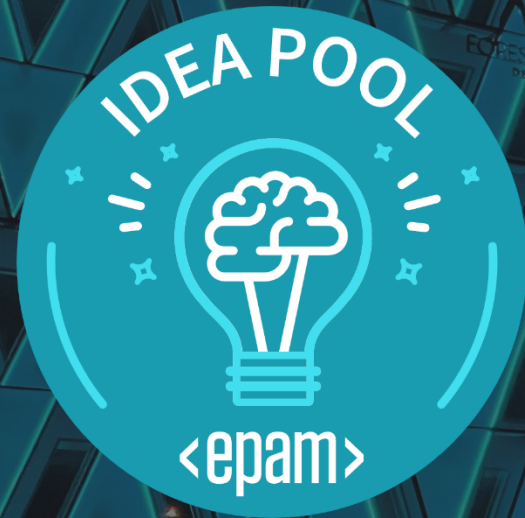


<epam>

TA TOOL EVALUATION — ROBOT FRAMEWORK

LASZLO SZIKSZAI

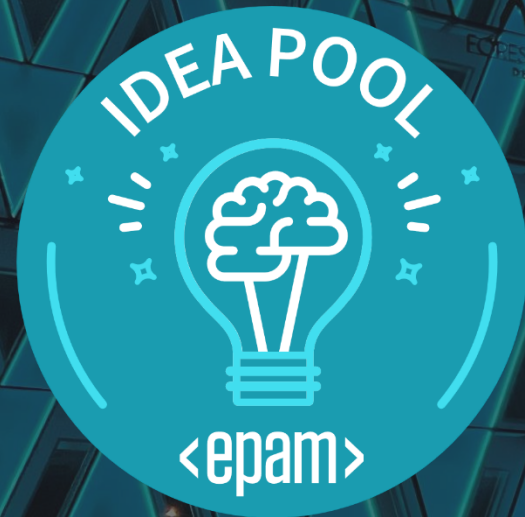


<epam>

TA TOOL EVALUATION — ROBOT FRAMEWORK

LASZLO SZIKSZAI

🕒 STARTING SOON...



ABOUT THE SPEAKER



LASZLO SZIKSZAI

SOLUTION ARCHITECT I, AUTOMATED TESTING

- 15 years in IT
- 7 years in EPAM
- 6 years in QA/Test Automation
- JavaScript/TypeScript/Python
- Open Source - GitHub/NPM/PyPi

AGENDA

- 1 WHAT IS THE ROBOT FRAMEWORK?
- 2 FEATURES
- 3 USAGE
- 4 PROS/CONS
- 5 WHEN TO/NOT TO USE IT?
- 6 FOR MORE INFO...



ROBOT FRAME WORK/

GETTING STARTED | RESOURCES | COMMUNITY | DEVELOPMENT

DOCS ▾ | EXTERNAL SITES ▾

🔊 [RoboCon 2022 tickets available!](#) 🔊

INTRODUCTION

Robot Framework is a generic open source automation framework. It can be used for test automation and robotic process automation (RPA).

Robot Framework is supported by [Robot Framework Foundation](#). Many industry-leading companies use the tool in their software development.

Robot Framework is open and extensible. Robot Framework can be integrated with virtually any other tool to create powerful and flexible automation solutions. Robot Framework is free to use without licensing costs.

Robot Framework has an easy syntax, utilizing human-readable

Tweets by @robotframework



Robot Framework
@robotframework



Last call for the [#RoboCon2022](#) Early Bird -tickets! We will raise the prices today at 6 pm GMT.

You do not want to miss this, check the agenda and purchase your tickets now: [robocon.io/#RoboCon2022](#) [#RoboCon](#) [#Community](#) [#RobotFramework](#)



Feb 25, 2022



Robot Framework
@robotframework



Community time tomorrow 2pm GMT! See you there!

On the agenda: [@MateuszNojek13](#) and [Bartlomiej Hirs](#) present their [#RobotFramework](#) Ecosystem project

LIBRARIES

BUILT-IN

TOOLS

Separately developed external libraries that can be installed based on your needs. Creating your own libraries is a breeze. For instructions, see [creating test libraries](#) in Robot Framework User Guide.

