
Pext Documentation

Release 0.35

Sylvia van Os

Dec 07, 2022

GENERAL INFORMATION:

1	Installation	5
1.1	Windows	5
1.2	macOS	5
1.3	GNU/Linux	5
2	Compiling from source	7
2.1	Windows	7
2.2	macOS	7
2.3	GNU/Linux	8
3	Metadata	15
3.1	Fields	15
3.2	Settings	15
4	Requirements	17
4.1	Example	17
5	API	19
5.1	Base	19
5.2	Helpers	20
6	pext_dev	21
7	Theme	23
7.1	File format	23
8	Indices and tables	25

Pext

```
<a href="https://repology.org/metapackage/pext/versions">
  
```

```
</a>
```

```
![Lilly the leoger](/assets/logo.png)
```

```
![REUSE status](https://api.reuse.software/badge/github.com/Pext/Pext){}(https://api.reuse.software/info/
github.com/Pext/Pext) ![Linux & macOS Build Status](https://github.com/Pext/Pext/actions/workflows/
main.yml/badge.svg){}(https://github.com/Pext/Pext/actions/workflows/main.yml) ![Windows Build sta-
tus](https://ci.appveyor.com/api/projects/status/730aa4x1spa5vumx/branch/master?svg=true){}(https://ci.
appveyor.com/project/TheLastProject/pext/branch/master) ![ReadTheDocs](https://readthedocs.org/projects/
pext/badge/?version=latest){}(https://pext.readthedocs.io/en/latest/?badge=latest) ![Translation status](https:
//hosted.weblate.org/widgets/pext/-/svg-badge.svg){}(https://hosted.weblate.org/engage/pext/?utm_source=widget)

![Matrix](https://img.shields.io/matrix/pext:matrix.org.svg){}(https://riot.im/app/#/room/#pext:matrix.org)
```

Contents

- [Introduction](#introduction)
- [How it works](#how-it-works)
- [Installation](#installation)
- [Usage](#usage)
- [Hotkeys](#hotkeys)
- [Community](#community)
- [License](#license)

Introduction

Pext stands for **P**ython-based **e**x**tensible **t**ool. It is built using Python 3 and Qt5 QML and has its behaviour decided by modules. Pext provides a simple window with a search bar, allowing modules to define what data is shown and how it is manipulated.

For example, say you want to use Pext as a password manager. You load in the pass module, and it will show you a list of your passwords which you can filter with the search bar. When you select a password in the list, it will copy the password to your clipboard and Pext will hide itself, waiting for you to ask for it again.

Depending on the module you choose, what entries are shown and what happens when you select an entry changes. So choose the module appropriate for what you want to do, and Pext makes it easy.

Several modules are available for effortless install right within Pext.

```
![Pext Introduction](/assets/pext_intro.gif)
```

How it works

Pext is designed to quickly pop up and get out of your way as soon as you're done with something. It is recommended to bind Pext to some global hotkey, or possibly run multiple instances of it with different profiles under multiple hotkeys. Example Pext workflows look as follows:

```
![Pext workflow graph](/assets/workflow_graph.png)
```

Simply put:

- Open (Pext)
- Search (for something)

- Select (with Enter)
- Hide (automatically)

Installation

Check the install instructions at [Read the Docs](<https://pext.readthedocs.io/en/latest/installation.html>).

Usage

To actually use Pext, you will first have to install one or more modules. Check out the Pext organisation on [GitHub](<https://github.com/Pext>) or use *Module -> Install module -> From online module list* in the application for a list of modules.

After installing at least one module, you can load it from the *Module -> Load module* menu. After that, experiment! Each module is different.

For command line options, use *-help*.

Hotkeys

Entry management

- Escape: Go one level up
- Shift+Escape: Go up to top level and trigger minimize
- Tab: Tab-complete the current input
- Enter / Left mouse button: Select entry or run command
- Shift+Enter: Select entry or run command but explicitly disable minimizing
- Ctrl+Enter: Run command with arguments
- Ctrl+. / Right mouse button on any item: Open context menu
- Ctrl+J / Down arrow: Go one entry down
- Ctrl+H / Up arrow: Go one entry up
- Ctrl+F / Page down: Go one page down
- Ctrl+B / Page up: Go one page up

Tab management

- Ctrl+T: Open new tab
- Ctrl+W: Close current tab
- Ctrl+Tab: Switch to next tab
- Ctrl+Shift+Tab: Switch to previous tab
- Alt+`<number>`: Switch to tab <number>
- F5: Reload tab, including code changes to the module

Session management

- Ctrl+Q: Quit

Community

If you need support or just want to chat with our community, we have the following options:

- Matrix: #pext:matrix.org ([webchat](<https://riot.im/app/#/room/#pext:matrix.org>))
- Telegram: [@PextTool](<https://t.me/PextTool>)

Both channels are linked to each other, so there is no need to worry about missing out.

