

This tutorial contains the instructions on how to use the Robidium. It is preferable to use the web version of the tool available at <http://robidium.cloud.ut.ee/>. However, if you prefer to use the desktop version, the tutorial also provides instructions on how to setup the tool locally.

SETUP

Download the source code:

Back-end application – <https://github.com/volodymyrLeno/Robidium>

Front-end application – <https://github.com/stdevi/robidium-frontend>

To discover data transformations, the tool requires Ubuntu terminal that can be downloaded from the Microsoft Store:



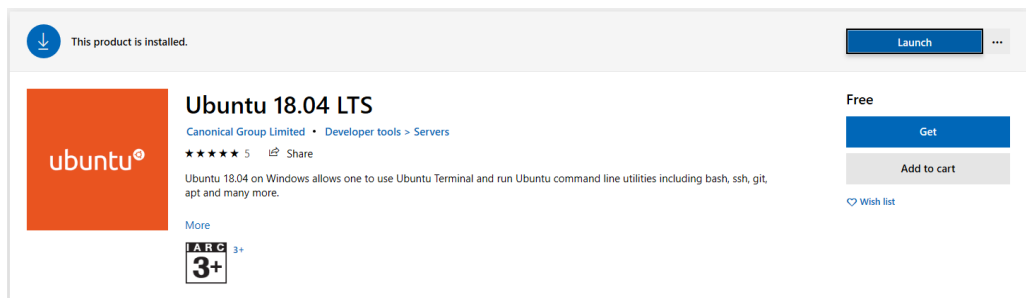
Before installing any Linux distributions on Windows, you must enable the "Windows Subsystem for Linux" optional feature.

For this, open PowerShell as Administrator and run command:

```
dism.exe /online /enable-feature /featurename:Microsoft-Windows-Subsystem-Linux /all /norestart  
dism.exe /online /enable-feature /featurename:VirtualMachinePlatform /all /norestart
```

You will have to restart the system for all the changes to be applied

After Ubuntu terminal is downloaded open it by clicking button "Launch".



Wait for the installation to finish. You will have to set the username and password.

```
vleno@DESKTOP-42GR8SQ: ~  
Installing, this may take a few minutes...  
Please create a default UNIX user account. The username does not need to match your Windows username.  
For more information visit: https://aka.ms/wslusers  
Enter new UNIX username: vleno  
Enter new UNIX password:  
Retype new UNIX password:  
passwd: password updated successfully  
Installation successful!  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.
```

In Ubuntu terminal execute the following commands:

```
sudo apt update  
sudo apt upgrade  
sudo apt install python2.7  
sudo apt install python-pip  
python -m pip install -U pip setuptools  
sudo apt-get install libboost-all-dev
```

Install all required python modules:

```
python -m pip install numpy  
python -m pip install tabulate  
python -m pip install cherrypy  
python -m pip install editdistance  
python -m pip install python-Levenshtein  
python -m pip install matplotlib
```

Navigate to */foofah* directory in the source code of the tool via Ubuntu terminal:

```
vleno@DESKTOP-42GR8SQ:/$ cd mnt  
vleno@DESKTOP-42GR8SQ:/mnt$ cd c/Volodymyr/Robidium/foofah  
vleno@DESKTOP-42GR8SQ:/mnt/c/Volodymyr/Robidium/foofah$
```

