



# VILT Minium Manager

## User Guide

Version 2.4



# Contents

<b>1. Introduction</b> .....	<b>1</b>
<b>2. Projects</b> .....	<b>2</b>
2.1. Monitoring .....	2
2.1.1. Check if the website is ok .....	3
2.1.2. Do a search on the website with refresh and check how long it took to load. . .	4
2.1.3. Click on a link and get the performance of the page .....	4
2.2. Cookie .....	4
2.3. Setup a project .....	5
2.4. Project permissions .....	13
2.5. Organise projects by Labels or Groups .....	15
2.5.1. Manage Labels .....	15
2.5.2. Assign labels to the projects .....	16
2.5.3. Manage groups .....	17
Manage groups using LDAP authentication .....	18
Manage groups without LDAP authentication .....	18
2.5.4. Organise Projects .....	20
Clean project filters .....	21
Auto Refresh .....	22
2.6. Configuration profiles .....	22
2.7. Secret configuration properties .....	23
2.8. Delete a project .....	24
<b>3. Check the available browsers</b> .....	<b>26</b>
<b>4. Project Statistics</b> .....	<b>27</b>
<b>5. Test executions</b> .....	<b>29</b>
5.1. Launch test executions .....	29
5.1.1. Launch a test execution manually. ....	29
Multiple Providers .....	31
5.1.2. Schedule test executions .....	32
Multiple Providers .....	34
5.1.3. REST API .....	34
5.2. Follow the progress of a test execution. ....	37
<b>6. Monitor test results</b> .....	<b>38</b>
6.1. Feature overview .....	47
6.2. Reports .....	48
6.3. Daily Reports .....	49
6.4. Data Extraction Reports .....	50
<b>7. Roles</b> .....	<b>52</b>
7.1. Assign roles .....	52
7.2. License information .....	53
<b>Minium Recorder User Guide</b> .....	<b>54</b>
<b>8. Configure Minium Developer</b> .....	<b>55</b>
<b>9. Launch Minium Recorder</b> .....	<b>56</b>

<b>10. Start recording</b> .....	<b>58</b>
<b>11. Record interactions</b> .....	<b>59</b>
<b>12. Stop recording</b> .....	<b>61</b>
<b>13. Import the recorded script in Minium Developer</b> .....	<b>62</b>

## 1. Introduction

This document describes the user interface of the VILT - Minium Manager, version 2.4.

**Minium Manager** is a platform to manage, run and analyze end-to-end tests, designed to run tests across all possible combination of browsers and OS's on a continuous integration fashion. It allows managers to control the quality of a project, providing detailed reports with useful information for failures with errors messages and screenshots.

## 2. Projects

Configure projects in Minium Manager, in order to run and analyze end-to-end tests. All projects need to have a repository (`Git` or `SVN`) associated, where the code is stored, except the Cookie Crawler.

There are three type of projects: Web application testing, Monitoring and Cookie.

The Web application testing project is a Minium project that test a website in the same way a human would.

The Monitoring project is a Minium project that are executed with high frequency (every 5, 10, 20 minutes), and can give us a report about the availability of a certain page and performance metrics, with the objective to check that the page are working as expected.

The development workflow for a Monitoring project in Minium Developer will be similar to the web application testing project, providing an easy and quick way to write test. The configuration in Minium Manager is similar to other projects. The monitoring project does not have the browser configuration.

The Cookie project generates a report of cookies found during the navigation or crawl of a website. The Cookie is divided in two projects: Cookie Report project and Cookie Crawler project. The Cookie Report project is a Minium project (based on the monitoring project) that extracts the all browser cookies at the end of a scenario. The Cookie Crawler project is a crawler that given one or more URLs, it crawls the websites and, for each URL, extracts the all browser cookies.

### 2.1. Monitoring

To start the development of a monitoring project, you need to check the checkbox `Is a Monitoring Project?` to generate the monitoring project:

**New project cucumber**

---

**Project type** Cucumber Project Automator Project

---

**Parent Directory**

Parent Directory

---

**Project Name**

Project Name

---

**Is a Monitoring Project?** This configuration will generate a different Cucumber Project, focused to retrieve information about the availability of a certain page and performance metrics.

---

Advanced
Create
Cancel

Here are some examples of the development workflow in Minium Developer of the monitoring project:

### 2.1.1. Check if the website is ok

Use case: You want to test if the URL is ok and get the performance in the reports

Scenario:

```
Scenario: Check Blog La caixa
  When Check if website responds: "https://blog.caixabank.es/"
```

Step:

```
When(/^Check if website responds: "[^"]*"$/, function(url) {
  browser.get(url);
});
```

For a monitoring project, the expression `browser.get(url);` retrieves the data related to page load performance, the URL status, the number of requests, the page size and javascript errors. This information will be presented at the Minium Manager report.

## 2.1.2. Do a search on the website with refresh and check how long it took to load.

Use case: You want to perform a search on the website (where's there's a refresh after the search) and check if the results appear and get the performance of the page.

Scenario:

```
Scenario: Search
  Given I'm at
  "https://www.caixabank.es/particular/home/particulares_es.html"
  When Search for "CaixaBank" and check the results
```

Steps:

```
When(/^I'm at "([^"]*)"$/, function(url) {
  browser.get(url);
});

When(/^Search for "([^"]*)" and check the results$/, function(search) {
  $("#cookies-accept-full a").click();
  $("#search-field").fill(search);
  $("#prebuscadorCabecera input[type='submit']").click();
  expect($(".search-result-block").to.exist());
  scenario.write(browser.getPerformance());
});
```

The expression `scenario.write(browser.getPerformance());` retrieves and store the data related to the last page load performance, the URL status, the number of requests, the page size and javascript errors. This information will be presented at the Minium Manager report.

## 2.1.3. Click on a link and get the performance of the page

```
When(/^I click on link with text "([^"]*)"$/, function(text) {
  $("a").withText(text).click();
  expect($(".page-title").to.exist());
  scenario.write(browser.getPerformance());
});
```

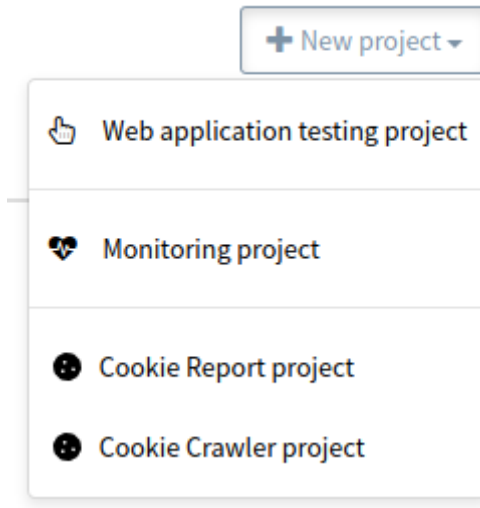
## 2.2. Cookie

Since the Cookie Report project is a Minium project (based on the monitoring project), to create a Cookie Report project, simply generate a Monitoring project in Minium Developer. It is not necessary to add extra instructions to the minium code, since all browser cookies will be automatically extracted at the end of each scenario.

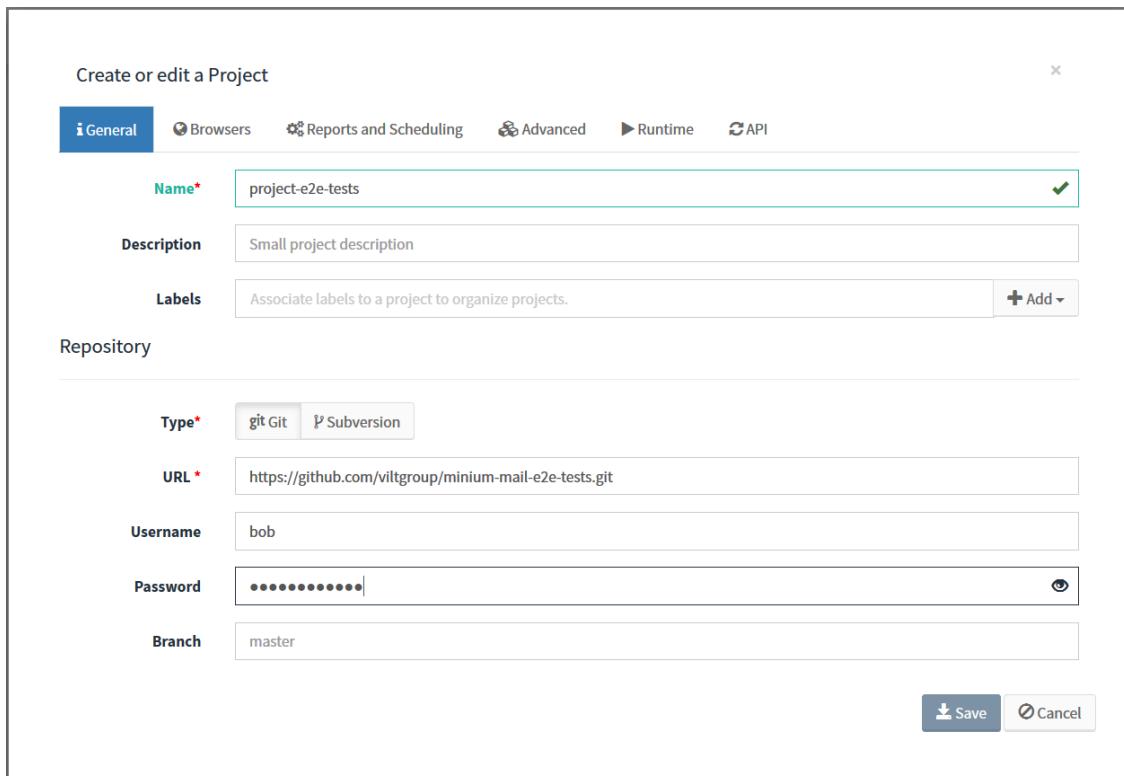
For the Cookie Crawler project, development is not necessary, you only need to provide one or more URLs to crawl.