License

Pext is licensed under the [GNU GPLv3+](LICENSES/GPL-3.0-or-later.txt), with exception of artwork and documentation, which are licensed under the [Creative Commons Attribution Share-Alike 4.0 license](LICENSES/CC-BY-SA-4.0.txt).

Under artwork and documentation fall:

- All files in the following directories: - assets/ - docs/ - pext/images/ - .github/
- All Markdown files in the root directory.

When attributing the logo (which was donated by [vaeringjar](https://notabug.org/vaeringjar)), it should be attributed as Lilly the leoger by White Paper Fox. Alternatively, it may be referred to as the Pext logo. Please link to Pext with <<https://github.com/Pext/Pext>> or <<https://pext.io/>> and to White Paper Fox with <<http://www.whitepaperfox.com/>> where possible.

INSTALLATION

Please note that these instructions are for normal installation. If you instead want to help develop, look into [compiling from source](#).

1.1 Windows

A Windows installer is available [in the releases section on GitHub](#).

1.2 macOS

A macOS .dmg file is available [in the releases section on GitHub](#).

Alternatively, you may install Pext through pip:

```
pip install pext
```

If you are unable to run Pext due to Gatekeeper settings, run the following command to allow running programs from any source:

```
sudo spctl --master-disable
```

You may also need to go to Security & Privacy settings and select “Allow apps downloaded from: Anywhere”

1.3 GNU/Linux

1.3.1 Arch Linux

Pext is available as [pext](#) and [pext-git](#). These packages are maintained by [Agesly Danzig](#).

1.3.2 Other distros

An AppImage is available [in the releases section on GitHub](#). It can be ran in-place on pretty much any GNU/Linux distribution. Make sure to right click the AppImage and mark it as executable in properties. Then, just double click.

Alternatively, you may install Pext through pip:

```
pip3 install --user pext
```

On some systems, you may need to use pip instead of pip3.

COMPILING FROM SOURCE

Please note that these instructions are for developers. If you want to just use Pext, look into [installation](#).

2.1 Windows

2.1.1 Preparation

Assuming you have no previous python installation, either

- Use a package manager like [Chocolatey](#) to install Python 3
- Install Python 3.6 manually from [python.org](#)

Then, assuming python and pip are installed, run *pip install dulwich PyQt5 pynput* in a command window.

2.1.2 Starting Pext

Pext can be ran by ran from a command window by running one of the following commands in the place where you saved Pext to (type `cd place_you_saved_pext` to go there):

- `python -m pext` to start Pext itself
- `python -m pext_dev` to start the Pext tools for module and theme developmentAlternatively, see See [Installing Pext from source](INSTALL_FROM_SOURCE.md) (not recommended and unsupported)

2.2 macOS

2.2.1 Preparation

The following commands need to be run. If you do not have the brew command, follow the installation instructions on [Homebrew's website](#).

Before running the Install Certificates command, which is only necessary to be able to retrieve the online module list, please read <https://bugs.python.org/msg283984>.

```
brew install libnotify python3 qt5
pip3 install certifi dulwich pyqt5 urllib3
/Applications/Python\ 3.6/Install\ Certificates.command
```

2.2.2 Starting Pext

After installing the dependencies, Pext can be ran by running one of the following commands in a terminal window in the place where you saved Pext to:

- `python3 -m pext` to start Pext itself
- `python3 -m pext_dev` to start the Pext tools for module and theme development

If desired, it can also be installed using the following command:

```
pip3 install . --user --upgrade --no-deps
```

After doing this (and adding “\$HOME/Library/Python/3.6/bin” to your \$PATH), you can start Pext like any application, or use `pext` and `pext_dev` on the command line.

