
Pyforms Terminal Documentation

Release 4.0

Ricardo Jorge Vieira Ribeiro

May 28, 2019

PYFORMS TERMINAL

1	Overview	3
1.1	Pyforms Terminal	3
1.2	Pyforms	3
1.3	Developer	3
2	Install & configure	5
3	First application	7
3.1	Create the first app	7
4	Python	11
4.1	BaseWidget	11
4.1.1	Overview	11
4.1.2	API	11
4.2	Controls	12
4.2.1	ControlBase	12
4.2.2	ControlBoundingSlider	12
4.2.3	ControlButton	12
4.2.4	ControlCheckBox	12
4.2.5	ControlCombo	13
4.2.6	ControlDir	13
4.2.7	ControlEmptyWidget	13
4.2.8	ControlFile	13
4.2.9	ControlImage	13
4.2.10	ControlLabel	14
4.2.11	ControlPlayer	14
4.2.12	ControlProgress	14
4.2.13	ControlSlider	15
4.2.14	ControlText	15
4.2.15	ControlTextArea	15
5	Indices and tables	17
	Python Module Index	19

Pyforms Terminal is Python 3 framework to allow pyforms applications to execute in terminal mode.

The framework aims the boost the development productivity by providing an API in Python to allow the execution of applications developed for GUI and Web mode in terminal mode.

Source code <https://github.com/UmSenhorQualquer/pyforms-terminal>

```
usage: SingleFly.py [-h] [--_outputfile _OUTPUTFILE] [--_filename _FILENAME]
                  [--_outputvideo _OUTPUTVIDEO]
                  [--_mov_threshold _MOV_THRESHOLD] [--_led_y _LED_Y]
                  [--_led_x _LED_X] [--_arenaRadius _ARENARADIUS]
                  [--exec EXEC]

optional arguments:
  -h, --help            show this help message and exit
  --_outputfile _OUTPUTFILE
                        Output CSV file
  --_filename _FILENAME
                        Video file name
  --_outputvideo _OUTPUTVIDEO
                        Output video file
  --_mov_threshold _MOV_THRESHOLD
                        Movement threshold
  --_led_y _LED_Y       LED Y position
  --_led_x _LED_X       LED X position
  --_arenaRadius _ARENARADIUS
                        Arena radius (mm)
  --exec EXEC           Function from the application that should be executed.
                        Use | to separate a list of functions.
```

Note: This framework is a software layer part of the Pyforms framework.

Pyforms <https://pyforms.readthedocs.io>

OVERVIEW

1.1 Pyforms Terminal



Pyforms Terminal is part the Pyforms framework. It implements a software layer that handles the execution of pyforms applications in over the terminal.

1.2 Pyforms



Pyforms is a Python 3 framework to develop applications capable of executing in 3 diferent environments, Desktop GUI, Terminal and Web.

1.3 Developer

Ricardo Ribeiro	Champalimaud Scientific Software Platform ricardo.ribeiro@research.fchampalimaud.org ricardojvr@gmail.com
--------------------	-----------------------------------------------------------------------------------------------------------------

Note: Please **star** the project at the [Github repository](#) to support the project.

INSTALL & CONFIGURE

- Install Pyforms using **pip**.

```
pip install pyforms-terminal
```


FIRST APPLICATION

Note: More documentation to read about this example at:

- `pyforms_terminal.basewidget.BaseWidget`
- `pyforms_terminal.controls.control_base.ControlBase`

Here it is shown how to create the first pyforms app.

3.1 Create the first app

Create the file **example.py** and add the next code to it.

```
from pyforms.basewidget import BaseWidget
from pyforms.controls import ControlFile
from pyforms.controls import ControlText
from pyforms.controls import ControlSlider
from pyforms.controls import ControlPlayer
from pyforms.controls import ControlButton

class ComputerVisionAlgorithm(BaseWidget):

    def __init__(self, *args, **kwargs):
        super().__init__('Computer vision algorithm example')

        #Definition of the forms fields
        self._videofile = ControlFile('Video')
        self._outputfile = ControlText('Results output file')
        self._threshold = ControlSlider('Threshold', 114, 0,255)
        self._blobsize = ControlSlider('Minimum blob size', 100, 100,2000)
        self._player = ControlPlayer('Player')
        self._runbutton = ControlButton('Run')

        #Define the function that will be called when a file is selected
        self._videofile.changed_event = self.__videoFileSelectionEvent
        #Define the event that will be called when the run button is processed
        self._runbutton.value = self.__runEvent
        #Define the event called before showing the image in the player
        self._player.process_frame_event = self.__process_frame

        #Define the organization of the Form Controls
        self._formset = [
```