Name	Description	Stars ▼	Tags
SeleniumLibrary	Web testing library that uses popular Selenium tool internally.	1110	WEB, SELENIUM
HTTP RequestsLibrary (Python)	HTTP level testing using Python Requests internally.	381	HTTP
AppiumLibrary	Android and iOS testing. Uses Appium internally.	290	MOBILE
Browser Library	A modern web testing library powered by Playwright . Aiming for speed, reliability and visibility.	211	WEB
RESTinstance	Test library for HTTP JSON APIs.	172	HTTP
Database Library (Python)	Python based library for database testing. Works with any Python interpreter, including Jython.	117	DB
SikuliLibrary	Provides keywords to test UI through Sikulix . This library supports Python 2.x and 3.x.	117	UI

[SHOW MORE](#)

LIBRARIES

BUILT-IN

TOOLS

Libraries and tools that are bundled with the framework. Libraries provide the actual automation and testing capabilities to Robot Framework by providing keywords.

Name ▾	Description	Tags
Builtin	Provides a set of often needed generic keywords. Always automatically available without imports.	LIBRARY
Collections	Provides a set of keywords for handling Python lists and dictionaries.	LIBRARY
DateTime	Library for date and time conversions.	LIBRARY
Dialogs	Provides means for pausing the execution and getting input from users.	LIBRARY
Libdoc	Generate keyword documentation for test libraries and resource files.	TOOL
OperatingSystem	Enables various operating system related tasks to be performed in the system where Robot Framework is running.	LIBRARY
Process	Library for running processes in the system.	LIBRARY

SHOW MORE

LIBRARIES

BUILT-IN

TOOLS

Supporting tools ease automation: editing, running, building and so on. Most of these tools are developed as separate projects, but some are built into the framework itself.

Filter by tag

Name	Description	Stars ▾	Tags
RIDE	Standalone Robot Framework test data editor.	795	EDITOR
Pabot	A parallel executor for Robot Framework tests and tasks.	378	
RED	Eclipse based editor with a debugger by Nokia.	327	EDITOR
RCC	Share your Robot projects with ease. RCC allows you to create, manage, and distribute Python-based self-contained automation packages.	238	
Robot Framework Hub	Lightweight web server that provides access to the Robot Framework test assets via browser.	156	
Sublime assistant	A plugin for Sublime Text 2 & 3 by Andriy Hrytskiv.	109	EDITOR
Vim_plugin	Vim plugin for development with Robot Framework.	86	EDITOR

SHOW MORE

Robot Framework Quick Start Guide

Copyright © Nokia Networks. Licensed under the [Creative Commons Attribution 3.0 Unported](#) license.

Table of contents:

- Introduction
 - About this guide
 - Robot Framework overview
 - Demo application
- Executing this guide
 - Installations
 - Execution
 - Viewing results
- Test cases
 - Workflow tests
 - Higher-level tests
 - Data-driven tests
- Keywords
 - Library keywords
 - User keywords

```
QuickStart.rst x
1  .. default-role:: code
2
3  =====
4  Robot Framework Quick Start Guide
5  =====
6
7  Copyright © Nokia Networks. Licensed under the
8  `Creative Commons Attribution 3.0 Unported`__ license.
9
10 __ http://creativecommons.org/licenses/by/3.0/
11
12 .. contents:: Table of contents:
13    :local:
14    :depth: 2
15
16 Introduction
17 =====
18
19 About this guide
20 -----
21
22 *Robot Framework Quick Start Guide* introduces the most important `Robot
23 Framework <http://robotframework.org>`_ features. You can simply browse
24 through it and look at the examples, but you can also use the guide as an
25 `executable demo`__. All features shown here are explained more thoroughly
26 in `Robot Framework User Guide`_.
27
28 __ `Executing this guide`_
29 .. _Robot Framework User Guide: http://robotframework.org/robotframework/#user-guide
30
31 Robot Framework overview
32 -----
33
34 `Robot Framework`_ is a generic open source test automation framework for
```

```
QuickStart.rst x
184 =====
185
186 Workflow tests
187 -----
188
189 Robot Framework test cases are created using a simple tabular syntax. For
190 example, the following table has two tests:
191
192 - User can create an account and log in
193 - User cannot log in with bad password
194
195 .. code:: robotframework
196
197     *** Test Cases ***
198     User can create an account and log in
199         Create Valid User    fred    P4ssw0rd
200         Attempt to Login with Credentials    fred    P4ssw0rd
201         Status Should Be    Logged In
202
203     User cannot log in with bad password
204         Create Valid User    betty    P4ssw0rd
205         Attempt to Login with Credentials    betty    wrong
206         Status Should Be    Access Denied
207
208 Notice that these tests read like manual tests written in English rather
```