2.3 GNU/Linux

The following dependencies need to be installed:

Distribu- tion	Packages
Arch Linux	libnotify python-pip python-pyqt5 qt5-quickcontrols python-dulwich (AUR)
NixOS	libnotify python3Packages.pip python3Packages.dulwich python3Packages.pyqt5 qt5.qtquickcontrols
openSUSE	libnotify-tools python3-dulwich python3-pip python3-qt5

After installing the dependencies, Pext can be ran by running one of the following commands in the place where you saved Pext to:

- `python3 -m pext` to start Pext itself
- `python3 -m pext_dev` to start the Pext tools for module and theme development

If desired, it can also be installed using the following command (pip instead of pip3 on some systems):

```
pip3 install . --user --upgrade --no-deps
```

After doing this, you can start Pext like any application, or use `pext` and `pext_dev` on the command line.

Change Log

All notable changes to this project will be documented in this file.

The format is based on [Keep a Changelog](<http://keepachangelog.com/>) and this project adheres to [Semantic Versioning](<http://semver.org/>).

[0.35] - 2022-12-07 ### Fixed - Crash on launch due to ImportError

[0.34] - 2022-11-30 ### Packaging changes - Update Python to 3.11

API changes - Bump API version to 0.13.0 - Return instances of Selection class instead of dict

Fixed - Launcher icon not working

[0.32] - 2020-10-31 ### Fixed - Modules with Python dependencies could not install on Debian Buster - Fix installing modules and themes from URL and command line - Crash when failing to checkout branch

[0.31] - 2020-06-21 #### Changed - Search bar content is now unique per module - ask_input prefill now fills text instead of placeholder - Report error traceback in bug report as code block - Use git_branch_stable in metadata.json for modules and themes by default instead of forcing master

Fixed - Info menu something not showing up - Command argument handling - Can't minimize with escape during module update - Crash in forecast of weather module

[0.30] - 2020-04-17 #### Changed - Display update progress in module tab - Display crash reason in module tab - Open "Load Module" dialog after module finishes installing

Fixed - Setuptools missing from AppImage - Pext locking up when pressing "Update and reload" until update completes

[0.29] - 2020-03-07 #### Added - Shift+Escape hotkey to reset selection to base and minimize

Changed - Pext no longer deletes invalid modules from disk - Context options are now shown in the navigation tree

Fixed - Alignment of text and dismiss icon in alerts - Crash when checking for module/theme updates with no Internet connection - Pext sometimes showing null in the navigation tree

[0.28] - 2020-01-18 #### Added - Button to open module's context menu

Removed - Regex matching support

Changed - Improved hotkey reference screen with no modules loaded - Spacing between entry-specific and base context options for clarity

Fixed - Still processing module requests string containing %1 - Traditional Chinese incorrectly not being included as a selectable language - Crash when search string starts with /

[0.27] - 2019-11-15 #### Packaging changes - New dependency: [paramiko](<https://pypi.org/project/paramiko/>) - New dependency: [psutil](<https://pypi.org/project/psutil/>)

Added - Option to disable automatic module and theme update install - Ctrl+Y and Ctrl+N hotkeys to activate/dismiss top-most actionable - Ctrl+S hotkey to change the sorting mode

Changed - Enable update check by default - Use action to notify users about update checks being enabled - Don't update module while in use (could cause unexpected state) - Minimize to tray by default on non-macOS platforms - Always put exact match on top of results

Fixed - Update action being added before translations are available - Windows shortcut only being available for admin account - Sort mode not correctly saved/loaded

[0.26] - 2019-10-18 #### API changes - Bump API version to 0.12.0 - Modules can now offer a multiple choice dialog using Action.ask_choice

Added - Explanation on Shift+Return key - Support for metadata.json files to show a dropdown of choices - Support for middle mouse button causing selection without minimization - Buttons to open a new module or close the currently open module - Support for reporting bugs to modules directly

Changed - Sorting settings are now per loaded module - If there are no modules or themes, trying to load one will pop up the installation dialog - macOS now also uses Qt's Fusion theme (less native, but less glitches) - Update notifications are no longer in a dialog box - Nightly versions now check for nightly updates

Fixed - TRANSLATION MISSING: failed_to_update_dependencies - Global hotkey not working - Back button remaining grey