## 2.3. Setup a project

To set up a project, click on the button `New Project`:



Then, fill the form:



A screenshot of a 'Create or edit a Project' form. The form has a title bar with a close button (x) and a tabbed interface with the following tabs: 'General' (selected), 'Browsers', 'Reports and Scheduling', 'Advanced', 'Runtime', and 'API'. The 'General' tab contains the following fields:

- Name\***: A text input field containing 'project-e2e-tests' with a green checkmark on the right.
- Description**: A text input field containing 'Small project description'.
- Labels**: A text input field containing 'Associate labels to a project to organize projects.' with a '+ Add' button on the right.
- Repository**: A section header.
- Type\***: A dropdown menu with 'git Git' selected and 'Subversion' as an alternative.
- URL\***: A text input field containing 'https://github.com/viltgroup/minium-mail-e2e-tests.git'.
- Username**: A text input field containing 'bob'.
- Password**: A password input field with masked characters and a visibility toggle icon.
- Branch**: A text input field containing 'master'.

At the bottom right of the form, there are two buttons: 'Save' and 'Cancel'.

On the `General` tab, fill the following fields:

<b>Name</b>	Display name of the project (e.g gmail-e2e-tests)
<b>Description</b>	Optionally provide a project description
<b>Labels</b>	Associated labels to a project in order to organize and filter projects.
<b>Type</b>	Type of SCM repository
<b>Url</b>	URL of the repository where the project with the features is stored. The following are examples of valid git URL's (or a local file path):  <a href="https://github.com/github/git.git">https://github.com/github/git.git</a>  //dolos/repos/gmail-e2e-tests
<b>Username/Password</b>	Credentials for authentication on the repository.
<b>Branch</b>	The name of the branch you want to execute (GIT only). The default value is <code>master</code>

### Cookie

For the Cookie Crawler project the General tab is different:

Create or edit a Project ✕

---

General

Reports and Scheduling

Advanced

Runtime

API

**Name\***

**Description**

**Labels**  + Add ▾

**Cookie Crawler Settings**

**Urls crawler\***

**Domains to Filter\***

**Max Depth\***  ⌵

**Accept Cookies**  **Active**

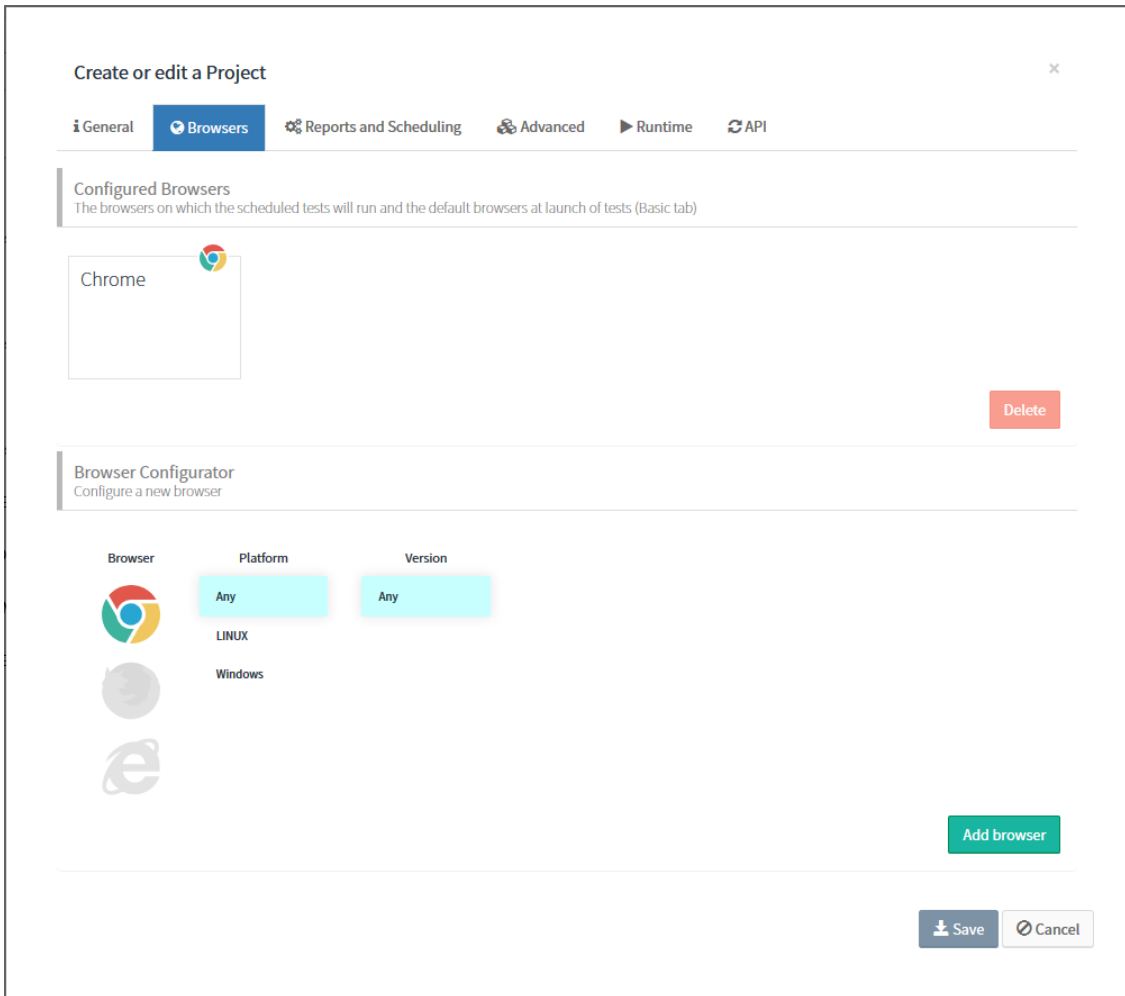
**Cookies Validations**

**Configurations** Set the cookie name and/or value and/or URL of the cookies you want to validate.

+

Save
Cancel

<b>Urls crawler</b>	The URLs you want to crawl (e.g. <a href="https://www.caixabank.es/index_es.html">https://www.caixabank.es/index_es.html</a> )
<b>Domains to Filter</b>	The domains you want to filter at the crawl (e.g. <a href="http://www.caixabank.es">www.caixabank.es</a> )
<b>Max Depth</b>	Maximum depth for the crawler to follow links automatically
<b>Accept Cookies</b>	Accept the cookie warning during the crawl (is related to the field <code>Script to accept cookies</code> at the Runtime tab of the Cookie Crawler project).
<b>Cookies Validations</b>	the cookie name and/or value and/or URL of the cookies you want to validate. After you configure one validation, click at the button + to add the validation.



On the `Browsers` tab, choose the browsers on which the scheduled tests will run, and the default browsers at launch of tests (Basic tab)

To configure the browsers, click `Browser Configurator` to expand the available browsers and select a browser (and the properties). After, click `Add browser`.

If you want to remove a configured browser, select the browser configuration (below the configured browsers) and click "Delete".



**Note:**

The `Browsers` tab is not available for the monitoring and cookie projects.

Project Configurations
✕

Simulate visitor interaction with your site automatically and get alerted when your critical site flows stop working correctly

---

General
Browsers
Reports and Scheduling
Advanced
▶ Runtime
↻ API

---

**Emails**

**Recipients**

Send only when there are test failures

**Report formats**

PDF

XLSX

PDF with screenshots

**Scheduling**

**Scheduler**

Never run

Every Day

Every Week

Every Month

Every Day at midnight

Custom

**Scheduler Custom Configurations**

At 00:00

Every:  at  :

Delete Project

Update

Cancel

On the `Reports and Scheduling` tab, fill the following fields:

<b>Recipients</b>	Set email addresses to receive a report of execution for the project. If you only want to receive the report of the executions that contains failed tests, check the option "Send only when there are test failures".
<b>Report formats</b>	Set the format(s) that the report of execution will be sent to the email. If you want to receive the pdf report with screenshots, check the option "PDF with screenshots".
<b>Scheduler</b>	Set the schedule you want to execute the project and send the report of the execution. If you choose the option <code>Custom</code> you can set a custom schedule.
<b>Scheduler Custom Configurations</b>	Set the schedule you want to execute the project and send the report of the execution with different levels of granularity.

Project Configurations
✕

Simulate visitor interaction with your site automatically and get alerted when your critical site flows stop working correctly

---

General
Browsers
Reports and Scheduling
Advanced
Runtime
API

**Permissions**

**Store execution videos**

**Max number of test executions passed stored**

**Max number of test executions failed stored**

**Module**

Name	Read	Write	Launch tests
Minium Team	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

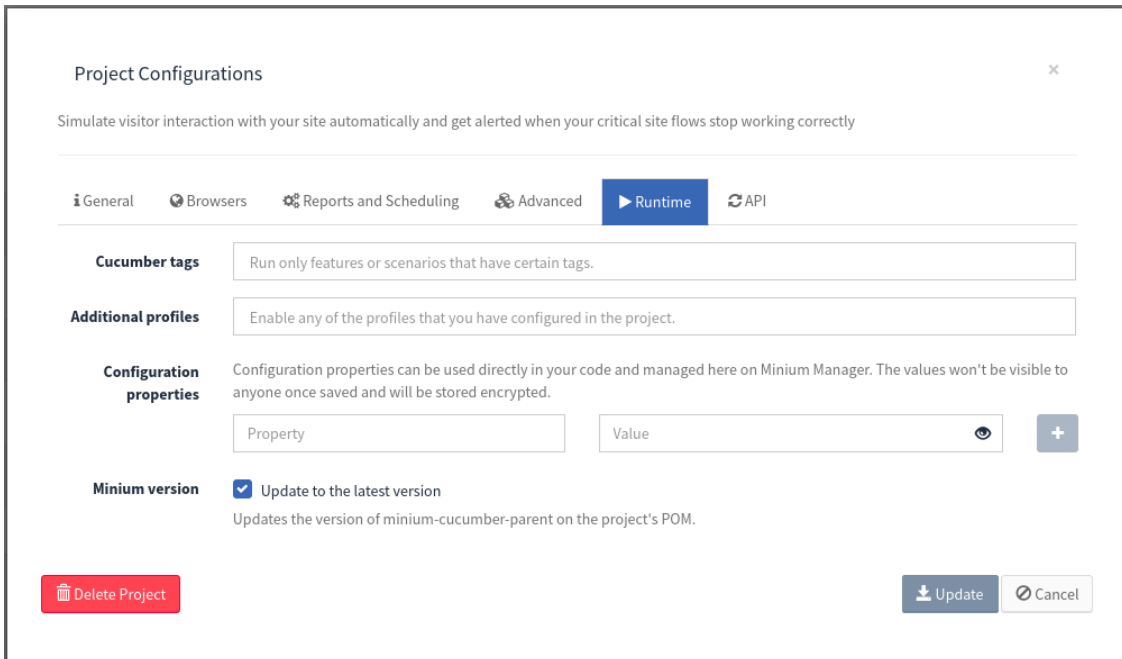
+ Append

Inactive

Delete Project
Update
Cancel

On the `Advanced` tab, fill the following fields:

<b>Permissions</b>	Add and personalize the level of access that assigns to each user or group.
<b>Store executions videos</b>	Configuration to allow the storage of videos of the executions.
<b>Max number of test executions passed stored</b>	Set the max number of executions passed to be stored at Minium Manager.
<b>Max number of test executions failed stored</b>	Set the max number of executions failed to be stored at Minium Manager.
<b>URL Loading time threshold</b>	Maximum time to load a page (in seconds) to warning the user at the executions page. <b>Available only to Monitoring projects</b>
<b>Module</b>	The path to the folder where the features are in the repository (e.g minium-developer-e2e-tests).



