

---

# **guibot Documentation**

*Release 0.11*

**Plamen Dimitrov and Thomas Jarosch**

**Jul 26, 2018**



|          |                               |          |
|----------|-------------------------------|----------|
| <b>1</b> | <b>guiobot package</b>        | <b>1</b> |
| 1.1      | Submodules                    | 1        |
| 1.1.1    | guiobot.calibrator module     | 1        |
| 1.1.2    | guiobot.config module         | 1        |
| 1.1.3    | guiobot.desktopcontrol module | 1        |
| 1.1.4    | guiobot.errors module         | 1        |
| 1.1.5    | guiobot.finder module         | 2        |
| 1.1.6    | guiobot.guiobot module        | 2        |
| 1.1.7    | guiobot.guiobot_proxy module  | 2        |
| 1.1.8    | guiobot.guiobot_simple module | 2        |
| 1.1.9    | guiobot.imagelogger module    | 2        |
| 1.1.10   | guiobot.inputmap module       | 2        |
| 1.1.11   | guiobot.location module       | 5        |
| 1.1.12   | guiobot.match module          | 5        |
| 1.1.13   | guiobot.path module           | 5        |
| 1.1.14   | guiobot.region module         | 6        |
| 1.1.15   | guiobot.target module         | 6        |
| 1.2      | Module contents               | 6        |
|          | <b>Python Module Index</b>    | <b>7</b> |



## 1.1 Submodules

### 1.1.1 guibot.calibrator module

### 1.1.2 guibot.config module

### 1.1.3 guibot.desktopcontrol module

### 1.1.4 guibot.errors module

**exception** `guibot.errors.GuiBotError`

Bases: `exceptions.Exception`

GuiBot exception base class

**exception** `guibot.errors.FileNotFoundError`

Bases: `guibot.errors.GuiBotError`

Exception raised when a picture file cannot be found on disc

**exception** `guibot.errors.IncompatibleTargetError`

Bases: `guibot.errors.GuiBotError`

Exception raised when a matched target is of type that cannot be handled by the finder

**exception** `guibot.errors.IncompatibleTargetFileError`

Bases: `guibot.errors.GuiBotError`

Exception raised when a matched target is restored from a file of unsupported type

**exception** `guibot.errors.FindError` (*failed\_target=None*)

Bases: `guibot.errors.GuiBotError`

Exception raised when an Image cannot be found on the screen

`__init__` (*failed\_target=None*)

Build the exception possibly providing the failed target.

**Parameters** `failed_target` (`target.Target` or `None`) – the target that wasn't found

**exception** `guiobot.errors.NotFindError` (*failed\_target=None*)

Bases: `guiobot.errors.GuiBotError`

Exception raised when an Image can be found on the screen but should not be

`__init__` (*failed\_target=None*)

Build the exception possibly providing the failed target.

**Parameters** `failed_target` (`target.Target` or `None`) – the target that was found

**exception** `guiobot.errors.UnsupportedBackendError`

Bases: `guiobot.errors.GuiBotError`

Exception raised when a non-existent method is used for finding a target

**exception** `guiobot.errors.MissingHotmapError`

Bases: `guiobot.errors.GuiBotError`

Exception raised when an attempt to access a non-existent hotmap in the image logger is made

**exception** `guiobot.errors.UninitializedBackendError`

Bases: `guiobot.errors.GuiBotError`

Exception raised when a region is created within an empty screen (a disconnected desktop control backend)

### 1.1.5 `guiobot.finder` module

### 1.1.6 `guiobot.guiobot` module

### 1.1.7 `guiobot.guiobot_proxy` module

### 1.1.8 `guiobot.guiobot_simple` module

### 1.1.9 `guiobot.imagellogger` module

### 1.1.10 `guiobot.inputmap` module

**class** `guiobot.inputmap.Key`

Bases: `object`

Helper to contain all key mappings for a custom desktop control backend.

`__init__` ()

Build an instance containing an empty key map.

`to_string` (*key*)

Provide with a text representation of a desired key according to the custom BC backend.

**Parameters** `key` (*str*) – selected key name according to the custom backend

**Returns** text representation of the selected key

**Return type** `str`

**Raises** `ValueError` if *key* is not found in the current key map

**class** `guibot.inputmap.AutoPyKey`

Bases: `guibot.inputmap.Key`

Helper to contain all key mappings for the AutoPy DC backend.

`__init__()`

Build an instance containing the key map for the AutoPy backend.

**class** `guibot.inputmap.XDoToolKey`

Bases: `guibot.inputmap.Key`

Helper to contain all key mappings for the xdotool DC backend.

`__init__()`

Build an instance containing the key map for the xdotool backend.