[0.25] - 2019-10-04 #### API changes - Bump API version to 0.11.1 so that modules can detect request handlers being fixed

Added - Shift+Return hotkey to explicitly disable minimizing and resetting when making a selection - Ability to choose separator between output entries in queue (if more than one string will be output)

Changed - Text to copy to clipboard is now queued until minimizing too - Default separator is now Enter instead of Tab - If a selection state change is requested by the module, ensure it is always done

Fixed - Module request handlers not being removed properly, causing multiple incorrect requests

[0.24.1] - 2019-09-08 #### Changed - Tab hotkeys changed from *Alt+<number>* to *Ctrl+<number>* on non-Linux for consistency with other applications

Fixed - Ctrl+Tab and Ctrl+Shift+Tab not working on macOS - Module menu options which aren't usable in current context are now correctly grayed out

[0.24.0] - 2019-06-28 #### Packaging changes - New macOS dependency: [PyAutoGUI](<https://pypi.org/project/PyAutoGUI/>)

Added - Turbo mode where Pext auto-selects options whenever reasonably confident

Changed - Notification when typing has completed

Fixed - Context menus are now searchable - Don't crash if pynput fails to import - User asked for internal update checker if disabled in constants - Add repeat polyfill to prevent rendering issues on systems with older Qt versions - Fix autotype on macOS

[0.23] - 2019-03-19 #### Packaging changes - New dependency: [watchdog](<https://pypi.org/project/watchdog/>)

API changes - Bump API version to 0.10.0 so that modules can hide the header if it was used to show the tree - Bump API version to 0.11.0 due to change in how a base context option is returned

Added - Searches can now be regular expressions. These need to be formatted as */search_string/flags* - Information when module requests are still being processed when showing no results screen - Notification on copying data to clipboard

Changed - Modules are now automatically unloaded after uninstallation - Passing arguments to modules is now done with Ctrl+Enter instead of typing it with the search bar - The start screen now shows a hotkey reference - The UI now shows what hotkey will activate what entry - Right click / Ctrl+Return a context menu entry to activate command input mode if available - The selection tree is now shown below the header location - The "base" context menu is now merged with the entry-specific one and shown below the entry-specific options - Command menu now has an "enter arguments" entry

Fixed - AppImage trying to store data inside itself in portable mode - Some incorrect hotkeys on macOS - Search sometimes missing entries

[0.22] - 2018-12-19 #### Added - Portable builds for Linux and macOS - *--portable* flag to make Pext behave more self-contained, readable by modules in settings

Changed - *--config* flag has been renamed to *--data-path* - Remember the geometry of the main window - The tray icon menu now lists all loaded modules for easier switching

Fixed - Sizing issues when moving to another monitor - Window not being resizeable - API version still being reported as 0.8.0 internally

[0.21] - 2018-11-02 #### API changes - Bump API version to 0.9.0 - Commands can now be multiple words and arguments will be given in a new 'args' field - *ask_question_default_yes* and *ask_question_default_no* are deprecated in favor of *ask_question*

Changed - Better error logging, using dialogs for critical errors - Pext's Window is now 800x600 by default and centered on the screen

Fixed - Module installation issues on Windows and Linux Mint - Title and tray tooltip are now translatable

[0.20] - 2018-10-12 #### Added - Are you sure message when closing Pext normally - Add installable touch bar quick action service for easier launching on macOS

Changed - Pick a more reasonable height on wide screens

Fixed - Minimize normally manually now works as intended - Autotype now correctly queues up multiple entries to type - Focus fix on macOS is now fast and reliable again - Logo background is no longer misaligned

[0.19] - 2018-09-05 ### Added - Foreground Pext at any time by pressing Ctrl+`

Changed - Move upstream URLs to pext.io

Fixed - Fixed --module flag - Fixed module install screens (from URL and from repo) failing when redirected

[0.18] - 2018-08-22 ### Added - Belarusian translation (thanks, Nelly Simkova!)