At the `Runtime` tab, fill the following fields:

<b>Cucumber tags</b>	Use this option to tell Minium Manager that only run features or scenarios that have certain tags.
<b>Additional Profiles</b>	Enable any of the profiles that are configured in the <code>./config/application.yml</code> project file. See <a href="#">Configuration profiles</a> for more details.
<b>Configuration properties</b>	This configuration properties can be used directly in your code and managed here on Minium Manager. The values won't be visible to anyone once saved and will be stored encrypted.

## Cookie

For the Cookie Crawler project the Runtime tab is different:

Create or edit a Project ×

---

General
Reports and Scheduling
Advanced
▶ Runtime
API

---

**Inject Cookie** Set the cookies you want to inject at the cookie crawler.

+

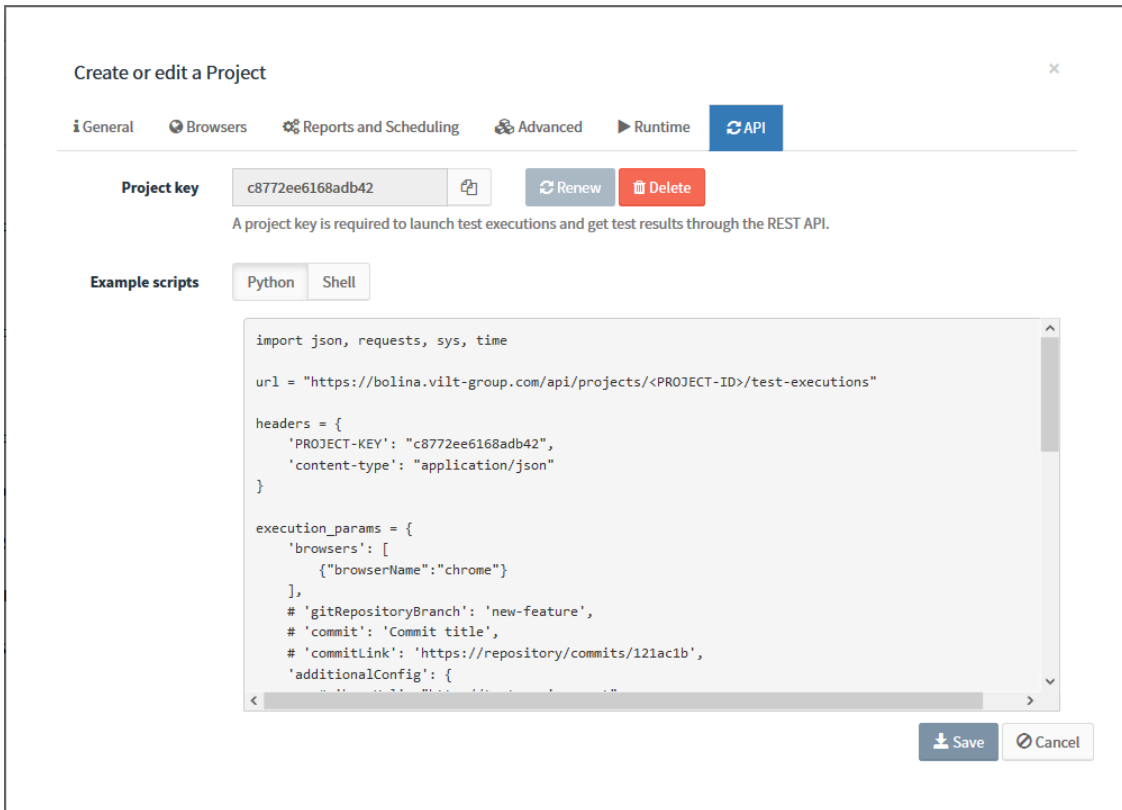
**Script to accept cookies** Add your script to accept the cookies.

```

await page.waitForSelector("#cookies-accept-full > a", { visible: true, timeout: 10000 });
await page.click("#cookies-accept-full > a");
await page.waitFor(2000);
        
```

Save
Cancel

<b>Inject Cookie</b>	The the cookies you want to inject at the crawl (e.g. cookie to accept the cookie warning)
<b>Script to accept cookies</b>	The script to accept the cookie warning (in <a href="#">puppeteer</a> ). This field is related to the field <a href="#">Accept Cookies</a> at the General tab of the Cookie Crawler project. The default script accepts the cookie warning for <a href="https://www.caixabank.es">https://www.caixabank.es</a> .



On the `API` tab, fill the following fields:

<b>Project key</b>	Generate a project key (required to launch test executions and get test results through the REST API).
<b>Example scripts</b>	Once the project has an API key, the scripts will be fulfilled with the project-specific data and ready to use. The first part of the script shows how to launch a test execution.

## 2.4. Project permissions

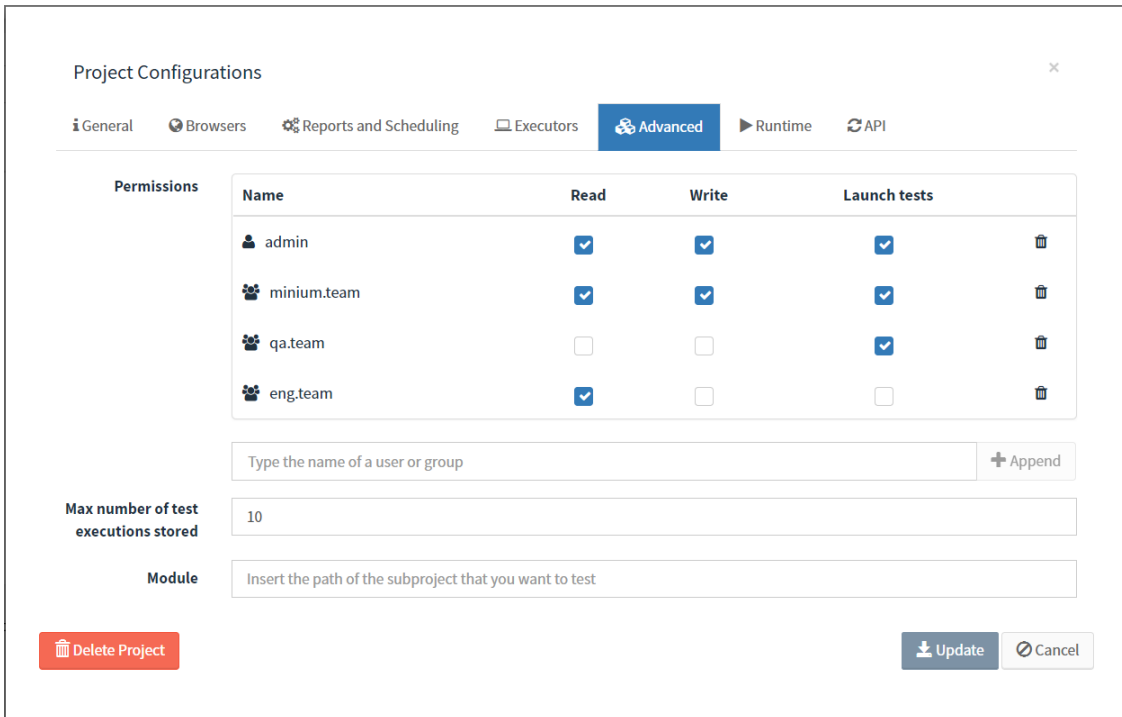
Manage the access levels of the users/groups in the projects. It is also possible to add and personalize the level of access that assigns to each user or group, per-project.



**Note:**

Write permissions is required to update the permissions of a project.

In order to change the project permission the first step is go to the project configuration and click on tab `Permissions`.



The screenshot shows the 'Project Configurations' dialog box with the 'Advanced' tab selected. The 'Permissions' section contains a table with columns for Name, Read, Write, and Launch tests. Below the table is an input field for adding users or groups, a 'Max number of test executions stored' field, and a 'Module' field. At the bottom, there are buttons for 'Delete Project', 'Update', and 'Cancel'.


Name	Read	Write	Launch tests
admin	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
minium.team	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
qa.team	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
eng.team	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Add user or groups to the permission table. Below the permission table, start typing the name of the user or group desired. Then click on the button `Append`.



**Note:**

It is possible to select more than one user or group.



The screenshot shows the input field from the previous image, now containing two user names: 'bob' and 'alice', each with a small 'x' icon to its right. A '+ Append' button is visible to the right of the input field.

After the users or groups was appended to the permissions table, define the permission for each entry appended and click `Update` to save the project configurations.

