

---

# Portal Selenium Tests Documentation

*Release 0.1*

**End Point**

January 02, 2015



## CONTENTS

<b>1</b>	<b>tests/extensions/functional/tests package</b>	<b>3</b>
1.1	test_display_kiosk module . . . . .	3
1.2	test_general module . . . . .	4
1.3	test_google_menu module . . . . .	4
1.4	test_platform module . . . . .	5
1.5	test_ros module . . . . .	5
1.6	test_runway module . . . . .	6
1.7	test_zoom module . . . . .	6
	<b>Python Module Index</b>	<b>7</b>
	<b>Index</b>	<b>9</b>



Description of the test modules, classes and particular test cases in the **tests/extensions/functional/tests** portal project package.

The test cases have **test\_** prefix in their names.



---

## TESTS/EXTENSIONS/FUNCTIONAL/TESTS PACKAGE

### 1.1 test\_display\_kiosk module

The display extension tests.

The kiosk extension tests.

Cross-verification of DOM elements between the display and kiosk extensions.

**class** test\_display\_kiosk.**TestBaseDisplay**

Tests related to the display extension.

**test\_widgets\_not\_displayed** (\*args, \*\*kwargs)

**Test that the following graphical widgets (elements) are not displayed:**

- zoom in/out buttons,
- compass,
- google More Fun menu,
- search field,
- search button,
- runway panel (Points of Interest, Famous Places).

After the browser loads the initial URL, it takes some time until also the extensions are fully loaded and until the DOM is modified accordingly.

**class** test\_display\_kiosk.**TestBaseKiosk**

Tests related to the kiosk extension.

**test\_widgets\_displayed** (\*args, \*\*kwargs)

**Test that the following graphical widgets (elements) are displayed:**

- zoom in/out buttons,
- compass,
- google More Fun menu,
- search field,
- search button,
- runway panel (Points of Interest, Famous Places).

If elements from the elements list are renamed, the test `test_widgets_not_displayed()` would just test that they are not present, however they might be, but under a different name. Test that exactly the same are present when the display extension is not loaded, i.e. with kiosk extension loaded.

## 1.2 test\_general module

General Portal selenium tests.

**class** `test_general.TestMiscellaneous`

Other test cases not fitting any other current category.

**test\_eu\_cookies\_info\_bar\_is\_hidden** (\*args, \*\*kwargs)

Test that the EU cookies info bar is invisible.

And the map canvas has set top=0px, because the cookie container is 30 px higher, and the canvas is moved 30 px down.

**class** `test_general.TestSearch`

Test class related to the search box, search button. Various interaction scenarios.

**test\_no\_searchbox\_on\_other\_planets** (\*args, \*\*kwargs)

The test loads the initial config["maps\_url"] and zooms out. After this zoom out, the universe objects appear (Earth, Moon, Mars).

The search box should not be visible when Moon, Mars are clicked.

First Moon is clicked, disappearance of search box is verified. Second, the Earth is clicked, verify search box appeared. Last, Mars is clicked, disappearance of search box is verified.

**test\_search\_clicking\_search\_button** (\*args, \*\*kwargs)

Test that camera coordinates were changed from the initial position to something different after performing the search.

Interacts with search box and clicks on the search button.

**test\_search\_hitting\_return\_on\_search\_box** (\*args, \*\*kwargs)

Test that camera coordinates were changed from the initial position to something different after performing the search.

Interacts with search box and hits the return key on on the search box.

**test\_search\_hitting\_return\_on\_search\_button** (\*args, \*\*kwargs)

Test that camera coordinates were changed from the initial position to something different after performing the search.

Interacts with search box and sends return key event on the search button.

## 1.3 test\_google\_menu module

Google Menu related tests.

**class** `test_google_menu.TestGoogleMenu`

Google Menu tests.

**test\_clicking\_doodle\_item** (\*args, \*\*kwargs)

Test that clicking on the doodle item changes the URL to the doodles page.

**test\_google\_items\_are\_visible\_on\_click** (\*args, \*\*kwargs)

Test that Google Menu (More fun) items are visible after clicking it.



**test\_google\_menu\_is\_visible** (\*args, \*\*kwargs)

Test that Google Menu (More fun) is displayed along with some items.

## 1.4 test\_platform module

Browser platform tests (webgl, drivers etc).

**class** test\_platform.**TestPlatform**

Following tests will check whether:

- version of the browser matches the version in config.json

**Check:**

- gpu\_data.clientInfo.version => Chrome/37.0.2062.120
- gpu\_data.gpuInfo.basic\_info[n] => description : 'Direct Rendering'

**Log:**

- gpu\_data.gpuInfo.basic\_info[n] => description : GL\_VERSION

**test\_get\_chrome\_gpu\_data** ()

Check if the chrome supports GPU.

## 1.5 test\_ros module

Tests involving ROS communication between browsers.

**class** test\_ros.**TestBaseSingleBrowserROS**

Tests involving single browser.

Listening and asserting ROS traffic based on the actions performed in the (kiosk extension) browser. Browser runs in one process, another process is ROS topic listener.

**test\_ros\_position\_after\_search** (\*args, \*\*kwargs)

Run browser and type something in the search box, a place with a known position.

Helper background process listening on `/portal_kiosk/current_pose` ROS topic and checks ROS position messages until the one expected arrives (within certain timeout).

Pass/fail flag is set accordingly by the listener process via shared memory value which this test cases evaluates eventually.

