:\Work\workspace\diff), each with a Change button.

At the bottom, there is a language dropdown set to 'en', an Execute button, a Deamon button, and an Exit button.

This is the reflection of all parameters that are defined in Gipter application. You should know how to set them. What's important here is toolkit credentials. Use NCDMZ credentials. Now here is a short description of buttons:

Save – saving current settings. You will decide if existing '*application.properties*' should be overridden or '*ui-application.properties*' should be created. To decide you will see below window



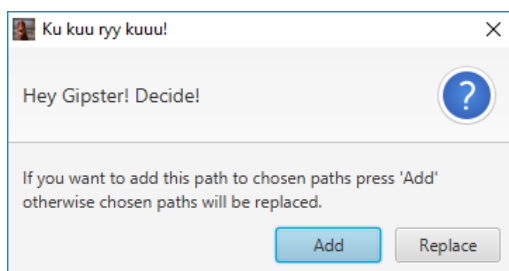
Why? Because *application.properties* is a source of truth for CLI mode. And from some reason maybe you would like to distinguish parameters of UI mode from CLI mode.

Execute – executes diff generation and upload to toolkit with parameters set in UI.

Deamon – minimize application to tray and saves current UI parameters to *ui-application.properties*.

If you choose '*No*' nothing will happen. If '*Yes*' then window with job details will be displayed, but this is described later.

Change – for '*Project paths*' allows you to pick up all projects, that you want to combine and use as copyright item. You can either replace previously chosen or add a new one. To do so Gipster will ask you to decide with the below window:



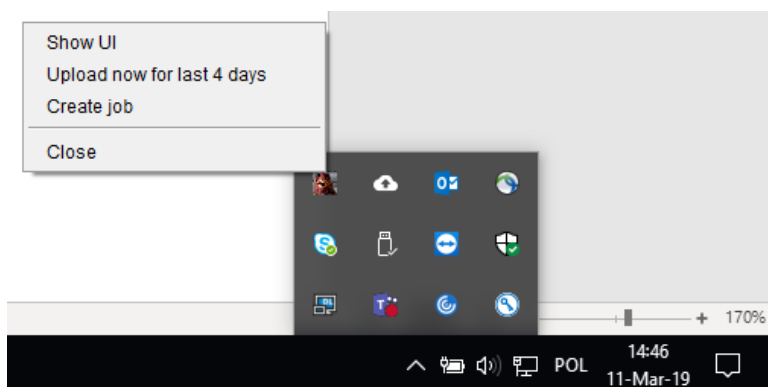
Change – for '*Item path*' allows you to choose where to store your copyright items.

Exit – terminates the program.

You can also change the language to Polish.

Tray description

When you look at the tray you will see a new icon there like below:



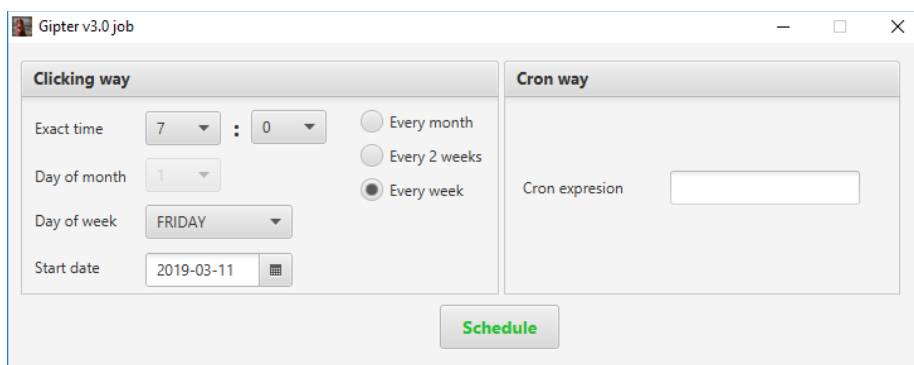
Yes, yes! It's a chicken 😊 When you right-click that chicken then you will see the menu. Below is a short description of all available options.

Show UI – brings back the main window with application parameters.

Upload now for the last {number-of-days} days—uploads copyright item for the last amount of days. Number-of-days is equal to *'period-in-days'* from the main window. This value can be updated dynamically. You just need to change the value of period-in-days on the main window and either save current configuration to ui-application properties or press Deamon to minimize the main window to the tray icon.

Close – terminates the program.

Create job – launches the window where you can setup the gipter-job and schedule it. Below is a screenshot with that window:



You can define gipter-job both ways: by clicking in predefined values or specifying CRON expression. CRON expression is well known but if you want to know more read [this](#). If you need with building the CRON expression go [here](#).

If you choose to click the way to define the job then:

Exact time – is an hour of the day when the job must be executed.

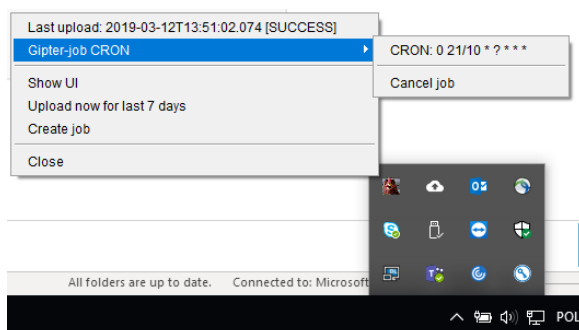
Day of the month – the exact day of the month to execute the job.

Day of week – the day of the week when to execute the job.

Start date – when to schedule the job. At this date above job definition will start to be valid.

Radio buttons are self-explanatory (I think).

Schedule – will create gipter-job and put it into quartz scheduler. Once the job is created you will the difference in the tray:



In tray area now you will see the menu with gipter-job type and details. Also, you can **Cancel job** that job. If you do that this section will disappear from the tray.

You might notice that there is one additional entry on the very top of the tray. This is the info when the last upload was executed and if it was success or failure.

The important thing here is that this job will work only then **when Gipter is working**. If you close the application, then the job will be deleted.

That's it! Enjoy 😊 (I hope).