Project Configurations ✕

General Browsers Reports and Scheduling Executors Advanced Runtime API

---

**Permissions**

Name	Read	Write	Launch tests	
admin	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
minium.team	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
qa.team	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
eng.team	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
bob	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
alice	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Type the name of a user or group + Append

**Max number of test executions stored**

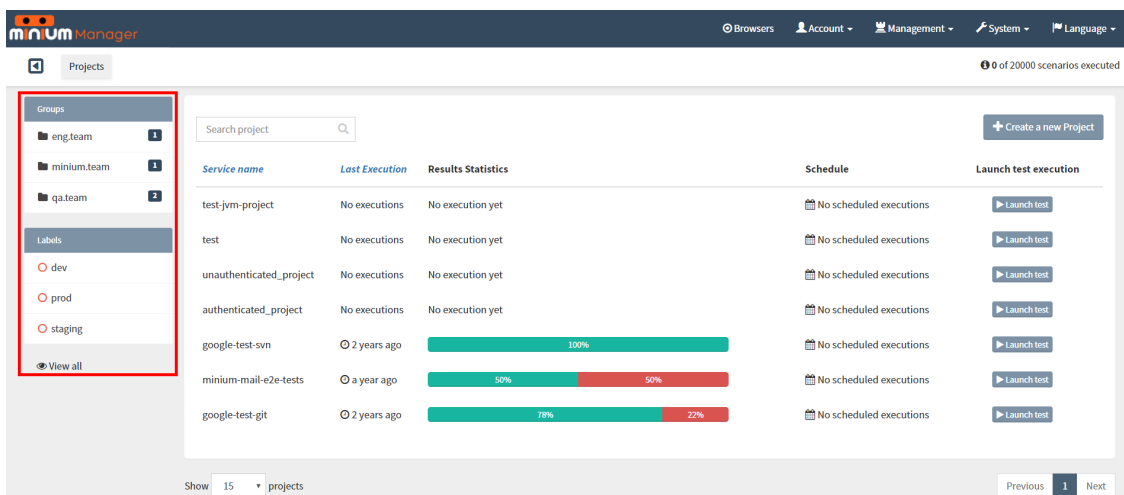
**Module**

Delete Project

Update
Cancel

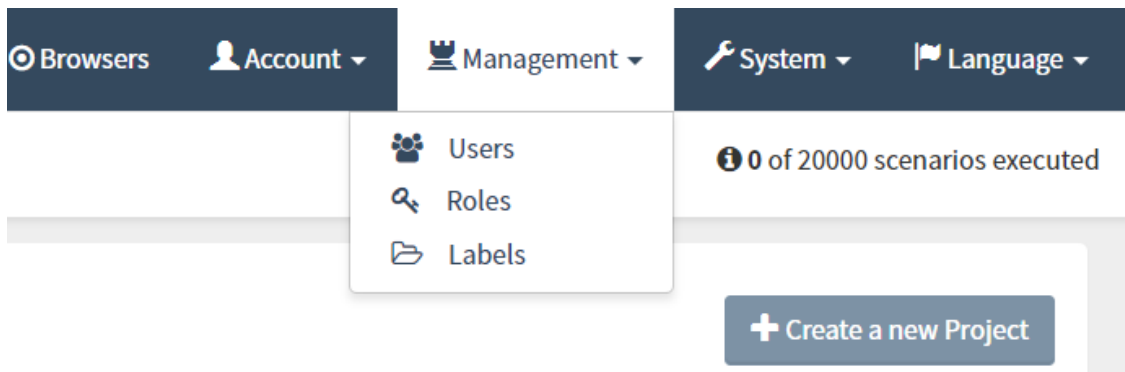
## 2.5. Organise projects by Labels or Groups

**Labels** were meant to be used only to filter projects in the Projects page providing a useful way to organize the projects.

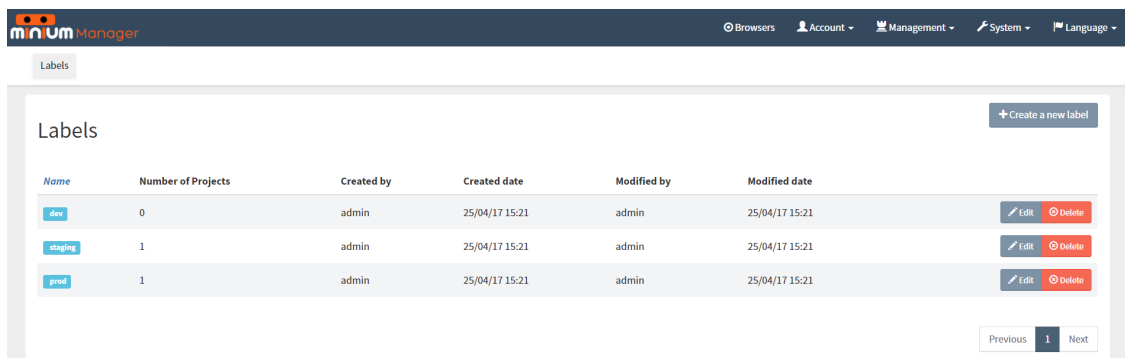


### 2.5.1. Manage Labels

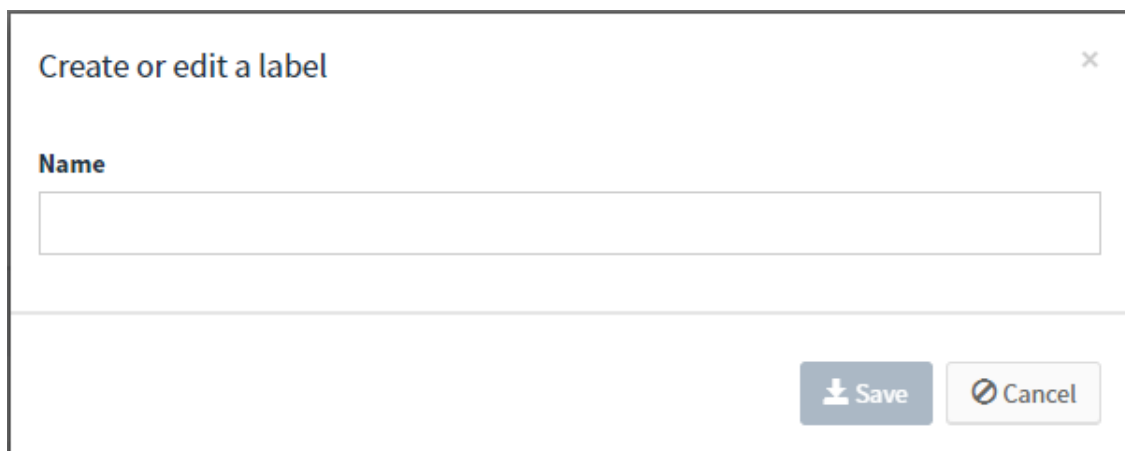
To manage the labels, open the `Management` menu and click on `Labels`:



The Labels page is shown with all current labels. Edit, delete and create new ones if desired.

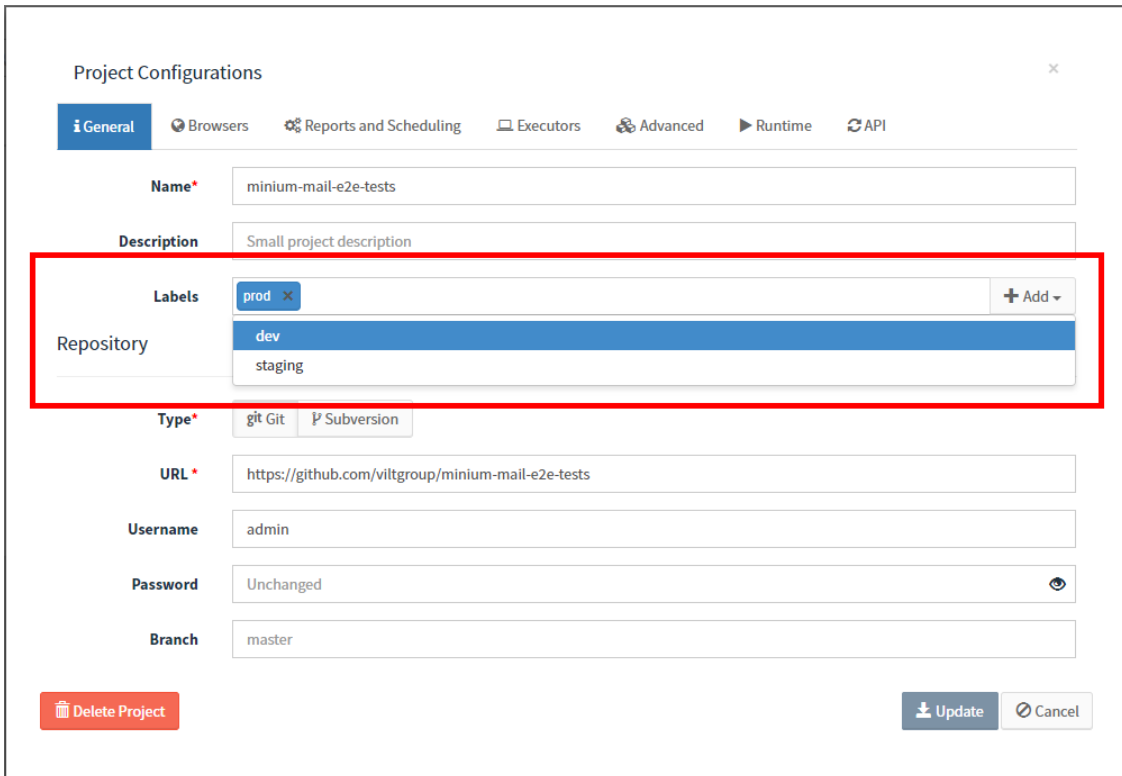


To create a new label, click on `Create a new label`. To delete or edit an existing label, click on `Edit` or `Delete` respectively.