szikszail Updated workflow

7ba6fc6 1 minute ago 8 commits

folder	.github/workflows	Updated workflow	1 minute ago
folder	data	Implementation (#1)	20 minutes ago
folder	resources	Implementation (#1)	20 minutes ago
folder	schema	Implementation (#1)	20 minutes ago
folder	test_cases	Implementation (#1)	20 minutes ago
file	.gitattributes	Initial commit	last month
file	.gitignore	Implementation (#1)	20 minutes ago
file	LICENSE.md	Initial commit	last month
file	README.md	Implementation (#1)	20 minutes ago
file	requirements.txt	Implementation (#1)	20 minutes ago

This repository contains tests implemented using the Robot Framework.

python evaluation test-automation robotframework ideapool

Readme

MIT License

0 stars

1 watching

0 forks

Languages

RobotFramework 90.8%
JavaScript 9.2%

evaluation-py-robot

This repository contains tests implemented using the Robot Framework.

Prerequisites

1. Python 3.8+
2. Pip

Framework used

- Name: Robot Framework
- Home page: <https://robotframework.org/>
- User Guide: <https://robotframework.org/robotframework/latest/RobotFrameworkUserGuide.html>
- Libraries: <https://robotframework.org/?tab=libraries#resources>
 - Standard libraries: <http://robotframework.org/robotframework/#standard-libraries>
 - robotframework-requests: <https://github.com/MarketSquare/robotframework-requests>
 - robotframework-jjsonschema: <https://github.com/jstaffans/robotframework-jjsonschema>
 - robotframework-selenium2library: <https://github.com/robotframework/SeleniumLibrary>
 - RESTinstance: <https://github.com/asyrjasalo/RESTinstance/>

API

Test cases

Precondition

1. Given the API key is set for all requests

TC-1 A movie can be retrieved by ID

1. Given the {movie_id} is added to the GET request
2. When the "GET movie by id" request is sent
3. Then the response should contain status code 200

TC-2 The retrieved movie has a proper schema

1. Given the {movie_id} is added to the GET request
2. When the "GET movie by id" request is sent
3. Then the response should contain the proper movie title
4. And the response should have a proper schema

```
1  *** Settings ***
2  Library           RequestsLibrary
3  Library           JSONSchemaLibrary    schema
4  Library           Process
5  Force Tags       level:api
6  Default Tags     type:positive
7  Suite Setup      Run Keywords    Create A Guest Session
8  ...              AND    Load Movie
9  Resource         ../../resources/data.resource
10
11 *** Variables ***
12 ${API_KEY}       e231ae412af0eaab685d11c26505bda6
13 ${BASE_URL}      https://api.themoviedb.org/3
14 ${MOVIE}         Load JSON From File    data/movie/76341
15 ${MOVIE_ID}      76341
16 ${MOVIE_TITLE}   Mad Max: Fury Road
17
18 *** Test Cases ***
19 TC-01 A movie can be retrived by ID
20     Get A Movie By ID    ${MOVIE}[id]    200
21
22 TC-02 The retrieved movie has a proper schema
23     ${response}=    Get A Movie By ID    ${MOVIE}[id]
24     Should Be Equal As Strings    ${MOVIE}[title]    ${response.json()[title]}
25     Validate Json    movie.json    ${response.json()}
26
27 TC-03 A rating can be added to a movie
28     [Template]    Rate A Movie
29     FOR    ${rate} IN RANGE    1    11
30         ${MOVIE}[id]    ${rate}    201
31     END
32
```

```

54 [Type] type-negative
55 Get A Movie By ID    ${null}    404
56
57 *** Keywords ***
58 Get A Movie By ID
59 [Arguments]    ${movie_id}    ${status}=200
60 ${response}=    GET    ${BASE_URL}/movie/${movie_id}
61 ...    params=api_key=${API_KEY}
62 ...    expected_status=${status}
63 [Return]    ${response}
64
65 Rate A Movie
66 [Arguments]    ${movie_id}    ${rating}    ${status}=201
67 &{params}=    Create Dictionary    api_key=${API_KEY}    guest_session_id=${GUEST_SESSION_ID}
68 &{body}=    Create Dictionary    value=${rating}
69 ${response}=    POST    ${BASE_URL}/movie/${movie_id}/rating
70 ...    params=${params}
71 ...    json=${body}
72 ...    expected_status=${status}
73 [Return]    ${response}
74
75 Delete Movie Rating
76 [Arguments]    ${movie_id}    ${status}=200
77 &{params}=    Create Dictionary    api_key=${API_KEY}    guest_session_id=${GUEST_SESSION_ID}
78 ${response}=    DELETE    ${BASE_URL}/movie/${movie_id}/rating
79 ...    params=${params}
80 ...    expected_status=${status}
81 [Return]    ${response}
82