Run command:

```
sudo python setup.py install
```

Install Node.js – <https://nodejs.org/en/>

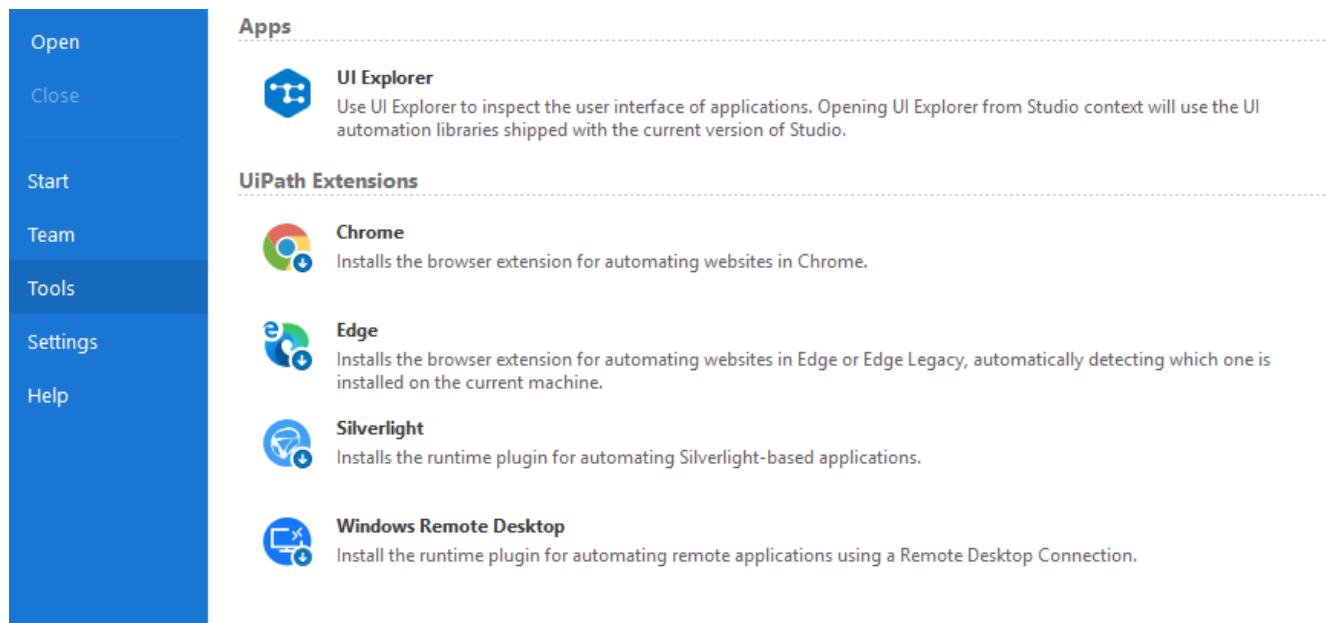
Navigate to */front-end* directory and run the next commands:

```
npm install
```

```
npm install -g create-react-app
```

```
npm install react-scripts
```

Open UIPath client. Navigate to “Tools” and select Chrome to install the corresponding extension.



The screenshot shows the UIPath client interface. On the left is a blue sidebar with a menu containing: Open, Close, Start, Team, Tools (highlighted), Settings, and Help. The main area is titled 'Apps' and contains a section for 'UiPath Extensions'. This section lists four extensions with their respective icons and descriptions:

- UI Explorer**: Use UI Explorer to inspect the user interface of applications. Opening UI Explorer from Studio context will use the UI automation libraries shipped with the current version of Studio.
- Chrome**: Installs the browser extension for automating websites in Chrome.
- Edge**: Installs the browser extension for automating websites in Edge or Edge Legacy, automatically detecting which one is installed on the current machine.
- Silverlight**: Installs the runtime plugin for automating Silverlight-based applications.
- Windows Remote Desktop**: Install the runtime plugin for automating remote applications using a Remote Desktop Connection.

USAGE

- 1) To run the front-end:

Using command line or PowerShell navigate to \front-end directory and run “npm start”

- 2) To run the back-end:

In PowerShell navigate to the main directory of the tool and run “./mvnw spring-boot:run”

- 3) The tool is available at <http://localhost:3000>

- 4) Upload the UI log. Currently the tool supports only the logs recorded by the Action Logger (https://github.com/volodymyrLeno/RPA_UILogger). The minimum requirements for the UI log are:

- The log should be in a CSV format
- There should be a single timestamp attribute, which should be called *timeStamp*
- If the log is segmented, the segment ID should be in a column called *caseID*. If the log is unsegmented this requirement is not applicable
- The log should contain the information about the user and the application where the recorded action took place, saved in the *userID* and *targetApp* attributes, respectively
- The performed action (e.g., copyCell, editField) should be stored in the attribute *eventType*
- The log should contain all context attributes that capture the concept of step 6
- The order of the attributes in the log should be the following: caseID (if applicable), timeStamp, userID, targetApp, eventType, other attributes.