Fixed - Module installation issues in Windows distribution

[0.17] - 2018-07-08 ### Packaging changes - New dependency: [requests](https://pypi.org/project/requests/)

Added - Metadata i18n support - No result text when filtering empties list

Fixed - Pext on macOS now ignores -psn_0_* arguments - USE_INTERNAL_UPDATER is now used correctly - Terminal window opening on Windows - macOS certificate check failing on update check - Off-by-one error in git describe version generation - Installing module fails (rebuild on dulwich 0.9.15)

[0.16] - 2018-06-22 ### Packaging changes - New dependency: [pynput](https://pypi.org/project/pynput/) - New macOS dependency: [accessibility](https://pypi.org/project/accessibility) - Dependency removal: notify-send

Added - Ability to switch output location on runtime - Ability to type output directly - Ability to automatically update modules - Windows installer - Polish translation

Changed - Switch to Qt5 for notifications - Remove delay in showing notifications - Core and module update checks are now done together - Critical module errors now create a dialog box

Fixed - Inconsistent behaviour between clicking or selecting an entry - MacOS menu not merging on non-English languages

[0.15] - 2018-06-07 ### Packaging changes - The macOS .dmg is now officially supported

Changed - Remove quit without saving option - Configuration changes are now saved instantly, instead of only on a clean quit

Fixed - Focus not resetting after Pext hiding on macOS - Updated PyQt5 to fix some emoji display issues - macOS .dmg not being able to install all modules

[0.14] - 2018-04-22 ### Packaging changes - Packagers can now modify pext/constants.py to more easily control some behaviour

Added - French translation (thanks, Aurora Yeen!)

Changed - Modules are now installed by metadata.json, instead of by git URL - Make pext_dev default to CC-BY-3.0 for themes

Fixed - Make text properly wrap in the installing from repository dialog - Crashes on tab completion and minor errors (regression in 0.13) - Crash on module reloading (regression in 0.13) - IDs and names are now used more consistently - Crash when trying to load a theme as a module - Update pext_dev to be create files compatible with current Pext - Pext profile locks are now per-user instead of globally (fixes being unable to start if another user is running Pext)

[0.13] - 2018-04-07 ### Added - Hindi translation (thanks, Satyam Singh!) - Add automatic AppImage builds (thanks, TheAssassin!)

Changed - Tray icon is now always shown when the application is minimized to tray - Hide minimize to tray on macOS (too broken, can cause crashes) - The main window now has a minimal size of 500x300 - Versioning now complies with PEP440 - Merge module and theme repo and object selection into a single screen to save a click - Modules and themes are now saved based on the location of their ID - Modules and themes being in an incorrect location for their ID are automatically removed

Fixed - Themes now apply properly on Windows (forcing Fusion styling) - Properly fix i18n handling and giving i18n to modules - The name setting in metadata.json is now consistently respected - Make &Pext translatable

[0.12] - 2018-03-04 #### Added - Support renaming profiles - Switching profile from the GUI - Opening a second instance with another profile from the GUI - Basic profile management from the GUI - Ability to change language through the UI - *--list-locales* argument to show supported languages - The installation screens now tell you if you already have a module or theme installed - Russian translation (thanks, Ivan Semkin)

Changed - Profile name is no longer displayed if default - Trying to create a profile that already exists throws an error - Trying to delete a profile that is currently in use throws an error - Use argparse for argument parsing instead of getopt - Update checking now happens if the last check was over 24 hours, instead of each app launch - Combine all menu groups in settings for organizational purposes - Relicensed documentation under CC BY-SA 4.0

Fixed - Pext crash when module tries to empty context_menu_base - Inconsistent font sizing - Improved main screen resizing and logo showing - Pext passing None as locale to modules in some cases - *--background* stealing focus on macOS - Modules and themes are now sorted correctly in the install from repository lists

Removed - Removed manpage

[0.11.1] - 2017-12-19 #### Packaging changes - Fix missing translation files

[0.11] - 2017-12-19 #### Packaging changes - Pext now depends on dulwich - Pext no longer depends on pygit2

Translation updates - Added Norwegian Bokmål (thanks, Allan Nordhøy!) - Update Chinese (Traditional) translation - Update Spanish translation - Update Hungarian translation - Update Dutch translation

Fixed - Ubuntu/Debian compatibility for git operations over HTTPS - Install module from URL screen not working (regression from adding theming support for 0.9) - Theme selector now correctly displays current theme before switching - Pext no longer creates an empty theme file for the system theme and doesn't show it in the list of themes

[0.10] - 2017-11-11 #### Packaging changes - Pext now depends on pygit2, which uses libgit2, instead of git