```

Project ▾

- evaluation-py-robot C:\Repo\evaluation-py-robot
 - .github
 - .vscode
 - data
 - movie
 - 76341.json
 - selectors
 - texts
 - resources
 - data.resource
 - element.resource
 - is_in_viewport.js
 - scroll_by_page.js
 - schema
 - test_cases
 - venv library root
 - .gitattributes
 - .gitignore
 - LICENSE.md
 - log.html
 - output.xml
 - README.md
 - report.html
 - requirements.txt
 - selenium-screenshot-1.png

data.resource x

```
1 *** Settings ***
2 Library      JSONLibrary
3
4 *** Keywords ***
5 Load Data
6 [Arguments]  ${type} ${id}
7 ${json}=    Load JSON From File  ${EXECDIR}/${data}/${type}/${id}.json
8 [Return]    ${json}
9
```

76341.json x

```
1 {
2   "id": 76341,
3   "title": "Mad Max: Fury Road"
4 }
```


Project

api_movie_db.robot

evaluation-py-robot C:\Repo\evaluation-py-robot

- .github
- .vscode
- data
- resources
- schema
 - movie.json
- test_cases
 - api
 - API.md
 - api_movie_db.robot
 - ui
- venv library root
 - .gitattributes
 - .gitignore
 - LICENSE.md
 - log.html
 - output.xml
 - README.md
 - report.html
 - requirements.txt
 - selenium-screenshot-1.png
 - selenium-screenshot-2.png
 - selenium-screenshot-3.png
 - selenium-screenshot-4.png
 - selenium-screenshot-5.png
- External Libraries
- Scratches and Consoles

```
Library RequestsLibrary
Library JSONSchemaLibrary schema
Library Process
Force Tags level:api
Default Tags type:positive
Suite Setup Run Keywords Create A Guest Session
... AND Load Movie
Resource ../../resources/data.resource

*** Variables ***
12 ${API_KEY} e231ae412af0eaab685d11c26505bda6
13 ${BASE_URL} https://api.themoviedb.org/3
14 ${MOVIE} Load JSON From File data/movie/76341
15 ${MOVIE_ID} 76341
16 ${MOVIE_TITLE} Mad Max: Fury Road

*** Test Cases ***
18 TC-01 A movie can be retrieved by ID
19 Get A Movie By ID ${MOVIE}[id] 200
21
22 TC-02 The retrieved movie has a proper schema
23 ${response}= Get A Movie By ID ${MOVIE}[id]
24 Should Be Equal As Strings ${MOVIE}[title] ${response.json()[title]}
25 Validate Json movie.json ${response.json()}

movie.json
1 {
2   "$schema": "https://json-schema.org/draft/2020-12/schema",
3   "$id": "https://epa.ms/ideapool/schema/movie.schema.json",
4   "title": "Movie",
5   "description": "The details of a movie from TMDB.org",
6   "type": "object",
7   "properties": {
8     "adult": {
9       "type": "boolean"
```