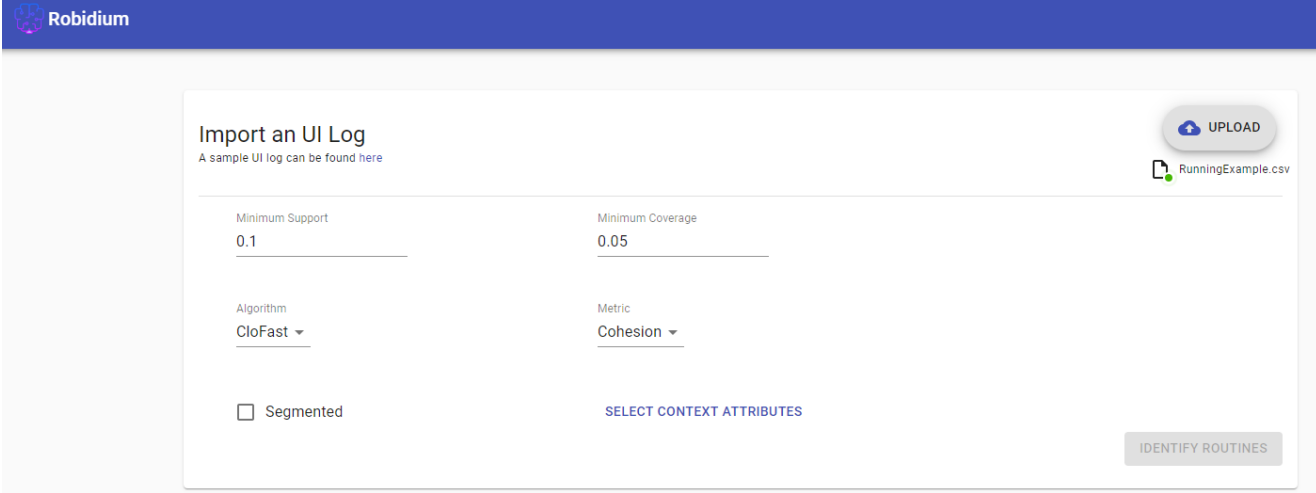
Here is an example of the UI log:

	A	B	C	D	E	F	G	H
1	timeStamp	userID	targetApp	eventType	content	target.id	target.name	target.value
2	2019-06-11T05:17:24.058Z	vleno	Excel	copyCell	John Doe	A2		John Doe
3	2019-06-11T05:17:26.839Z	vleno	Chrome	paste	John Doe		Name_First	
4	2019-06-11T05:17:28.384Z	vleno	Chrome	editField			Name_First	John
5	2019-06-11T05:17:28.783Z	vleno	Chrome	paste	John Doe		Name_Last	
6	2019-06-11T05:17:31.072Z	vleno	Chrome	editField			Name_Last	Doe
7	2019-06-11T05:17:35.817Z	vleno	Excel	copyCell	11/03/1986	B2		11/03/1986
8	2019-06-11T05:17:38.911Z	vleno	Chrome	paste	11/03/1986	Date-date	date	
9	2019-06-11T05:17:39.661Z	vleno	Chrome	editField		Date-date	date	11/03/1986

- 5) Specify the parameters for the UI log mining algorithms:

- Minimum support (used to discover only frequently repetitive routines, a float number [0, 1]) – minimum ratio of cases that contain the routine
- Minimum coverage (used to filter out short routines with low frequency, a float number [0, 1]) – minimum ratio of the log covered by the routine
- Algorithm (sequence pattern mining algorithm to extract frequent routines, currently BIDE+ and CloFast algorithms are supported)

- Metric (specifies routine selection strategy). Depending on the metric, different routines (e.g., the longest ones, the most frequent ones) will be selected. Currently Robidium supports the following metrics: length, coverage, cohesion and frequency.
- Segmented (specifies whether the given log is already segmented, or it requires segmentation first)