**class** `guibot.inputmap.VNCDoToolKey`

Bases: `guibot.inputmap.Key`

Helper to contain all key mappings for the VNCDoTool DC backend.

`__init__()`

Build an instance containing the key map for the VNCDoTool backend.

**class** `guibot.inputmap.QemuKey`

Bases: `guibot.inputmap.Key`

Helper to contain all key mappings for the Qemu DC backend.

`__init__()`

Build an instance containing the key map for the Qemu backend.

**class** `guibot.inputmap.KeyModifier`

Bases: `object`

Helper to contain all modifier key mappings for a custom desktop control backend.

`__init__()`

Build an instance containing an empty modifier key map.

`to_string(key)`

Provide with a text representation of a desired modifier key according to the custom BC backend.

**Parameters** `key` (*str*) – selected modifier name according to the current backend

**Returns** text representation of the selected modifier

**Return type** `str`

**Raises** `ValueError` if `key` is not found in the current modifier map

**class** `guibot.inputmap.AutoPyKeyModifier`

Bases: `guibot.inputmap.KeyModifier`

Helper to contain all modifier key mappings for the AutoPy DC backend.

`__init__()`

Build an instance containing the modifier key map for the AutoPy backend.

**class** `guibot.inputmap.XDoToolKeyModifier`

Bases: `guibot.inputmap.KeyModifier`

Helper to contain all modifier key mappings for the xdotool DC backend.

`__init__()`

Build an instance containing the modifier key map for the xdotool backend.

**class** `guibot.inputmap.VNCDoToolKeyModifier`

Bases: `guibot.inputmap.KeyModifier`

Helper to contain all modifier key mappings for the VNCDoTool DC backend.

`__init__()`

Build an instance containing the modifier key map for the VNCDoTool backend.

**class** `guibot.inputmap.QemuKeyModifier`

Bases: `guibot.inputmap.KeyModifier`

Helper to contain all modifier key mappings for the Qemu DC backend.

`__init__()`

Build an instance containing the modifier key map for the Qemu backend.

**class** `guibot.inputmap.MouseButton`

Bases: `object`

Helper to contain all mouse button mappings for a custom desktop control backend.

`__init__()`

Build an instance containing an empty mouse button map.

`to_string(key)`

Provide with a text representation of a desired mouse button according to the custom BC backend.

**Parameters** `key` (*str*) – selected mouse button according to the current backend

**Returns** text representation of the selected mouse button

**Return type** `str`

**Raises** `ValueError` if `key` is not found in the current mouse map

**class** `guibot.inputmap.AutoPyMouseButton`

Bases: `guibot.inputmap.MouseButton`

Helper to contain all mouse button mappings for the AutoPy DC backend.

`__init__()`

Build an instance containing the mouse button map for the AutoPy backend.

**class** `guibot.inputmap.XDoToolMouseButton`

Bases: `guibot.inputmap.MouseButton`

Helper to contain all mouse button mappings for the xdotool DC backend.

`__init__()`

Build an instance containing the mouse button map for the xdotool backend.

**class** `guibot.inputmap.VNCDoToolMouseButton`

Bases: `guibot.inputmap.MouseButton`

Helper to contain all mouse button mappings for the VNCDoTool DC backend.

`__init__()`

Build an instance containing the mouse button map for the VNCDoTool backend.

**class** `guibot.inputmap.QemuMouseButton`

Bases: `guibot.inputmap.MouseButton`

Helper to contain all mouse button mappings for the Qemu DC backend.

`__init__()`

Build an instance containing the mouse button map for the Qemu backend.

### 1.1.11 guibot.location module

**class** `guibot.location.Location` (*xpos=0, ypos=0*)

Bases: `object`

Simple location on a 2D surface, region, or screen.

**\_\_init\_\_** (*xpos=0, ypos=0*)

Build a location object.

#### Parameters

- **xpos** (*int*) – x coordinate of the location
- **ypos** (*int*) – y coordinate of the location

**\_\_str\_\_** ()

Provide a compact form for the location.

**x**

Getter for readonly attribute.

**Returns** x coordinate of the location

**Return type** `int`

**y**

Getter for readonly attribute.

**Returns** y coordinate of the location

**Return type** `int`

### 1.1.12 guibot.match module

### 1.1.13 guibot.path module

**class** `guibot.path.Path`

Bases: `object`

Handler for currently used target paths or sources of targets with a desired name.

The methods of this class are shared among all of its instances.

**add\_path** (*directory*)

Add a path to the list of currently accessible paths if it wasn't already added.

**Parameters** **directory** (*str*) – path to add

**remove\_path** (*directory*)

Remove a path from the list of currently accessible paths.