## 2.5.2. Assign labels to the projects

To assign some labels to a project, go to the project configurations:

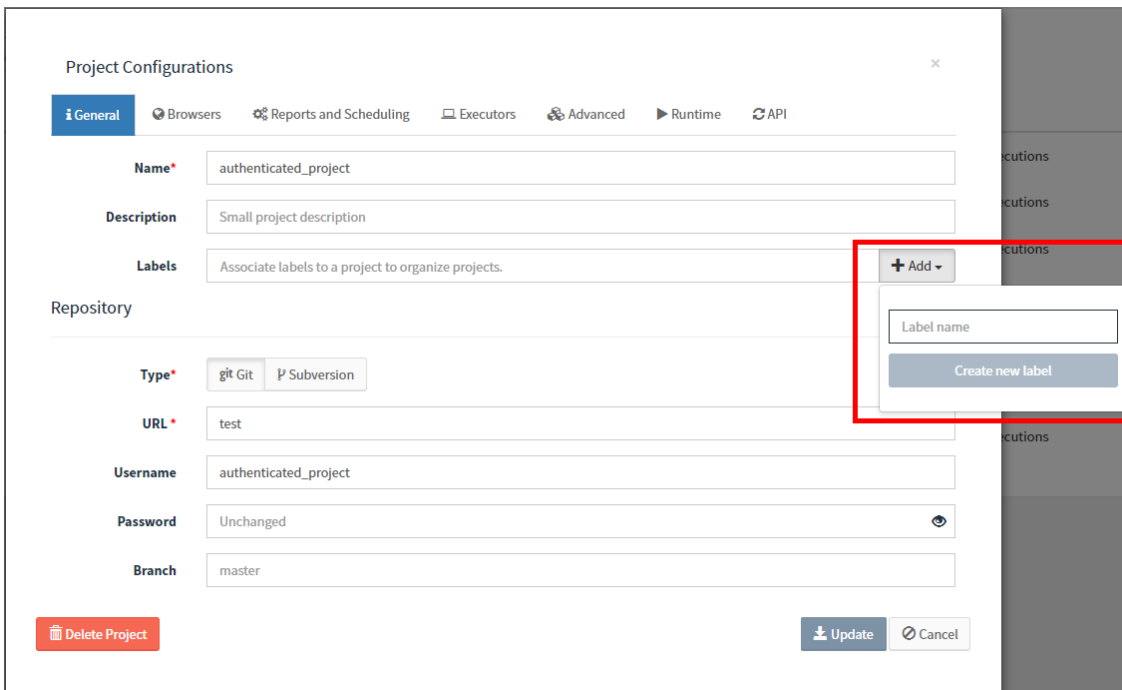


The screenshot shows the 'Project Configurations' dialog box with the following fields and options:

- Name\***: minium-mail-e2e-tests
- Description**: Small project description
- Labels**: A dropdown menu with 'prod' selected and an '+ Add' button.
- Repository**: A dropdown menu with 'dev' selected and 'staging' as an option.
- Type\***: Radio buttons for 'git Git' and 'Subversion'.
- URL\***: https://github.com/viltgroup/minium-mail-e2e-tests
- Username**: admin
- Password**: Unchanged
- Branch**: master

Buttons at the bottom include 'Delete Project', 'Update', and 'Cancel'.

The Add button can be used to create new labels if needed.



The screenshot shows the 'Project Configurations' dialog box with the 'Add' button in the Labels section highlighted. A modal dialog box is open, showing a 'Label name' input field and a 'Create new label' button.

The background dialog box shows the following fields and options:

- Name\***: authenticated\_project
- Description**: Small project description
- Labels**: Associate labels to a project to organize projects. + Add
- Repository**
- Type\***: Radio buttons for 'git Git' and 'Subversion'.
- URL\***: test
- Username**: authenticated\_project
- Password**: Unchanged
- Branch**: master

Buttons at the bottom include 'Delete Project', 'Update', and 'Cancel'.

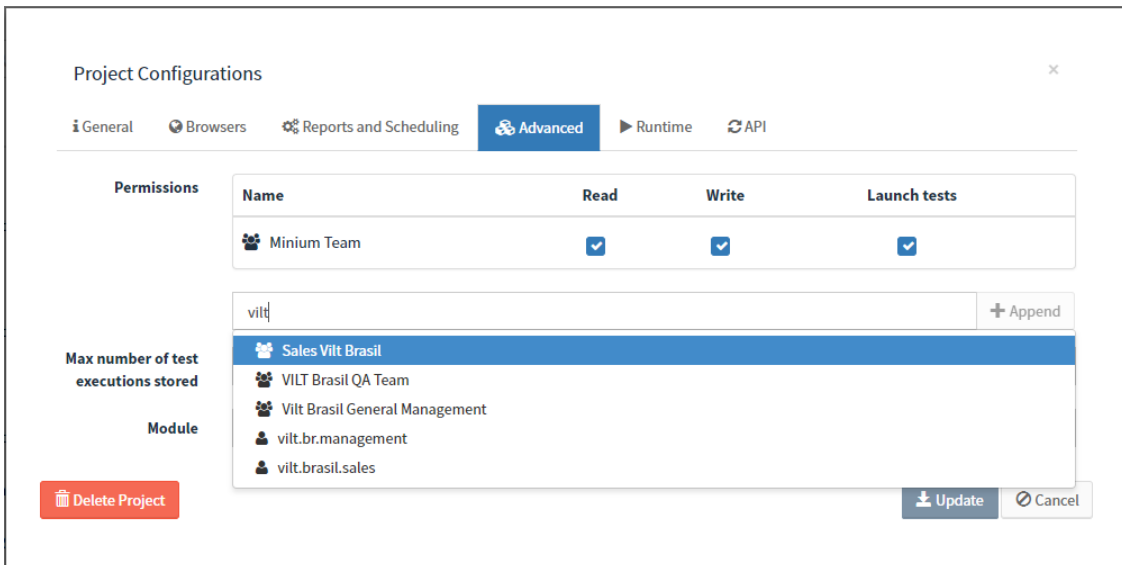
### 2.5.3. Manage groups

**Groups** are intended to organize projects at a higher level and to perform changes in several projects at a time.

## Manage groups using LDAP authentication

If Minium Manager is using LDAP to manage the users, the groups are configured through the LDAP.

In the project configuration, search for groups. Below the permission table, start typing the name of the user or group. This will perform a search into the LDAP directory.



The screenshot shows the 'Project Configurations' dialog box with the 'Advanced' tab selected. It features a permissions table and a search input field. The search results list several LDAP groups.

Name	Read	Write	Launch tests
Minium Team	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Search input: vilt

- Sales Vilt Brasil
- VILT Brasil QA Team
- Vilt Brasil General Management
- vilt.br.management
- vilt.brasil.sales

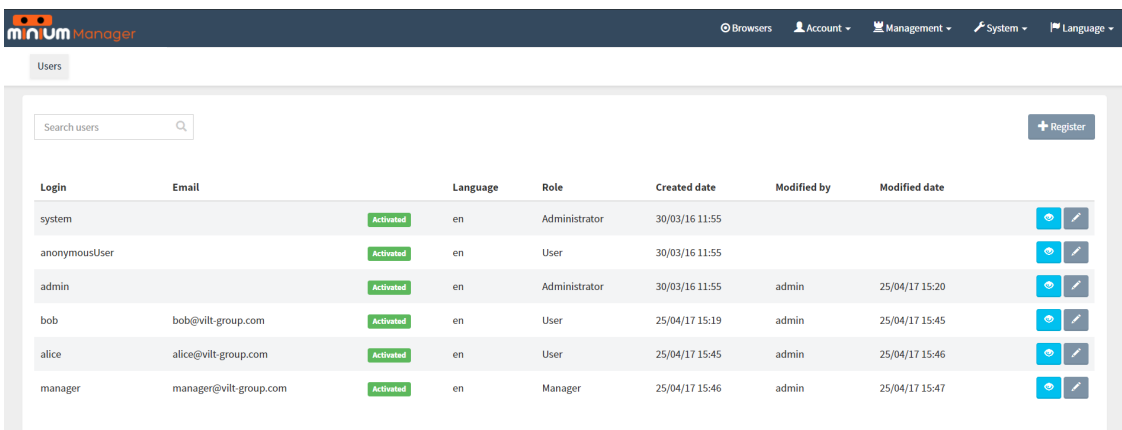
## Manage groups without LDAP authentication

If Minium Manager is **not** using LDAP authentication, create new groups through the user management UI.



### Note:

Only the users with `Admin` privileges are allowed to create groups.



The screenshot shows the 'Users' management page. It includes a search bar, a '+ Register' button, and a table of users with columns for Login, Email, Language, Role, Created date, Modified by, and Modified date.

Login	Email	Language	Role	Created date	Modified by	Modified date
system		Activated	en	Administrator	30/03/16 11:55	
anonymousUser		Activated	en	User	30/03/16 11:55	
admin		Activated	en	Administrator	30/03/16 11:55	admin 25/04/17 15:20
bob	bob@vilt-group.com	Activated	en	User	25/04/17 15:19	admin 25/04/17 15:45
alice	alice@vilt-group.com	Activated	en	User	25/04/17 15:45	admin 25/04/17 15:46
manager	manager@vilt-group.com	Activated	en	Manager	25/04/17 15:46	admin 25/04/17 15:47

Create or edit a user
✕

**Login**

**First name**

**Last name**

**Email**

**Activated**

**Language**

**Groups**

managers.team ✕
minium.team ✕
eng.team ✕
qa.team ✕
+ Add ▾

Cancel

✓

Create

**Login**

**E-mail**

**Groups**

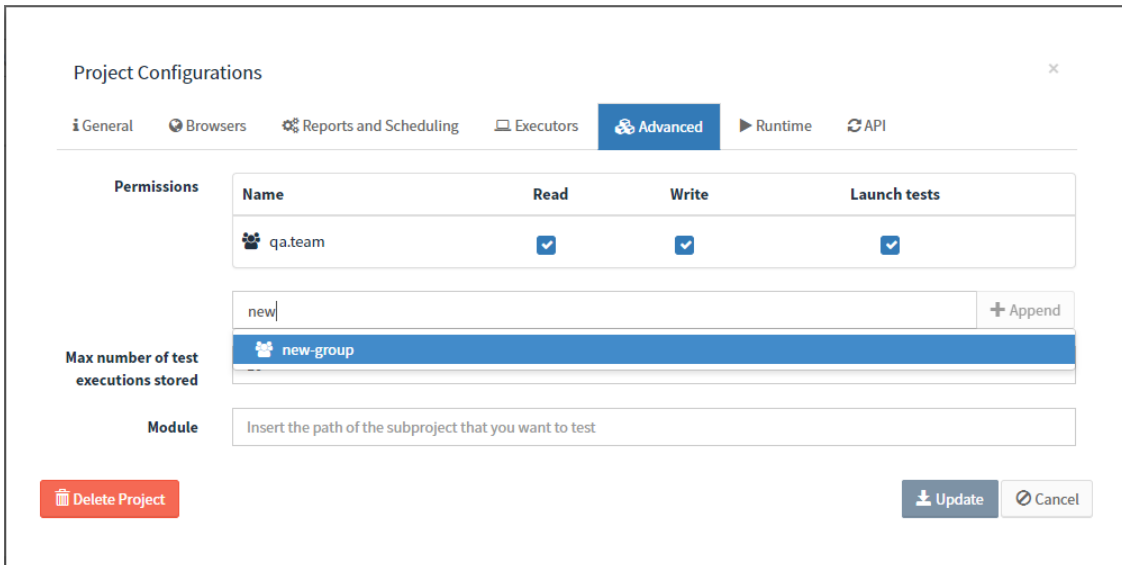
qa.team ✕
minium.team ✕
eng.team ✕
+ Add ▾