Robidium

Import an UI Log

[A sample UI log can be found here](#)

[RunningExample.csv](#)

Minimum Support:

Minimum Coverage:

Algorithm:

Metric:

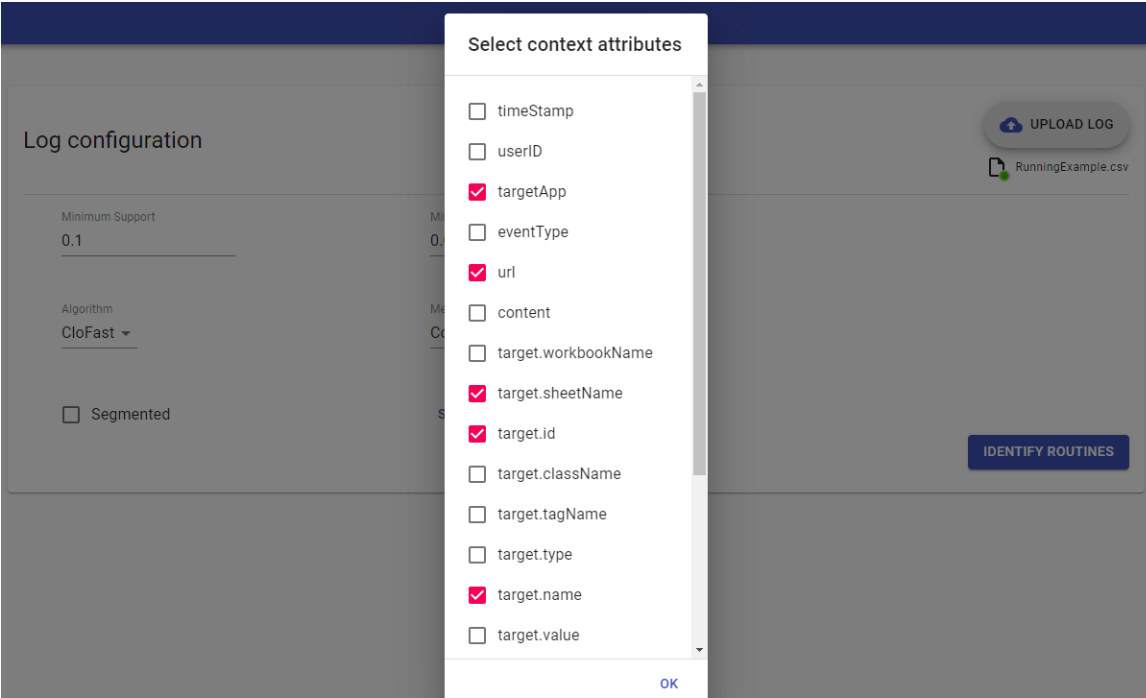
☐ Segmented

[SELECT CONTEXT ATTRIBUTES](#)

[IDENTIFY ROUTINES](#)

[UPLOAD](#)

6) Select context attributes:



Log configuration

Minimum Support:

Algorithm:

☐ Segmented

Select context attributes

- ☐ timeStamp
- ☐ userID
- ☒ targetApp
- ☐ eventType
- ☒ url
- ☐ content
- ☐ target.workbookName
- ☒ target.sheetName
- ☒ target.id
- ☐ target.className
- ☐ target.tagName
- ☐ target.type
- ☒ target.name
- ☐ target.value

OK

[UPLOAD LOG](#)

[RunningExample.csv](#)

[IDENTIFY ROUTINES](#)

- 7) Click button *Identify Routines* to extract automatable routines from the log. Discovered routines will be available under Routine selection

Routine selection

<> GENERATE SCRIPT

☐ Routine 1

☐ Routine 2

☐ Routine 3

☐ Routine 4

- 8) You can check the discovered routines by expanding them

☐ Routine 4

1	copyCell	Excel	Sheet1	A
2	paste	Chrome	SingleLine	https://forms....
3	editField	Chrome	SingleLine	https://forms....
4	copyCell	Excel	Sheet1	B
5	paste	Chrome	SingleLine	https://forms....
6	editField	Chrome	SingleLine	https://forms....
7	copyCell	Excel	Sheet1	C
8	paste	Chrome	Date-date	https://forms....
9	editField	Chrome	Date-date	https://forms....
10	copyCell	Excel	Sheet1	D

- 9) Select the routine to be automated by clicking the corresponding radio button and click the button *Generate Script*.