UI

Test cases

TC-1 Checking landing pages elements

1. **Given** <https://angular.io> URL is opened
2. **Then** Angular logo in the top navbar should be visible
3. **And** Angular logo in the hero section should be visible
4. **And** text in hero section should be "One framework. Mobile & desktop."
5. **And** Get started button should be visible in the hero section
6. **When** Get started button is clicked in the hero section
7. **Then** the URL should be <https://angular.io/start>
8. **And** the title on the content should be "Getting Started with Angular: Your First App"

TC-2 Checking search field on the landing page

1. **Given** <https://angular.io> URL is opened
2. **Then** Search input in the top navbar should be visible
3. **And** it should be empty
4. **And** it should be "Search" as a placeholder

Project

angular.robot

evaluation-py-robot

github

vscode

data

resources

schema

movie.json

test_cases

api

ui

angular.robot

bootstrap.robot

UI.md

venv library root

gitattributes

gitignore

LICENSE.md

log.html

output.xml

README.md

report.html

requirements.txt

selenium-screenshot-1.png

selenium-screenshot-2.png

selenium-screenshot-3.png

selenium-screenshot-4.png

selenium-screenshot-5.png

External Libraries

Scratches and Consoles

angular.robot

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

*** Settings ***

Library SeleniumLibrary

Library String

Force Tags level:ui

Default Tags type:positive

Suite Setup Load Page Data

Test Teardown Close Browser

Resource ../../resources/data.resource

*** Variables ***

\${BASE_URL} https://angular.io

\${BROWSER} Chrome

*** Test Cases ***

TC-01 Checking landing page elements

Given Angular.io is opened

Then Element Should Be Visible \${PAGE}[angular_logo_navbar]

And Element Should Be Visible \${PAGE}[angular_logo_hero]

And Element Text Should Be \${PAGE}[hero_text] \${TEXTS}[hero]

And Element Should Be Visible \${PAGE}[hero_button]

When Click Element \${PAGE}[hero_button]

And Wait Until Element Is Visible \${PAGE}[title]

Then Location Should Be \${BASE_URL}/start

And Element Text Should Be \${PAGE}[title] \${TEXTS}[getting_started_title]

TC-02 Checking search field on the landing page

Given Angular.io is opened

Then Element Should Be Visible \${PAGE}[search_input]

And Element Text Should be \${PAGE}[search_input] \${EMPTY}

And Element Attribute Value Should Be \${PAGE}[search_input] placeholder Search

When Click Element \${PAGE}[search_input]

And Input Text \${PAGE}[search_input] directive

And Click Element \${PAGE}[search_input]

Then Element Should Be Visible \${PAGE}[search_clear icon]

```

41 *** Keywords ***
42 Angular.io is opened
43     Open Browser    ${BASE_URL}    ${BROWSER}
44     Maximize Browser Window
45
46 Load Page Data
47     ${selectors}= Load Data type=selectors id=angular
48     Set Suite Variable    ${PAGE}    ${selectors}
49     ${texts}= Load Data type=texts id=angular
50     Set Suite Variable    ${TEXTS}    ${texts}
51
52 Menu Item Should Be In Section
53     [Arguments]    ${item}    ${section}
54     Wait Until Element Is Visible    ${PAGE}[search_results]
55     ${section_selector}= Replace String    ${PAGE}[section_by_text]    TEXT    ${section}
56     Wait Until Element Is Visible    ${section_selector}
57     ${item_selector}= ReplaceString    ${PAGE}[item_by_text]    TEXT    ${item}
58     Element Should Be Visible    ${section_selector} >> ${item_selector}
59
60 Menu Item Is Clicked In Section
61     [Arguments]    ${item}    ${section}
62     Wait Until Element Is Visible    ${PAGE}[search_results]
63     ${section_selector}= Replace String    ${PAGE}[section_by_text]    TEXT    ${section}
64     Wait Until Element Is Visible    ${section_selector}
65     ${item_selector}= ReplaceString    ${PAGE}[item_by_text]    TEXT    ${item}
66     Click Element    ${section_selector} >> ${item_selector}
67     Wait Until Location Contains    ${item}