API changes - Bump API version to 0.7.0 - Add *set_entry_info* queue call - Add *replace_entry_info_dict* queue call - Add *set_command_info* queue call - Add *replace_command_info_dict* queue call - Add *set_base_info* queue call - Add *set_entry_context* queue call - Add *replace_entry_context_dict* queue call - Add *set_command_context* queue call - Add *replace_command_context_dict* queue call - Add *set_base_context* queue call - Add *extra_info_request* function - Add a none *SelectionType* - Made more parameters optional

Added - Add info panels which modules can use to show extra info on the current status on selected entry - Add context panels for state changes and extra actions for entries/commands - Traceback is now printing when an exception is triggered - Last updated info for modules - Version info for modules - Windows support - Support for checking for updates (stable versions only)

Changed - Command mode no longer locks onto the first entry - Commands are always displayed in italics, instead of using italics for whatever is unfocused - Versioning is now more precise - Check if a module/theme has an update before updating it - Pext now auto-restarts after changing the theme - Pext now displays less broken when the height is higher than the width - Removed tray menu because it can't be translated due to PyQt limitations - Make clicking the tray icon toggle visibility on macOS - Minimizing normally after Pext is done is now the default on all platforms - Module requesting window hide will only reset the selection of that module instead of all - The *--exit* option got removed, Pext now will only start the UI if no options were given or all options were session-related

Fixed - Regression introduced in 0.9 which could cause selections to trigger wrongly when emptying the search bar - Page up and down causing QML errors when used too close to the start/end of the list - Minimizing behaviour didn't always work - Git commands are now properly limited to Pext directories - Desktop notifications now also show when Pext is minimized normally - Modules no longer lock up Pext while making a selection - Direct Git URL clone ending in / no longer creates an undeletable module - Modules now always properly get localization info - Ugly line between entries and entry info in some themes - No themes available dialog now correctly shows - Modules can't crash Pext by throwing an exception on stopping on Pext exit

[0.9] - 2017-08-23 #### API changes - Whenever the state changes (either by the user going back, selecting something or *set_selection* being called), the queue is now emptied - *ask_input* and *ask_input_password* now ask for a prefill before

the identifier

Translation updates - Added traditional Chinese (thanks, Jeff Huang!) - Added Spanish (thanks, Emily Lau!) - Updated Dutch (thanks, Heimen Stoffels!)

Added - Theming support based on QPalette - UI option to choose minimizing behaviour - UI option to choose sorting behaviour - UI toggle to enable/disable tray icon - --background command line option to make Pext not launch/foreground the UI

Changed - The design philosophy is now explained in the empty state screen - pyqt5 is added as install_requires - The about dialog now thanks translators - Info-only CLI parameters will no longer launch Pext as well (-help, -version, -list-styles, -list-modules, -list-themes) - Closing the main window will now quit Pext and save state

Fixed - pext_dev's generated base file now leaves the copyright open for the author to fill in - Not being able to select an entry until the list is fully loaded - Selection constantly resetting while items are being added - Loading and reloading a module while text is in search now applies the filter correctly - Fix crash in command mode when pressing enter while hovering over a wrong entry

[0.8] - 2017-04-28 ### API changes - The settings variable now contains _api_version ([major, minor, patch]) and _locale by default - Queue requests that cause process_response to be called can now optionally give an identifier to receive when process_response is called - Modules must now declare their settings in metadata.json

Added - Simple pext_dev command to help module development - Support metadata.json for showing info on installed modules - i18n support - Dutch translation

Changed - Move all UI code to QML - Improved installation dialogs - Improve load module dialog - Get rid of update and uninstall dialogs in favor for a central module management dialog - Check module functions parameter length on module load to prevent some runtime crashes for modules - Module settings is no longer a freeform input field - Display "Waiting" instead of "Ready" in the statusbar when not processing and the active module has not sent anything yet

Fixed - Crash when picking a command while there are also other entries to display

Removed - config.ini for editing Pext config directory (use \$XDG_CONFIG_HOME or \$HOME instead)

[0.7] - 2017-04-10 ### Added - Clear/back button in the UI

Changed - Minor UI font size changes - Pext's QML now uses StandardKeys in most places

Fixed - Fix Debian detection (no longer incorrectly detects openSUSE as Debian) - Fix nonsense load/update/uninstall dialogs if no modules are installed