← Back
Register

✓

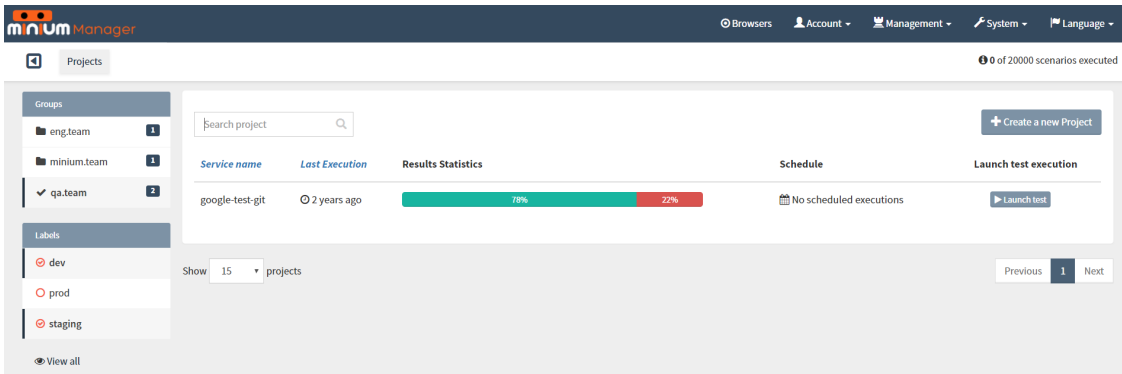
Create

If desired add permissions to the group recently created in the project configuration. Below the permission table, start typing the name of the group created.



## 2.5.4. Organise Projects

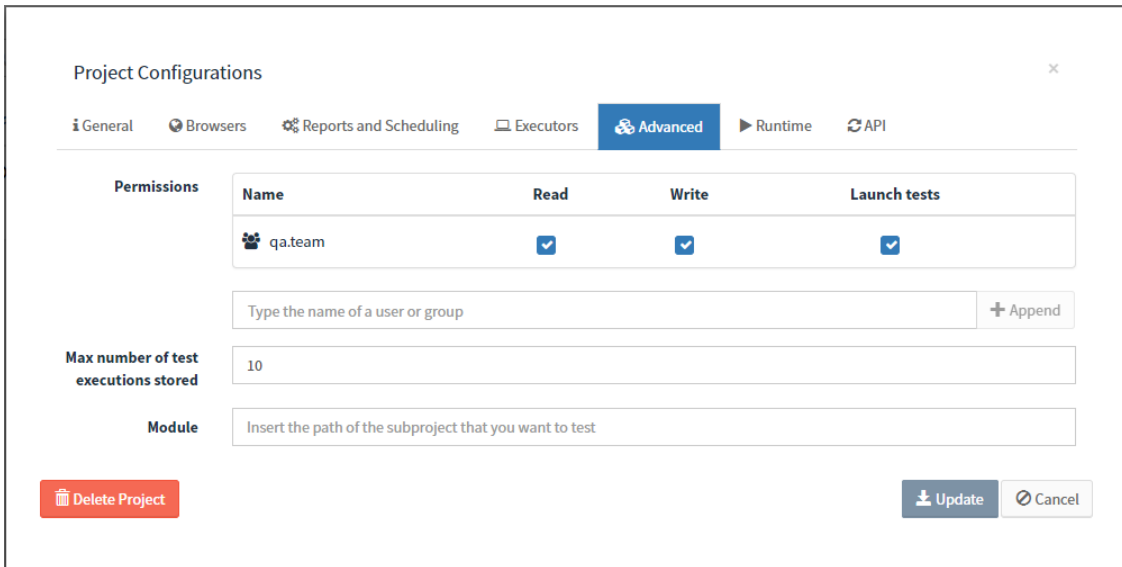
Using **groups** and **labels** to organise the projects selecting a group and labels from the sidebar.



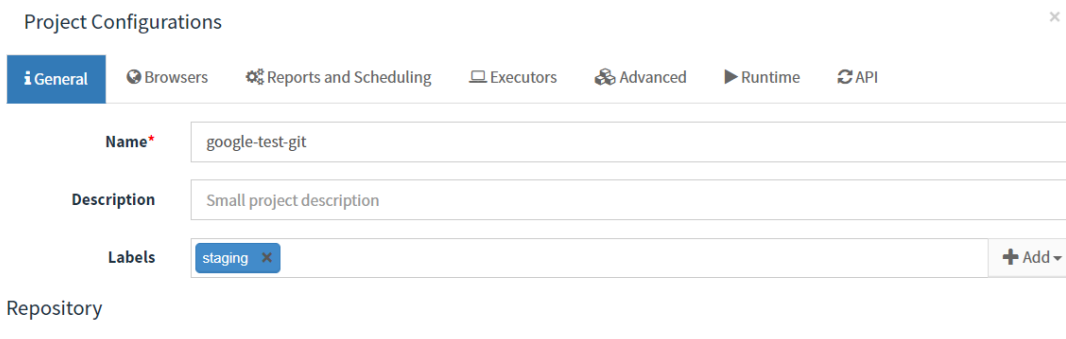
Select one **group** at a time, but select multiple **labels** and combine both.

In the case of the figure above, notice that the **group qa.team** is selected and two **labels** are also selected (**dev** and **staging**). With filters applied, exists one project shown in the project list.

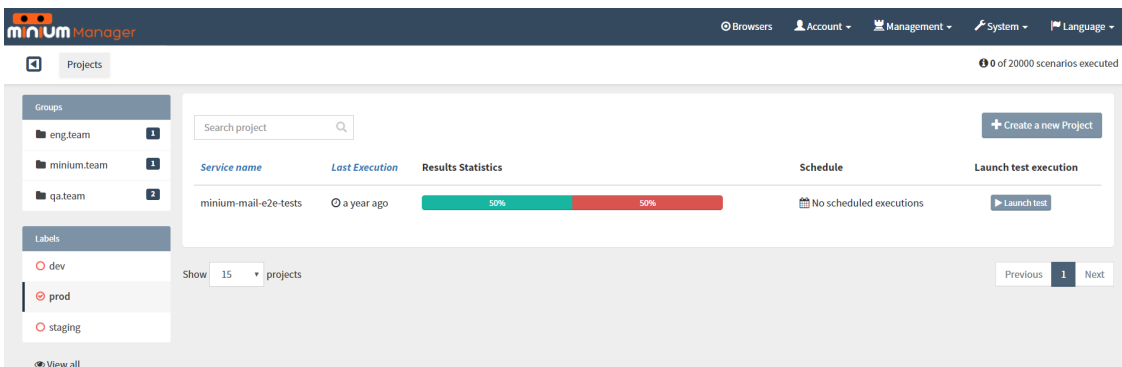
This means that the one project belong to the **group qa.team** (see it in the figure below).



The project either has the **label dev** or **staging** associated with it in the project configuration (as shown in the figure below).

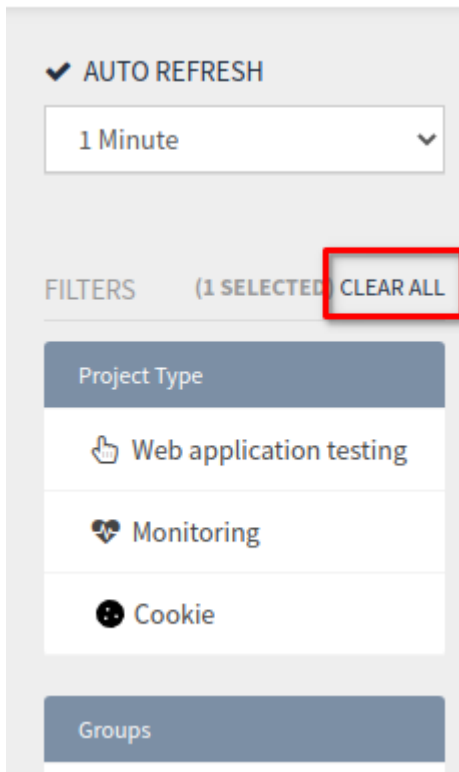


Select a **group** or **one or more label**, like shown in the figure below. In this specific case, are selected only the **label prod**, without selecting any group. It means that all the projects in the list have the **label prod** associated.



## Clean project filters

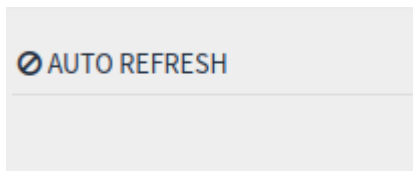
By clicking on the link **CLEAR ALL**, it will remove all the selected filter.



## Auto Refresh

By default, the project page will be refreshed every minute. Also, you can configure the project page to refresh every 5 and 10 minutes.

To disable the auto refresh, just click on the link "AUTO REFRESH":



### Note:



The link "AUTO REFRESH" with the ban icon, means that is disabled; The link "AUTO REFRESH" with the check icon, means that is enabled.