**class** test\_ros.**TestBaseTwoBrowsersROS**

Tests involving two browsers.

General idea is to selenium-manipulate one browser (with kiosk extension) and assert accordingly adjusted state (as propagated through ROS) in the other browser (display extension).

**test\_ros\_positions\_aligned\_after\_search** (\*args, \*\*kwargs)

Perform search in the kiosk browser and assert on the automatically synchronized final position in the display browser.

## 1.6 test\_runway module

Tests related to Runway elements.

The Runway is the element at to bottom of the touchscreen browser, it contains a list of Points of Interests (if the camera is zoomed in to some level, and there are some interesting places around) or the list consisting of the Earth, Moon, Mars.

There is also a list of Famous Places (Earth tours), which is always filled (loaded by a static list of entries read from a file).

**class** `test_runway.TestRunway`

Runway element interaction tests.

**prepare\_poi** ()

Prepare for Points of Interest tests.

Load browser, search for a location with POIs and wait some time for the runway tray to populate.

**test\_runway\_buttons\_basic** (\*args, \*\*kwargs)

Test that Point of Interest and Famous Places are displayed.

**test\_runway\_check\_earth\_icon\_click** (\*args, \*\*kwargs)

The Earth icon (most left picture), clicking it should bring the view to a considerably zoomed out position.

NB: position object values differ between subsequent runs.

**test\_runway\_planets\_on\_max\_zoom\_out** (\*args, \*\*kwargs)

Test there is Mars, Earth, Moon loaded in Points of Interest tray on maximal zoom out.

**test\_runway\_points\_of\_interest** (\*args, \*\*kwargs)

Test Points of Interest is loaded, we can load one, exit it and and load another Point of Interest.

## 1.7 test\_zoom module

Tests related to Zoom buttons, zoom operation.

**class** `test_zoom.TestZoomButtons`

Tests for checking the zoom buttons are functional.

**test\_zoom\_buttons** (\*args, \*\*kwargs)

Test that the zoom in and out buttons are displayed.

**test\_zoom\_in\_button\_change** (\*args, \*\*kwargs)

Test clicks on the zoom out button and checks the pose object coordinates (altitude, latitude, longitude) according change.

**test\_zoom\_out\_button\_change** (\*args, \*\*kwargs)

Test clicks on the zoom in button and checks the pose object coordinates (altitude, latitude, longitude) according change.

**t**

test\_display\_kiosk, 3  
test\_general, 4  
test\_google\_menu, 4  
test\_platform, 5  
test\_ros, 5  
test\_runway, 6  
test\_zoom, 6



## P

prepare\_poi() (test\_runway.TestRunway method), 6

## T

test\_clicking\_doodle\_item()  
(test\_google\_menu.TestGoogleMenu method), 4

test\_display\_kiosk (module), 3

test\_eu\_cookies\_info\_bar\_is\_hidden()  
(test\_general.TestMiscellaneous method), 4

test\_general (module), 4

test\_get\_chrome\_gpu\_data() (test\_platform.TestPlatform method), 5

test\_google\_items\_are\_visible\_on\_click()  
(test\_google\_menu.TestGoogleMenu method), 4

test\_google\_menu (module), 4

test\_google\_menu\_is\_visible()  
(test\_google\_menu.TestGoogleMenu method), 5

test\_no\_searchbox\_on\_other\_planets()  
(test\_general.TestSearch method), 4

test\_platform (module), 5

test\_ros (module), 5

test\_ros\_position\_after\_search()  
(test\_ros.TestBaseSingleBrowserROS method), 5

test\_ros\_positions\_aligned\_after\_search()  
(test\_ros.TestBaseTwoBrowsersROS method), 5

test\_runway (module), 6

test\_runway\_buttons\_basic() (test\_runway.TestRunway method), 6

test\_runway\_check\_earth\_icon\_click()  
(test\_runway.TestRunway method), 6

test\_runway\_planets\_on\_max\_zoom\_out()  
(test\_runway.TestRunway method), 6

test\_runway\_points\_of\_interest()  
(test\_runway.TestRunway method), 6

test\_search\_clicking\_search\_button()  
(test\_general.TestSearch method), 4

test\_search\_hitting\_return\_on\_search\_box()  
(test\_general.TestSearch method), 4

test\_search\_hitting\_return\_on\_search\_button()  
(test\_general.TestSearch method), 4

test\_widgets\_displayed()  
(test\_display\_kiosk.TestBaseKiosk method), 3

test\_widgets\_not\_displayed()  
(test\_display\_kiosk.TestBaseDisplay method), 3

test\_zoom (module), 6

test\_zoom\_buttons() (test\_zoom.TestZoomButtons method), 6

test\_zoom\_in\_button\_change()  
(test\_zoom.TestZoomButtons method), 6

test\_zoom\_out\_button\_change()  
(test\_zoom.TestZoomButtons method), 6

TestBaseDisplay (class in test\_display\_kiosk), 3

TestBaseKiosk (class in test\_display\_kiosk), 3

TestBaseSingleBrowserROS (class in test\_ros), 5

TestBaseTwoBrowsersROS (class in test\_ros), 5

TestGoogleMenu (class in test\_google\_menu), 4

TestMiscellaneous (class in test\_general), 4

TestPlatform (class in test\_platform), 5

TestRunway (class in test\_runway), 6

TestSearch (class in test\_general), 4

TestZoomButtons (class in test\_zoom), 6