[0.6.1] - 2017-04-01 ### Fixed - Clicking the tray icon no longer toggles visibility on macOS - XDG_CONFIG_HOME is now correctly used when available - The environment is no longer cleared when doing the initial git clone (security: the old behaviour would cause a proxy defined in the environment to be ignored)

[0.6] - 2017-03-27 ### Added - Install dependencies automatically if the module provides a requirements.txt file

Fixed - If module installation fails, the module directory is removed, so a subsequent installation doesn't instantly fail - Modules are now correctly unloaded when they raise a critical error - Added workaround for Ubuntu systems running the proprietary nvidia driver (<https://github.com/Pext/Pext/issues/11>)

[0.5] - 2017-03-22 ### API changes - Remove Action.notify_message and Action.notify_error, which are synonyms for add_message and add_error

Added - Documentation - Repository for third-party modules

Changed - Give more information upon installing modules and warn the user that they are code - User commands will now be auto-completed to the selected command

Fixed - Files unexpectedly existing in ~/.config/pext/modules/ no longer causes a crash

[0.4.1] - 2017-03-05 ### Changed - The default window is no longer explicitly borderless - The logo now has a white background which improves readability on dark themes

Fixed - An error occuring when retrieving the list of downloadable modules no longer causes a crash - Selecting a command entry after the entry list no longer causes a crash

[0.4] - 2017-02-20 #### Added - Basic Qt5 theming support using installed system themes - Allow for a pext/config.ini file to overwrite some default configuration - Allow the user to disable tray icon creation - Add entry to open homepage from help menu (so the user can find support)

Changed - Get list of installable modules from pext.hackerchick.me instead of pext.github.io

[0.3] - 2016-12-29 #### API changes - The entry and command list will now be emptied each time just before selection_made is called

Added - Busy indicator when the list of entries is empty for a more responsive look - Support for getting a list of installable modules from pext.github.io

Fixed - All commands now correctly show up after emptying the search bar - Module lists are now sorted alphabetically - selection_made is no longer unnecessarily triggered when closing the window

[0.2] - 2016-11-21 #### Added - System tray icon

[0.1] - 2016-11-19 Initial release

METADATA

All module or theme metadata is stored in a file named `metadata.json`. This file should be in the root directory of the module or theme.

For up-to-date examples, look at any of the modules or themes in the [Pext GitHub organisation](#).

3.1 Fields

Name	Type	Description
bugtracker	String	URL location of the bugtracker
bug-tracker_type	String	Enables Pext to prefill a bugtracker's values if a known type. Supported values: "github"
description	String	Description of the module
developer	String	Name of the developer
git_urls	List<String>	A list of git URLs the module/theme can be cloned from
homepage	String	URL location of the module/theme's homepage
id	String	Unique identifier of the module
license	String	Module license as SPDX identifier
name	String	Module name
platforms	List<String>	Supported platforms. Can be "Linux", "Darwin" (for macOS) or "Windows"
settings	List<Object>	A list of all possible settings. See Settings

3.2 Settings

Each value in the settings list has the following possible fields:

Name	Type	Description
default	Object	The default value
description	String	User-facing description of the option
name	String	Internally used name of the option
options	List<Object>	A list of possible options. Optional

REQUIREMENTS

Modules may specify Python requirements in a `requirements.txt` file.

The requirements should be specified one requirement per line, with the version number explicitly specified to prevent unexpected breakage due to upstream changes.

More advanced information on using a *requirements.txt* file can be found on https://pip.readthedocs.io/en/latest/reference/pip_install/#requirements-file-format.

For more up-to-date examples, look at any of the modules or themes in the [Pext GitHub](#) organisation.

4.1 Example

```
dulwich==0.19.13
paramiko==2.6.0
pypass==0.2.1
Babel==2.7.0
git+https://github.com/TheLastProject/pyotp@5d7bf9d10e3bdef8c93917c1f4f5ffcf799a671e
↳ #egg=pyotp
pyscreenshot==0.5.1
zbar-py==1.0.4
```


5.1 Base

Every module needs to have a module class which inherits from ModuleBase.