## 2.6. Configuration profiles

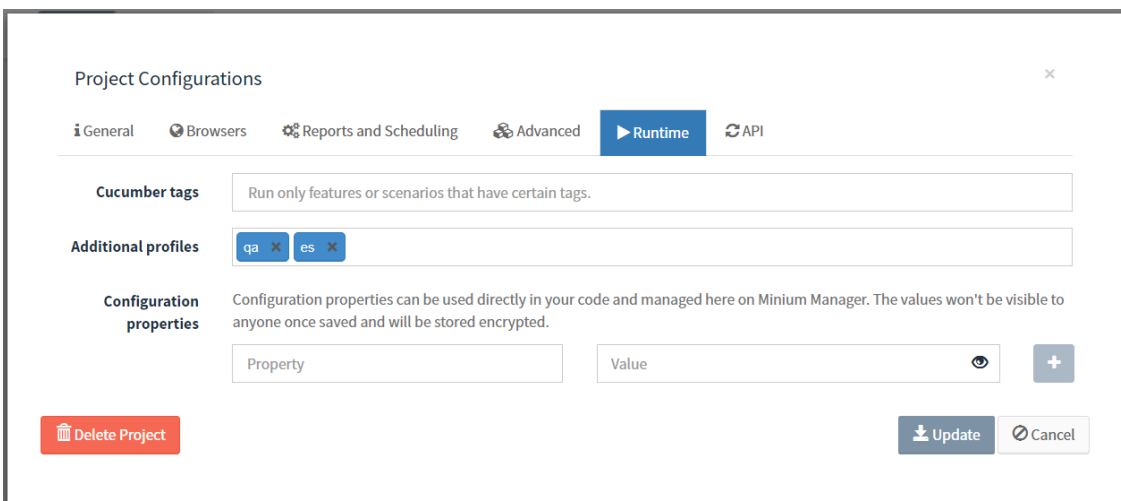
Configuration profiles can be defined on the `config/application.yml` of a project and used to load configuration properties that vary based on the environment being tested. Such a configuration could be for example the URL of the application or the language:

```
# default values
minium:
  config:
    baseUrl: http://localhost
    language: English
---
spring.profiles: qa
minium:
  config:
    baseUrl: http://staging
---
spring.profiles: es
minium:
  config:
    language: Spanish
```

These configuration properties can then be used on the step definitions like this:

```
When(/^I go to the homepage"$/, function() {
  browser.get(config.baseUrl);
  $(".dropdown").withText("Language").click();
  $(".dropdown a").withText(config.language).click();
});
```

Then to run test executions with the configuration properties of a profile, go to **Project Configurations** and add it to **Additional profiles**:



## 2.7. Secret configuration properties

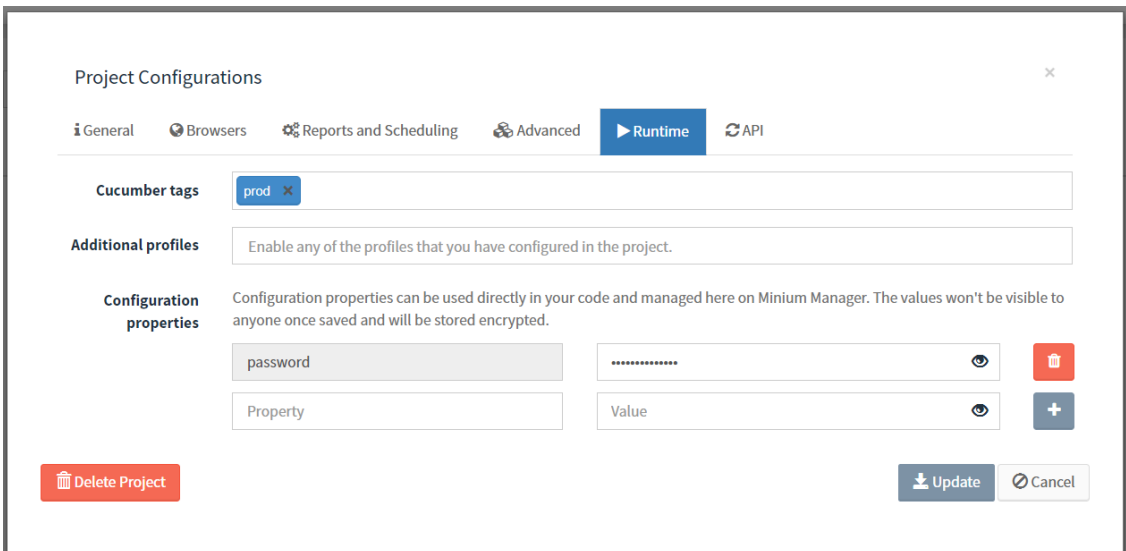
Sensitive configuration properties that cannot be included on the repository of the tests can be safely stored on Minium Manager. They can be used just like the configuration properties defined on the `config/application.yml` file of the project. As an example, consider that some tests need to be executed against a production environment which requires a password that cannot be exposed on the code. In that case, only the password to be used on the test environment would be defined on the `config/application.yml` file:

```
minium:  
  config:  
    baseUrl: http://staging  
    username: test  
    password: minium  
  
---  
spring.profiles: prod  
  
minium:  
  config:  
    baseUrl: https://production  
    username: admin
```

Which would be accessible through the `config.password` during the tests:

```
When(/^I login"$/, function() {  
  browser.get(config.baseUrl);  
  $("text").fill(config.username);  
  $("password").fill(config.password);  
  $("submit").click();  
});
```

The password to be used on the production environment would then be defined on the Project Configurations:




## 2.8. Delete a project

To delete a project, go to the project **configurations** and click on the button `Delete Project`, at the bottom-left corner:

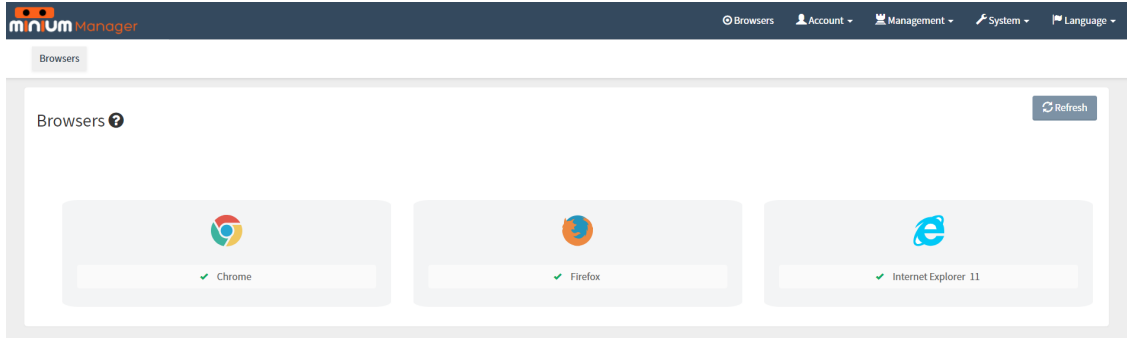
St

<b>Username</b>	<input type="text" value="admin"/>
<b>Password</b>	<input type="text" value="Unchanged"/>
<b>Branch</b>	<input type="text" value="master"/>

A red rectangular button with rounded corners, containing a white trash can icon and the text "Delete Project".

### 3. Check the available browsers

To check the available browsers, click on `Browsers`, at the navigation bar.

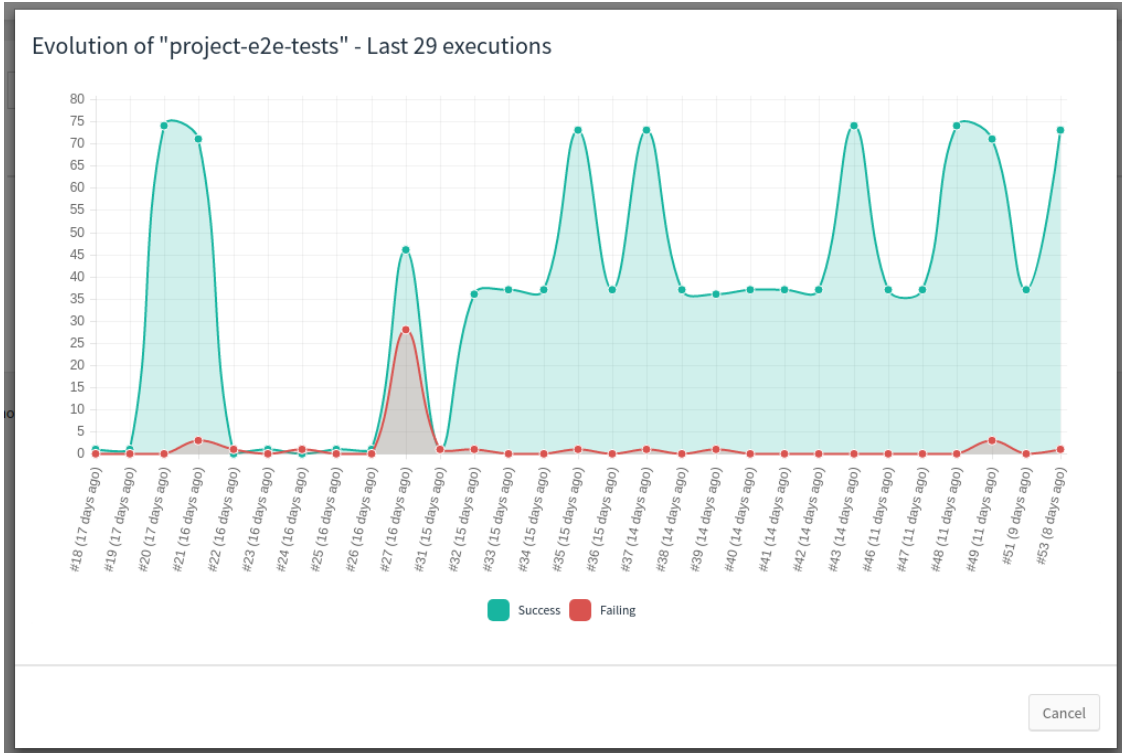


In the figure above, we only have one provider available, but we can have two or more providers configured (e.g. "BrowserStack" and/or "Selenium Grid Extras"). The provider configured has the browsers Chrome, Firefox and Internet Explorer available.