```


Project

evaluation-py-robot C:\Repo\evaluation-py-robot

> .github

> .vscode

> data

> resources

> schema

movie.json

> test_cases

> api

> ui

angular.robot

bootstrap.robot

UI.md

> venv library root

.gitattributes

.gitignore

LICENSE.md

log.html

output.xml

README.md

report.html

requirements.txt

selenium-screenshot-1.png

selenium-screenshot-2.png

selenium-screenshot-3.png

selenium-screenshot-4.png

selenium-screenshot-5.png

> External Libraries

Scratches and Consoles

bootstrap.robot

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

*** Settings ***

Library SeleniumLibrary implicit_wait=2

Library OperatingSystem

Library String

Force Tags level:ui

Default Tags type:positive

Suite Setup Load Page Data

Test Teardown Close Browser

Resource ../../resources/data.resource

Resource ../../resources/element.resource

*** Variables ***

_\${BASE_URL} https://getbootstrap.com/docs/4.4

_\${BROWSER} Chrome

*** Test Cases ***

TC-03 Checking form elements

Given Bootstrap is opened forms

There is some encoding issue with JSON file with the "." character

Then Title Should Be \${TEXTS}[forms_title]

Then Title Should Be Forms · Bootstrap

And Element Should Not Be In View \${PAGE}[readonly_input]

When Scroll Element Into View \${PAGE}[readonly_input]

And Sleep 2s

Then Element Should Be In View \${PAGE}[readonly_input]

And Element Should Be ReadOnly \${PAGE}[readonly_input]

TC-04 Interaction with checkbox form elements

Given Bootstrap is opened forms checkboxes-and-radios

Then Element Should Be Enabled \${PAGE}[default_checkbox]

And Element Should Be Disabled \${PAGE}[disabled_checkbox]

And Checkbox Should Not Be Selected \${PAGE}[default_checkbox]

When Select Checkbox \${PAGE}[default_checkbox]

Then Checkbox Should Be Selected \${PAGE}[default_checkbox]

```
35
36 TC-05 Interaction with radio form elements
37   Given Bootstrap is opened forms checkboxes-and-radios
38   Then Element Should Be Enabled  ${PAGE}[default_radio]
39   And Element Should Be Disabled  ${PAGE}[disabled_radio]
40   And Radio Button Should Be Set To  ${TEXTS}[radio_group] option1
41   # A radio button cannot be checked if selected
42   # it is derived from the radio value
43   # And Radio Button Should Not Be Selected  ${PAGE}[second_default_radio]
44   When Click Element  ${PAGE}[second_default_radio]
45   # A radio button cannot be checked if selected
46   # it is derived from the radio value
47   # Then Radio Button Should Not Be Selected  ${PAGE}[default_radio]
48   Then Radio Button Should Be Set To  ${TEXTS}[radio_group] option2
49
```

Project | element.resource

evaluation-py-robot C:\Repo\evaluati

- > .github
- > .vscode
- > data
- > resources
 - data.resource
 - element.resource
 - is_in_viewport.js
 - scroll_by_page.js
- > schema
 - movie.json
- > test_cases
 - > api
 - > ui
- > venv library root
 - .gitattributes
 - .gitignore
 - LICENSE.md
 - log.html
 - output.xml
 - README.md
 - report.html
 - requirements.txt
 - selenium-screenshot-1.png
 - selenium-screenshot-2.png
 - selenium-screenshot-3.png
 - selenium-screenshot-4.png

1 *** Settings ***

2 Library SeleniumLibrary implicit_wait=2

3 Library OperatingSystem

4 Library String

5 Library Collections

6

7 *** Keywords ***

8 Is Element In View

9 [Arguments] \${element}

10 \${js}= Get File resources\${/}is_in_viewport.js

11 \${visible}= Execute JavaScript \${js} return isInViewport(arguments[0]); ARGUMENTS \${element}

12 [Return] \${visible}

13

14 Page Is Scrolled Down By Page

15 [Arguments] \${n}=1

16 \${js}= Get File resources\${/}scroll_by_page.js

17 Execute JavaScript \${js} scrollByPage(arguments[0]); ARGUMENTS \${n}

18

19 Element Should Be In View

20 [Arguments] \${element}

21

is_in_viewport.js

1 /*

2 * Obtained from: https://gomakethings.com/how-to-test-if-an-element-is-in-the-viewport-with-vanilla-javascript/

3 * And updated to handle partial containment.