(continues on next page)

(continued from previous page)

```

        ('_videofile', '_outputfile'),
        '_threshold',
        ('_blobsize', '_runbutton'),
        '_player'
    ]

    def __videoFileSelectionEvent(self):
        """
        When the videofile is selected instanciate the video in the player
        """
        self._player.value = self._videofile.value

    def __process_frame(self, frame):
        """
        Do some processing to the frame and return the result frame
        """
        return frame

    def __runEvent(self):
        """
        After setting the best parameters run the full algorithm
        """
        pass

if __name__ == '__main__':

    from pyforms import start_app
    start_app(ComputerVisionAlgorithm)

```

Now execute in the terminal the next command:

```
$ python example.py terminal_mode --help
```

You will visualize the next result:

```

usage: example.py [-h] [--videofile _VIDEOFILE] [--outputfile _OUTPUTFILE]
                 [--threshold _THRESHOLD] [--blobsize _BLOBSIZE]
                 [--exec EXEC] [--load LOAD]
                 terminal_mode

positional arguments:
  terminal_mode          Flag to run pyforms in terminal mode

optional arguments:
  -h, --help            show this help message and exit
  --videofile _VIDEOFILE
                        Video
  --outputfile _OUTPUTFILE
                        Results output file
  --threshold _THRESHOLD
                        Threshold
  --blobsize _BLOBSIZE
                        Minimum blob size
  --exec EXEC           Function from the application that should be executed.
                        Use | to separate a list of functions.

```

(continues on next page)

(continued from previous page)

```
--load LOAD          Load a json file containing the pyforms form
                     configuration.
```

Note: In alternative if you would not like to use the **terminal_mode** parameter you can create the file **local_settings.py** in the same directory where you are going to run the application and add the next code:

```
SETTINGS_PRIORITY = 0
PYFORMS_MODE = 'TERMINAL'
```

This code will set pyforms to run in terminal mode.

Now you can run the application in terminal mode using the command:

```
$ python example.py --help
```


4.1 BaseWidget

4.1.1 Overview

The BaseWidget class is the base class of all pyforms applications.

4.1.2 API

```
class pyforms_terminal.basewidget.BaseWidget (*args, **kwargs)
    Bases: object

    set_margin (margin)
    init_form (parse=True)
    execute ()
    start_progress (total=100)
    update_progress ()
    end_progress ()
    message (msg, title=None, msg_type=None)
    success (msg, title=None)
    info (msg, title=None)
    warning (msg, title=None)
    alert (msg, title=None)
    critical (msg, title=None)
    about (msg, title=None)
    aboutQt (msg, title=None)
    executeCommand (cmd, cwd=None, env=None)
    exec_terminal_cmd (args, **kwargs)

    controls
        Return all the form controls from the the module
```

4.2 Controls

4.2.1 ControlBase

```
class pyforms_terminal.controls.control_base.ControlBase (*args, **kwargs)
    Bases: object
    init_form()
    changed_event()
        Function called when ever the Control value is changed
    show()
    hide()
    open_popup_menu(position)
    add_popup_submenu_option(label, options)
    enabled
    value
    label
    form
    parent
```

4.2.2 ControlBoundingSlider

```
class pyforms_terminal.controls.control_boundinglider.ControlBoundingSlider (*args,
                                                                              **kwargs)
    Bases: pyforms_terminal.controls.control_base.ControlBase
```

4.2.3 ControlButton

```
class pyforms_terminal.controls.control_button.ControlButton (*args, **kwargs)
    Bases: pyforms_terminal.controls.control_base.ControlBase
    value
```

4.2.4 ControlCheckBox

```
class pyforms_terminal.controls.control_checkbox.ControlCheckBox (*args,
                                                                    **kwargs)
    Bases: pyforms_terminal.controls.control_base.ControlBase
    value
```

4.2.5 ControlCombo

```
class pyforms_terminal.controls.control_combo.ControlCombo (*args, **kwargs)
    Bases: pyforms_terminal.controls.control_base.ControlBase

    add_item (label, value=None)

    clear ()

    items

    values

    value

    text
```