**Parameters** **directory** (*str*) – path to add

**Returns** whether the removal succeeded

**Return type** `bool`

**clear** ()

Clear all currently accessible paths.

**search** (*filename, restriction="", silent=False*)

Search for a filename in the currently accessible paths.

**Parameters**

- **filename** (*str*) – filename of the target to search for
- **restriction** (*str*) – simple string to restrict the number of paths
- **silent** (*bool*) – whether to return None instead of error out

**Returns** the full name of the found target file or None if silent and no file was found

**Return type** str or None

**Raises** `FileNotFoundError` if no such file was found and not silent

### 1.1.14 guibot.region module

### 1.1.15 guibot.target module

## 1.2 Module contents

**g**

guiobot, 6  
guiobot.errors, 1  
guiobot.inputmap, 2  
guiobot.location, 5  
guiobot.path, 5



## Symbols

\_\_init\_\_() (guiobot.errors.FindError method), 1  
 \_\_init\_\_() (guiobot.errors.NotFindError method), 2  
 \_\_init\_\_() (guiobot.inputmap.AutoPyKey method), 3  
 \_\_init\_\_() (guiobot.inputmap.AutoPyKeyModifier method), 3  
 \_\_init\_\_() (guiobot.inputmap.AutoPyMouseButton method), 4  
 \_\_init\_\_() (guiobot.inputmap.Key method), 2  
 \_\_init\_\_() (guiobot.inputmap.KeyModifier method), 3  
 \_\_init\_\_() (guiobot.inputmap.MouseButton method), 4  
 \_\_init\_\_() (guiobot.inputmap.QemuKey method), 3  
 \_\_init\_\_() (guiobot.inputmap.QemuKeyModifier method), 4  
 \_\_init\_\_() (guiobot.inputmap.QemuMouseButton method), 4  
 \_\_init\_\_() (guiobot.inputmap.VNCDoToolKey method), 3  
 \_\_init\_\_() (guiobot.inputmap.VNCDoToolKeyModifier method), 4  
 \_\_init\_\_() (guiobot.inputmap.VNCDoToolMouseButton method), 4  
 \_\_init\_\_() (guiobot.inputmap.XDoToolKey method), 3  
 \_\_init\_\_() (guiobot.inputmap.XDoToolKeyModifier method), 3  
 \_\_init\_\_() (guiobot.inputmap.XDoToolMouseButton method), 4  
 \_\_init\_\_() (guiobot.location.Location method), 5  
 \_\_str\_\_() (guiobot.location.Location method), 5

## A

add\_path() (guiobot.path.Path method), 5  
 AutoPyKey (class in guiobot.inputmap), 2  
 AutoPyKeyModifier (class in guiobot.inputmap), 3  
 AutoPyMouseButton (class in guiobot.inputmap), 4

## C

clear() (guiobot.path.Path method), 5

## F

FileNotFoundError, 1  
 FindError, 1

## G

guibot (module), 6  
 guibot.errors (module), 1  
 guibot.inputmap (module), 2  
 guibot.location (module), 5  
 guibot.path (module), 5  
 GuiBotError, 1

## I

IncompatibleTargetError, 1  
 IncompatibleTargetFileError, 1

## K

Key (class in guiobot.inputmap), 2  
 KeyModifier (class in guiobot.inputmap), 3

## L

Location (class in guiobot.location), 5

## M

MissingHotmapError, 2  
 MouseButton (class in guiobot.inputmap), 4

## N

NotFindError, 2

## P

Path (class in guiobot.path), 5

## Q

QemuKey (class in guiobot.inputmap), 3  
 QemuKeyModifier (class in guiobot.inputmap), 4  
 QemuMouseButton (class in guiobot.inputmap), 4

## R

remove\_path() (guiobot.path.Path method), 5

## S

search() (guiobot.path.Path method), 5

## T

to\_string() (guiobot.inputmap.Key method), 2

to\_string() (guiobot.inputmap.KeyModifier method), 3

to\_string() (guiobot.inputmap.MouseButton method), 4

## U

UninitializedBackendError, 2

UnsupportedBackendError, 2

## V

VNCDoToolKey (class in guiobot.inputmap), 3

VNCDoToolKeyModifier (class in guiobot.inputmap), 3

VNCDoToolMouseButton (class in guiobot.inputmap), 4

## X

x (guiobot.location.Location attribute), 5

XDoToolKey (class in guiobot.inputmap), 3

XDoToolKeyModifier (class in guiobot.inputmap), 3

XDoToolMouseButton (class in guiobot.inputmap), 4

## Y

y (guiobot.location.Location attribute), 5