4 */

5 var isInViewport = function (elem) {

6 if (typeof elem === "string") {

7 elem = document.querySelector(elem.replace("css:", ""))

8 }

9 var bounding = elem.getBoundingClientRect();

10 var top = 0, bottom = window.innerHeight || document.documentElement.clientHeight;

REPORT

Api Movie Db Report

Generated
20220226 13:06:53 UTC+01:00
27 minutes 50 seconds ago

LOG

Summary Information

Status: All tests passed
Start Time: 20220226 13:06:47.148
End Time: 20220226 13:06:53.143
Elapsed Time: 00:00:05.995
Log File: [log.html](#)

Test Statistics

Total Statistics	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
All Tests	8	8	0	0	00:00:05	<div></div>

Statistics by Tag	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
level:api	8	8	0	0	00:00:05	<div></div>
type:negative	2	2	0	0	00:00:01	<div></div>
type:positive	6	6	0	0	00:00:05	<div></div>

Statistics by Suite	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
Api Movie Db	8	8	0	0	00:00:06	<div></div>

Test Details

All Tags Suites Search

Suite:
Status: 8 tests total, 8 passed, 0 failed, 0 skipped
Start / End Time: 20220226 13:06:47.148 / 20220226 13:06:53.143
Elapsed Time: 00:00:05.995
Log File: [log.html#s1](#)

Name	Documentation	Tags	Status	Message	Elapsed	Start / End
Api Movie Db. TC-01 A movie can be retrived by ID		level:api, type:positive	PASS		00:00:00.187	20220226 13:06:47.795 20220226 13:06:47.982
Api Movie Db. TC-02 The retrieved movie has a proper schema		level:api, type:positive	PASS		00:00:00.164	20220226 13:06:47.982 20220226 13:06:48.146
Api Movie Db. TC-03 A rating can be added to a movie		level:api, type:positive	PASS		00:00:03.161	20220226 13:06:48.146 20220226 13:06:51.307
Api Movie Db. TC-04 The movie rating should return a proper message		level:api, type:positive	PASS		00:00:00.268	20220226 13:06:51.308 20220226 13:06:51.576
Api Movie Db. TC-05 Invalid rating cannot be aded to a movie		level:api, type:negative	PASS		00:00:00.260	20220226 13:06:51.576 20220226 13:06:51.836
Api Movie Db. TC-06 Movie rating can be deleted		level:api, type:positive	PASS		00:00:00.504	20220226 13:06:51.837 20220226 13:06:52.341
Api Movie Db. TC-07 Proper status message should be returned when rating is deleted		level:api, type:positive	PASS		00:00:00.513	20220226 13:06:52.350 20220226 13:06:52.863
Api Movie Db. TC-08 Missing ID should be handled properly		level:api, type:negative	PASS		00:00:00.274	20220226 13:06:52.863 20220226 13:06:53.137

Total Statistics	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
All Tests	8	8	0	0	00:00:05	
Statistics by Tag	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
level:api	8	8	0	0	00:00:05	
type:negative	2	2	0	0	00:00:01	
type:positive	6	6	0	0	00:00:05	
Statistics by Suite	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
Api Movie Db	8	8	0	0	00:00:06	

SUITE

Api Movie Db

Full Name:

Api Movie Db

Source:

[c:\Repo\evaluation-py-robot\test_cases\api\api_movie_db.robot](#)

Start / End / Elapsed:

20220226 13:06:47.148 / 20220226 13:06:53.143 / 00:00:05.995

Status:

8 tests total, 8 passed, 0 failed, 0 skipped

SETUP

Builtin

Run Keywords

Create A Guest Session, AND, Load Movie

TEST

TC-01 A movie can be retrived by ID

Full Name:

Api Movie Db.TC-01 A movie can be retrived by ID

Tags:

level:api, type:positive

Start / End / Elapsed:

20220226 13:06:47.795 / 20220226 13:06:47.982 / 00:00:00.187

Status:

PASS

KEYWORD

Get A Movie By ID \${MOVIE}[id], 200

Start / End / Elapsed:

20220226 13:06:47.795 / 20220226 13:06:47.982 / 00:00:00.187

KEYWORD

\$(response) = RequestsLibrary.GET \${BASE_URL}/movie/\${movie_id}, params=api_key=\${API_KEY}, expected_status=\${status}

Documentation:

Sends a GET request.