Routine selection

☒ Routine 1

1	copyCell	Excel	Sheet1	A
2	paste	Chrome	SingleLine	https://forms...
3	editField	Chrome	SingleLine	https://forms...
4	copyCell	Excel	Sheet1	B
5	paste	Chrome	SingleLine	https://forms...
6	editField	Chrome	SingleLine	https://forms...

GENERATE SCRIPT

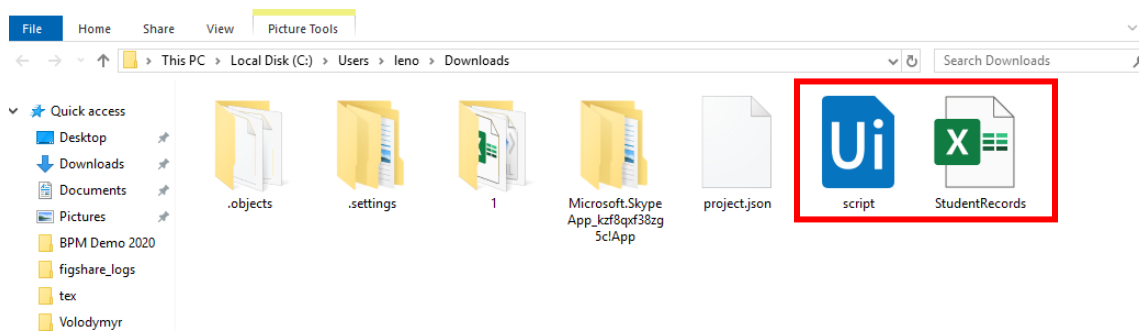
- 10) After the script was generated, a notification will pop up with the option to save the script on your machine. Click button *Save*

Save Script

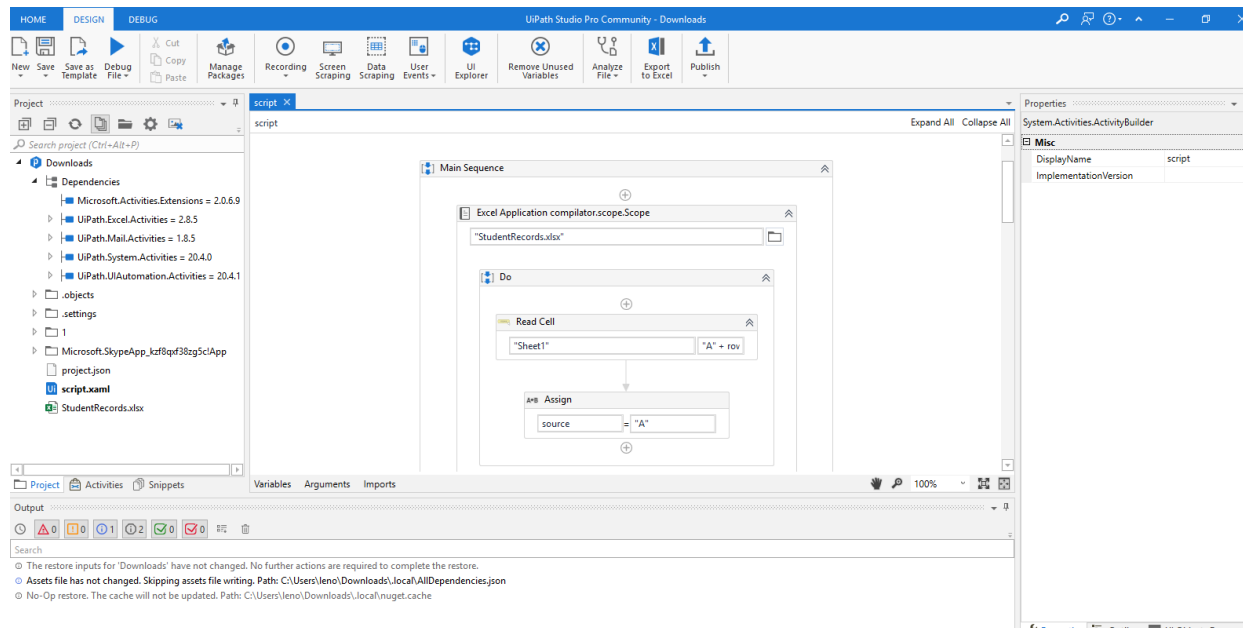
To save a UiPath script, please press "Save" button and select a directory.