4.2.6 ControlDir

```
class pyforms_terminal.controls.control_dir.ControlDir (*args, **kwargs)
    Bases: pyforms_terminal.controls.control_base.ControlBase
```

4.2.7 ControlEmptyWidget

```
class pyforms_terminal.controls.control_emptywidget.ControlEmptyWidget (*args,
                                                                            **kwargs)
    Bases: pyforms_terminal.controls.control_base.ControlBase

    load_form (data, path=None)
```

4.2.8 ControlFile

```
class pyforms_terminal.controls.control_file.ControlFile (*args, **kwargs)
    Bases: pyforms_terminal.controls.control_base.ControlBase
```

4.2.9 ControllImage

```
class pyforms_terminal.controls.control_image.ControlImage (*args, **kwargs)
    Bases: pyforms_terminal.controls.control_base.ControlBase

    repaint ()

    value
```

4.2.10 ControlLabel

```
class pyforms_terminal.controls.control_label.ControlLabel (*args, **kwargs)
    Bases: pyforms_terminal.controls.control_base.ControlBase
```

4.2.11 ControlPlayer

```
class pyforms_terminal.controls.control_player.ControlPlayer (*args, **kwargs)
    Bases: pyforms_terminal.controls.control_base.ControlBase

    play()
    stop()
    refresh()
    update_frame()
    double_click_event (event, x, y)
    click_event (event, x, y)
    drag_event (start_point, end_point)
    end_drag_event (start_point, end_point)
    key_release_event (event)
    process_frame_event (frame)
    save_Form (data)
    load_form (data)
    value
    video_index
    image
    frame_width
    frame_height
    max
    frame
    fps
        Return the video frames per second
```

4.2.12 ControlProgress

```
class pyforms_terminal.controls.control_progress.ControlProgress (*args,
                                                                    **kwargs)
    Bases: pyforms_terminal.controls.control_base.ControlBase

    value
    min
```

max

4.2.13 ControlSlider

```
class pyforms_terminal.controls.control_slider.ControlSlider (*args, **kwargs)
    Bases: pyforms_terminal.controls.control_base.ControlBase

    min

    max
```

4.2.14 ControlText

```
class pyforms_terminal.controls.control_text.ControlText (*args, **kwargs)
    Bases: pyforms_terminal.controls.control_base.ControlBase
```

4.2.15 ControlTextArea

```
class pyforms_terminal.controls.control_textarea.ControlTextArea (*args,
                                                                    **kwargs)
    Bases: pyforms_terminal.controls.control_base.ControlBase
```


INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

p

`pybpod_web.basewidget.BaseWidget`, [11](#)
`pyforms_terminal.basewidget`, [11](#)
`pyforms_terminal.controls`, [12](#)

INDEX

A

`about()` (*pyforms_terminal.basewidget.BaseWidget* method), 11

`aboutQt()` (*pyforms_terminal.basewidget.BaseWidget* method), 11

`add_item()` (*pyforms_terminal.controls.control_combo.ControlCombo* method), 13

`add_popup_submenu_option()` (*pyforms_terminal.controls.control_base.ControlBase* method), 12

`alert()` (*pyforms_terminal.basewidget.BaseWidget* method), 11

`ControlEmptyWidget` (class in *pyforms_terminal.controls.control_emptywidget*), 13

`ControlFile` (class in *pyforms_terminal.controls.control_file*), 13

`ControlImage` (class in *pyforms_terminal.controls.control_image*), 13

`ControlLabel` (class in *pyforms_terminal.controls.control_label*), 14

`ControlPlayer` (class in *pyforms_terminal.controls.control_player*), 14

B

`BaseWidget` (class in *pyforms_terminal.basewidget*), 11

C

`changed_event()` (*pyforms_terminal.controls.control_base.ControlBase* method), 12

`clear()` (*pyforms_terminal.controls.control_combo.ControlCombo* method), 13

`click_event()` (*pyforms_terminal.controls.control_player.ControlPlayer* method), 14

`ControlBase` (class in *pyforms_terminal.controls.control_base*), 12

`ControlBoundingSlider` (class in *pyforms_terminal.controls.control_boundingslider*), 12