Start / End / Elapsed:

20220226 13:06:47.795 / 20220226 13:06:47.982 / 00:00:00.187

13:06:47.982

INFO

GET Request : url=https://api.themoviedb.org/3/movie/76341?api_key=e231ae412af0eaab685d11c26505bda6
 path_url=/3/movie/76341?api_key=e231ae412af0eaab685d11c26505bda6
 headers={'User-Agent': 'python-requests/2.27.1', 'Accept-Encoding': 'gzip, deflate', 'Accept': '*/*', 'Connection': 'keep-alive'}
 body=None

13:06:47.982

INFO

GET Response : url=https://api.themoviedb.org/3/movie/76341?api_key=e231ae412af0eaab685d11c26505bda6
 status=200, reason=OK
 headers={'Content-Type': 'application/json;charset=utf-8', 'Transfer-Encoding': 'chunked', 'Connection': 'keep-alive', 'Date': 'Sat, 26 Feb 2022 11:44:43 GMT', 'Server': 'nginx', 'Access-Control-Allow-Origin': '*', 'Access-Control-Allow-Headers': '*', 'Access-Control-Expose-Headers': 'ETag, X-RateLimit-Limit, X-RateLimit-Remaining, X-RateLimit-Reset, Retry-After, X-Memc', 'Hit': 'X-Memc-key': '56eeae205cdab68212e8fb99865ebbd99d22d47ab', 'X-Memc-Age': '10385', 'X-Memc-Expires': '14828', 'ETag': 'W/"a63919a3baad5913fb83547e6eed945c"', 'X-CDN': 'CloudFront', 'Via': '1.1 71899bd3f76489e8a6e71cc77aaa6424.cloudfront.net (CloudFront)', 'X-Amz-Cf-Pop': 'TXL50-P3', 'X-Amz-Cf-Id': 'zu0cWu_Kg3JKNNdS3AYSdHKQPLYNVbodueCfAdult': false, 'hackeron_path': '/nlCHUHV8XvWbU5U0auC8vP8V8v5.ing', 'belongs_to_collection': 'CfId': '8946', 'name': 'Mad Max'

PROS / CONS

PROS

- ✓ Detailed report
- ✓ General test automation framework for all levels and test types: UI, API, Unit, Hardware, etc.
- ✓ Huge set of libraries and tools
- ✓ Keyword and Data Driven
- ✓ Fits for non-technical team as well
- ✓ Multiple formats to define tests:
Documentation and tests integrated

CONS

- ✗ Bigger learning curve to use effectively
- ✗ Python (and JS) knowledge is needed for more complex tests

WHEN TO USE IT?

- ✓ Internal, Enterprise, and POC projects
- ✓ TAF covering multiple test automation types and test levels
- ✓ Team is less technical, or test automation visibility and easy understandability is necessary
- ✓ ...



WHEN NOT TO USE IT?

- ✗ Test automation has to be done in other than Python
- ✗ Team is highly technical



MORE INFO

- **Idea Pool Community:** <https://epa.ms/idea-pool>
- **TA Tool evaluation project:**
<https://epa.ms/idea-pool-tool-evaluation>
- **TA Tool evaluation playlist:**
<https://epa.ms/idea-pool-tool-evaluation-playlist>
- **Robot Framework:**
<https://robotframework.org/>
- **Robot Framework User Guide:**
<https://robotframework.org/robotframework/latest/RobotFrameworkUserGuide.html>
- **Evaluation repository:**
<https://github.com/Idea-Pool/evaluation-py-robot>



QUESTIONS?



THANK YOU!

Thanks!