CANCEL **SAVE**

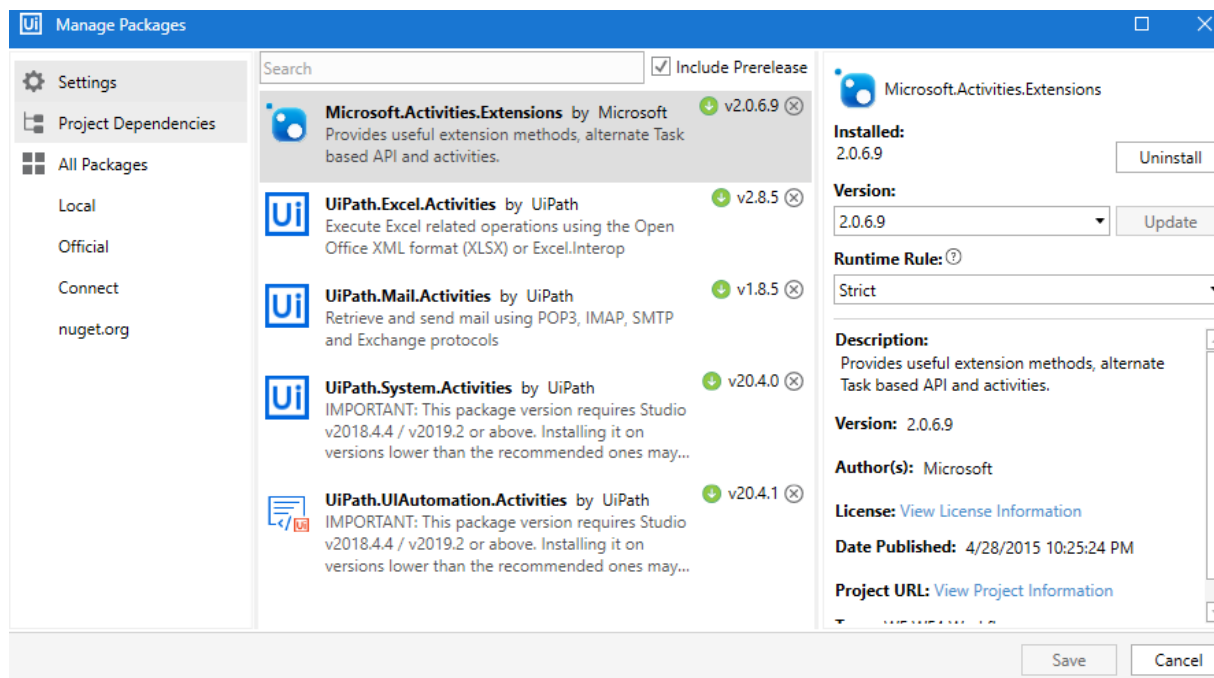
- 11) If the automated routine works with a spreadsheet, place the generated script and the corresponding spreadsheet in the same directory.



12) Open the script using UiPath client



13) Select Manage Packages and install Microft.Activities.Extensions



14) Set the input parameter for the script. In our example, it will be a row of a spreadsheet, that corresponds to a particular student entry

Name	Direction	Argument type	Default value
row	In	String	"2"
column	In	String	Enter a VB expression

Create Argument

Variables Arguments Imports

100%

15) Run the script by clicking the *Debug File* button