`ControlButton` (class in *pyforms_terminal.controls.control_button*), 12

`ControlCheckBox` (class in *pyforms_terminal.controls.control_checkbox*), 12

`ControlCombo` (class in *pyforms_terminal.controls.control_combo*), 13

`ControlDir` (class in *pyforms_terminal.controls.control_dir*), 13

`ControlProgress` (class in *pyforms_terminal.controls.control_progress*), 14

`controls` (*pyforms_terminal.basewidget.BaseWidget* attribute), 11

`ControlSlider` (class in *pyforms_terminal.controls.control_slider*), 15

`ControlText` (class in *pyforms_terminal.controls.control_text*), 15

`ControlTextArea` (class in *pyforms_terminal.controls.control_textarea*), 15

`critical()` (*pyforms_terminal.basewidget.BaseWidget* method), 11

D

`double_click_event()` (*pyforms_terminal.controls.control_player.ControlPlayer* method), 14

`drag_event()` (*pyforms_terminal.controls.control_player.ControlPlayer* method), 14

E

`enabled` (*pyforms_terminal.controls.control_base.ControlBase* attribute), 12

`end_drag_event()` (*pyforms_terminal.controls.control_player.ControlPlayer* method), 14

end_progress()	(pyforms_terminal.basewidget.BaseWidget method), 11	M	max (pyforms_terminal.controls.control_player.ControlPlayer attribute), 14
exec_terminal_cmd()	(pyforms_terminal.basewidget.BaseWidget method), 11	O	max (pyforms_terminal.controls.control_progress.ControlProgress attribute), 14
execute()	(pyforms_terminal.basewidget.BaseWidget method), 11	P	max (pyforms_terminal.controls.control_slider.ControlSlider attribute), 15
executeCommand()	(pyforms_terminal.basewidget.BaseWidget method), 11	parent	message () (pyforms_terminal.basewidget.BaseWidget method), 11
F		play	min (pyforms_terminal.controls.control_progress.ControlProgress attribute), 14
form (pyforms_terminal.controls.control_base.ControlBase attribute), 12		process_frame_event	min (pyforms_terminal.controls.control_slider.ControlSlider attribute), 15
fps (pyforms_terminal.controls.control_player.ControlPlayer attribute), 14		pybpod_web.basewidget.BaseWidget	
frame (pyforms_terminal.controls.control_player.ControlPlayer attribute), 14		pyforms_terminal.basewidget	open_popup_menu () (pyforms_terminal.controls.control_base.ControlBase method), 12
frame_height (pyforms_terminal.controls.control_player.ControlPlayer attribute), 14		pyforms_terminal.controls	
frame_width (pyforms_terminal.controls.control_player.ControlPlayer attribute), 14		refresh	
H		repaint	
hide () (pyforms_terminal.controls.control_base.ControlBase method), 12		save_Form	
I		set_margin	
image (pyforms_terminal.controls.control_player.ControlPlayer attribute), 14		show	
info () (pyforms_terminal.basewidget.BaseWidget method), 11		start_progress	
init_form () (pyforms_terminal.basewidget.BaseWidget method), 11		stop	
init_form () (pyforms_terminal.controls.control_base.ControlBase method), 12		success	
items (pyforms_terminal.controls.control_combo.ControlCombo attribute), 13			
K			
key_release_event () (pyforms_terminal.controls.control_player.ControlPlayer method), 14			
L			
label (pyforms_terminal.controls.control_base.ControlBase attribute), 12			
load_form () (pyforms_terminal.controls.control_emptywidget.ControlEmptyWidget method), 13			
load_form () (pyforms_terminal.controls.control_player.ControlPlayer method), 14			

T

text (*pyforms_terminal.controls.control_combo.ControlCombo* attribute), 13

U

update_frame() (*pyforms_terminal.controls.control_player.ControlPlayer* method), 14

update_progress() (*pyforms_terminal.basewidget.BaseWidget* method), 11

V

value (*pyforms_terminal.controls.control_base.ControlBase* attribute), 12

value (*pyforms_terminal.controls.control_button.ControlButton* attribute), 12

value (*pyforms_terminal.controls.control_checkbox.ControlCheckBox* attribute), 12

value (*pyforms_terminal.controls.control_combo.ControlCombo* attribute), 13

value (*pyforms_terminal.controls.control_image.ControlImage* attribute), 13

value (*pyforms_terminal.controls.control_player.ControlPlayer* attribute), 14

value (*pyforms_terminal.controls.control_progress.ControlProgress* attribute), 14

values (*pyforms_terminal.controls.control_combo.ControlCombo* attribute), 13

video_index (*pyforms_terminal.controls.control_player.ControlPlayer* attribute), 14

W

warning() (*pyforms_terminal.basewidget.BaseWidget* method), 11