Example:

```
# Copyright (c) <year> <Your name> <Your email>
#
# This program is free software: you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation, either version 3 of the License, or
# (at your option) any later version.
#
# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with this program. If not, see <http://www.gnu.org/licenses/>.

from pext_base import ModuleBase
from pext_helpers import Action, SelectionType

class Module(ModuleBase):
    def init(self, settings, q):
        pass

    def stop(self):
        pass

    def selection_made(self, selection):
        pass

    def process_response(self, response, identifier):
        pass
```

See `pext_base.py` for detailed information about every function.

5.2 Helpers

Pext provides 2 helpers for modules.

5.2.1 Action

The first type is the Action helper, which supply a list of actions which modules can request Pext to do.

See [pext_helpers.py](#) for a list of Actions.

5.2.2 SelectionType

The second type is the SelectionType helper, which supplies an enumerator containing possible selection types that Pext may pass to some functions.

See [pext_helpers.py](#) for a list of SelectionTypes.

PEXT_DEV

`pext_dev` is a helper script to aid in module and theme development. It is shipped in the source code versions of Pext and can be ran with Python 3.

```
$ python3 -m pext_dev --help
```

Options:

```
module init[=PATH]
```

Initialize a new module in the current directory or given path.

```
module run
```

Run the module in the current directory a new Pext instance. Added options are ↪
↪passed to Pext as-is.

```
theme init[=PATH]
```

Initialize a new theme in the current directory or given path.

```
theme run
```

Run the theme in the current directory a new Pext instance. Added options are passed ↪
↪to Pext as-is.

All theme information is stored in a file named `theme.conf`. This file should be in the root directory of the theme. For up-to-date examples, look at any of the themes in the [Pext GitHub organisation](#).

7.1 File format

The `theme.conf` file is in ini format and has 4 headers (color groups) which contain keys and values (color roles). Every color group and role is optional. If not defined, system colors are used.

Color Roles are written as simple `key = value` lines with the value containing 3 comma-separated numbers for the amount of red, green and blue (RGB) ranging from 0 to 255.

Please be aware that Qt, the graphical user interface library Pext uses, has some code to automatically slightly adjust colors based on their location for contrast, so you may not get the exact color you defined.

7.1.1 Color Groups

Name	Description
Disabled	Used when the element cannot be interacted with
Active	Used when the element is focused and interactable
Inactive	Used when the element is unfocused (background)
All	Used when no more specific match exists

7.1.2 Color Rules

These descriptions are taken from <https://doc.qt.io/qt-5/qpalette.html#ColorRole-enum>. Not all of these may have an effect in Pext, so experiment!

Name	Description
Win- dow	A general background color
Win- dow- Text	A general foreground color
Base	Used mostly as the background color for text entry widgets, but can also be used for other painting - such as the background of combobox drop down lists and toolbar handles. It is usually white or another light color
Al- ter- nate- Base	Used as the alternate background color in views with alternating row colors
ToolTip Base	Used as the background color for QToolTip and QWhatsThis. Tool tips use the Inactive color group of QPalette, because tool tips are not active windows
ToolTip Text	Used as the foreground color for QToolTip and QWhatsThis. Tool tips use the Inactive color group of QPalette, because tool tips are not active windows
Place- hold- er- Text	Used as the placeholder color for various text input widgets
Text	The foreground color used with Base. This is usually the same as the WindowText, in which case it must provide good contrast with Window and Base
But- ton	The general button background color. This background can be different from Window as some styles require a different background color for buttons
But- ton- Text	A foreground color used with the Button color
Bright- Text	A text color that is very different from WindowText, and contrasts well with e.g. Dark. Typically used for text that needs to be drawn where Text or WindowText would give poor contrast, such as on pressed push buttons. Note that text colors can be used for things other than just words; text colors are usually used for text, but it's quite common to use the text color roles for lines, icons, etc
Light	Lighter than Button color
Mid- light	Between Button and Light
Dark	Darker than Button
Mid	Between Button and Dark
Shadow	A very dark color. By default, the shadow color is black
High- light	A color to indicate a selected item or the current item. By default, the highlight color is dark blue
High- light- ed- Text	A text color that contrasts with Highlight. By default, the highlighted text color is white
Link	A text color used for unvisited hyperlinks. By default, the link color is blue
LinkVis- ited	A text color used for already visited hyperlinks. By default, the linkvisited color is magenta

INDICES AND TABLES

- `genindex`
- `modindex`
- `search`