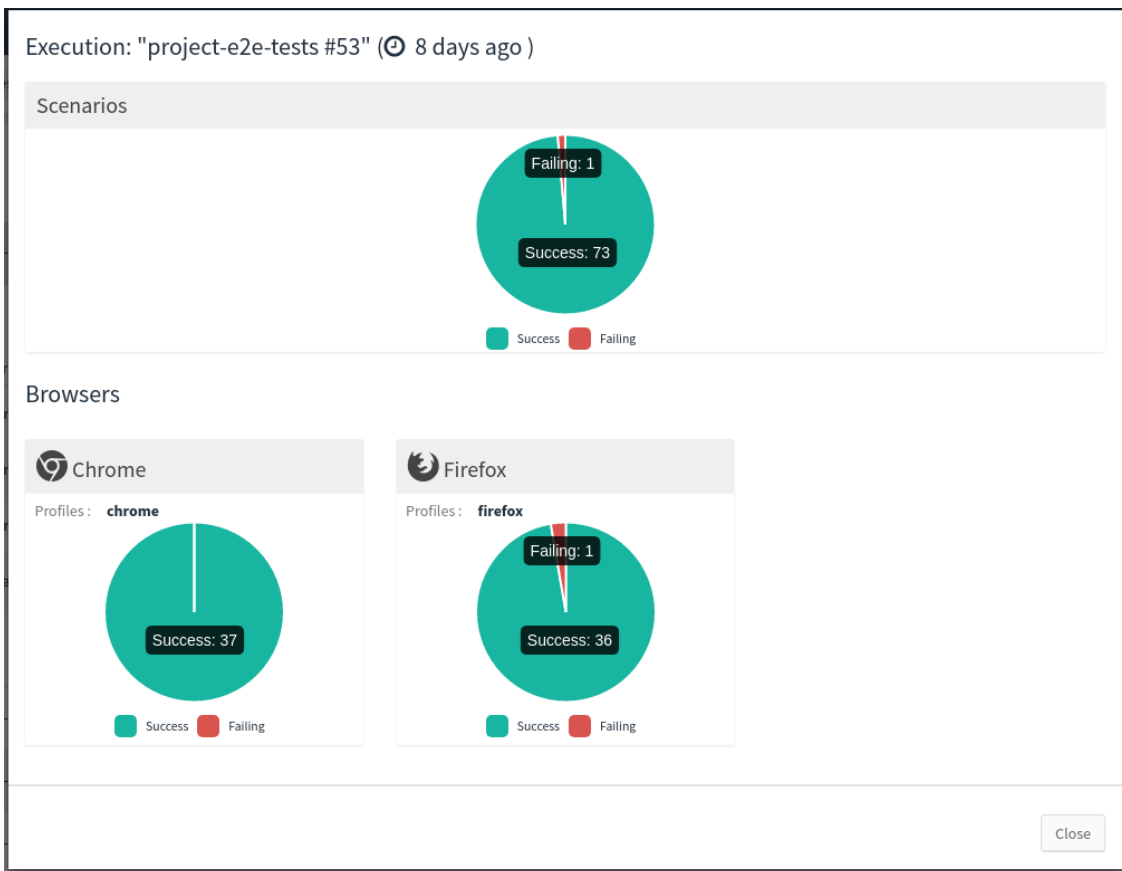
This means that with this environment is able to run the tests in Chrome, Firefox and Internet Explorer at the provider configured.

## 4. Project Statistics

Evolution chart to the test executions of a project.



Statistics for a single execution.



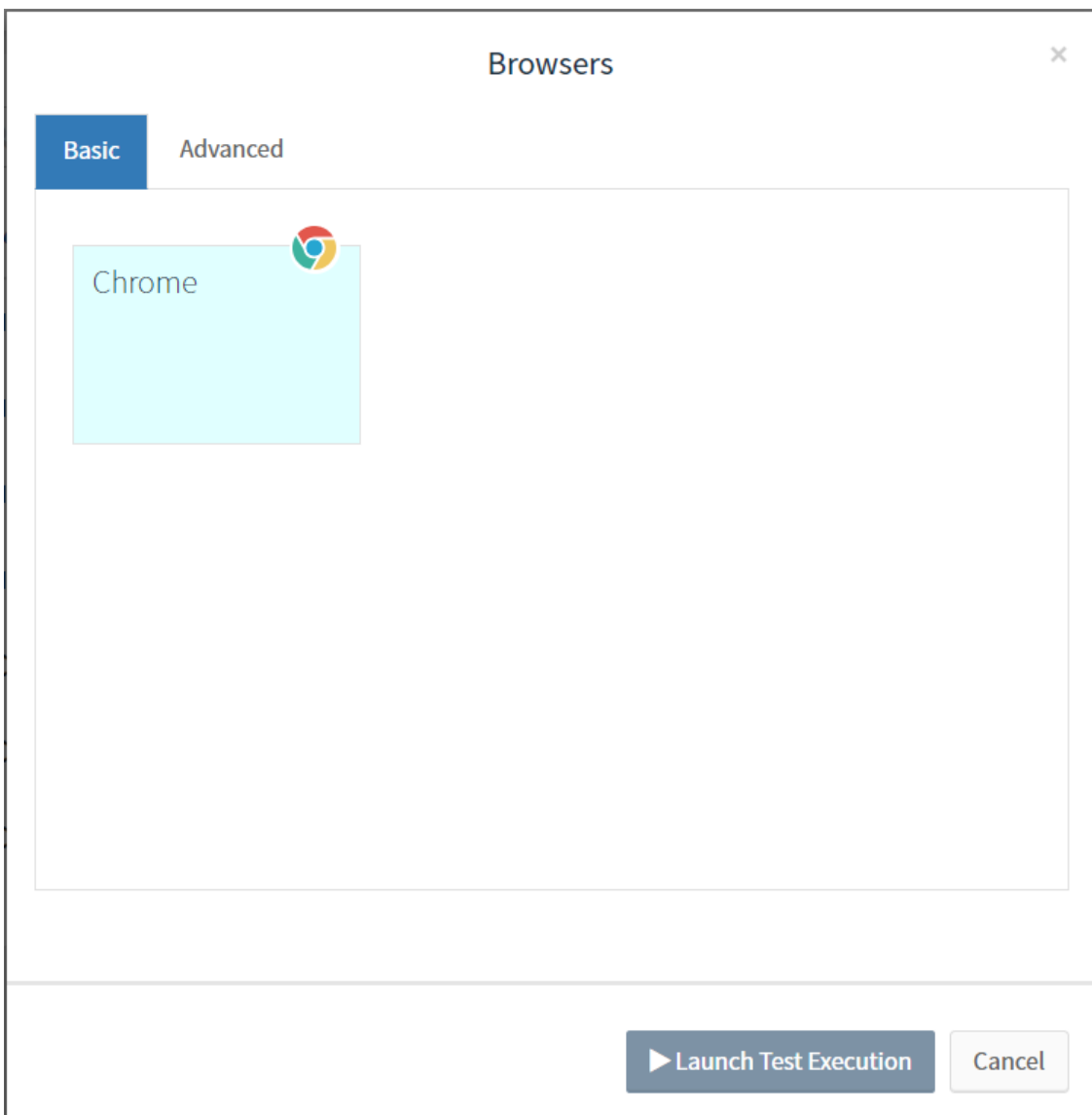
## 5. Test executions

### 5.1. Launch test executions

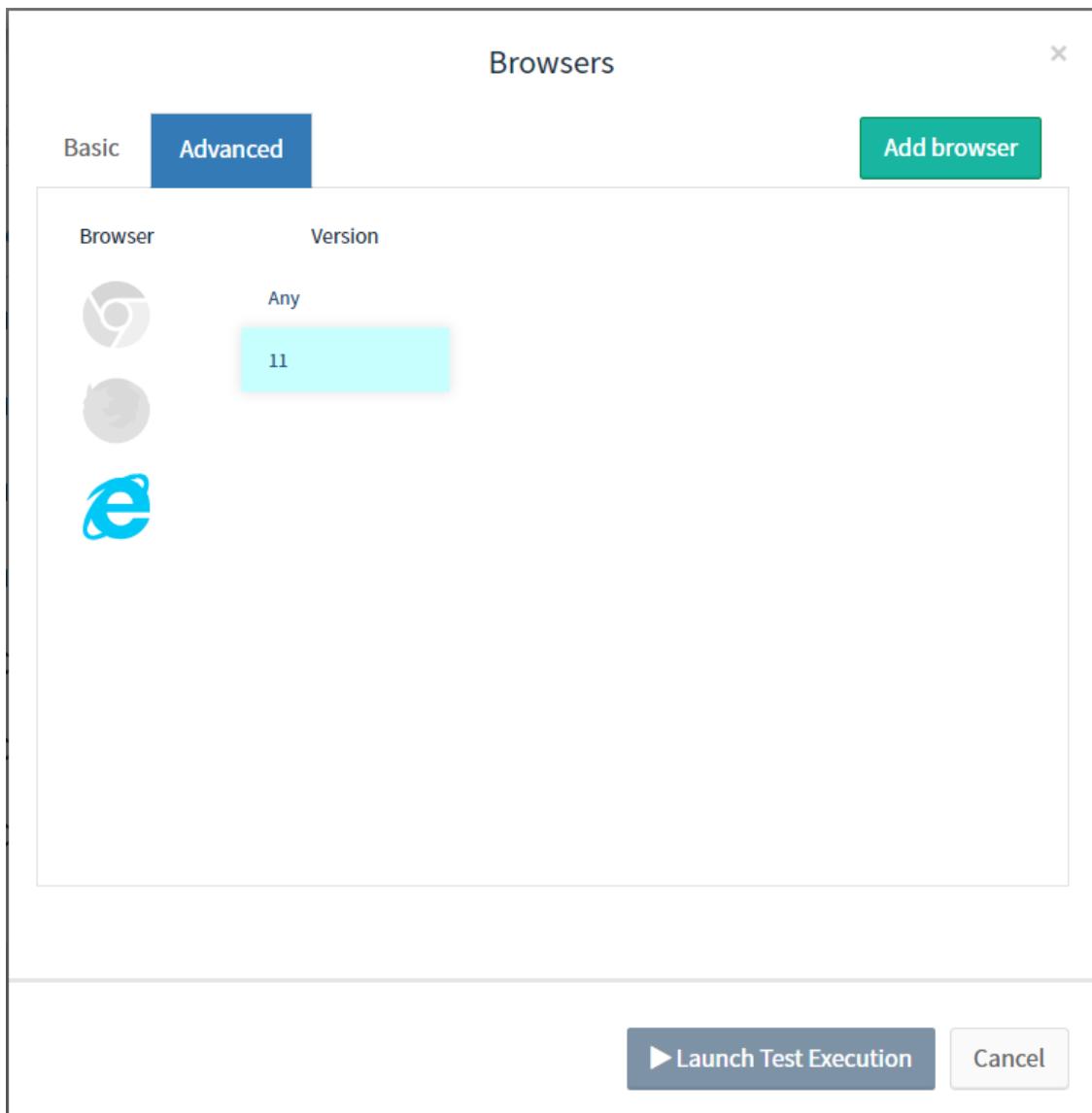
A test execution will execute the tests present in a project on one or more different browsers. There are alternative ways for launching test executions.

#### 5.1.1. Launch a test execution manually

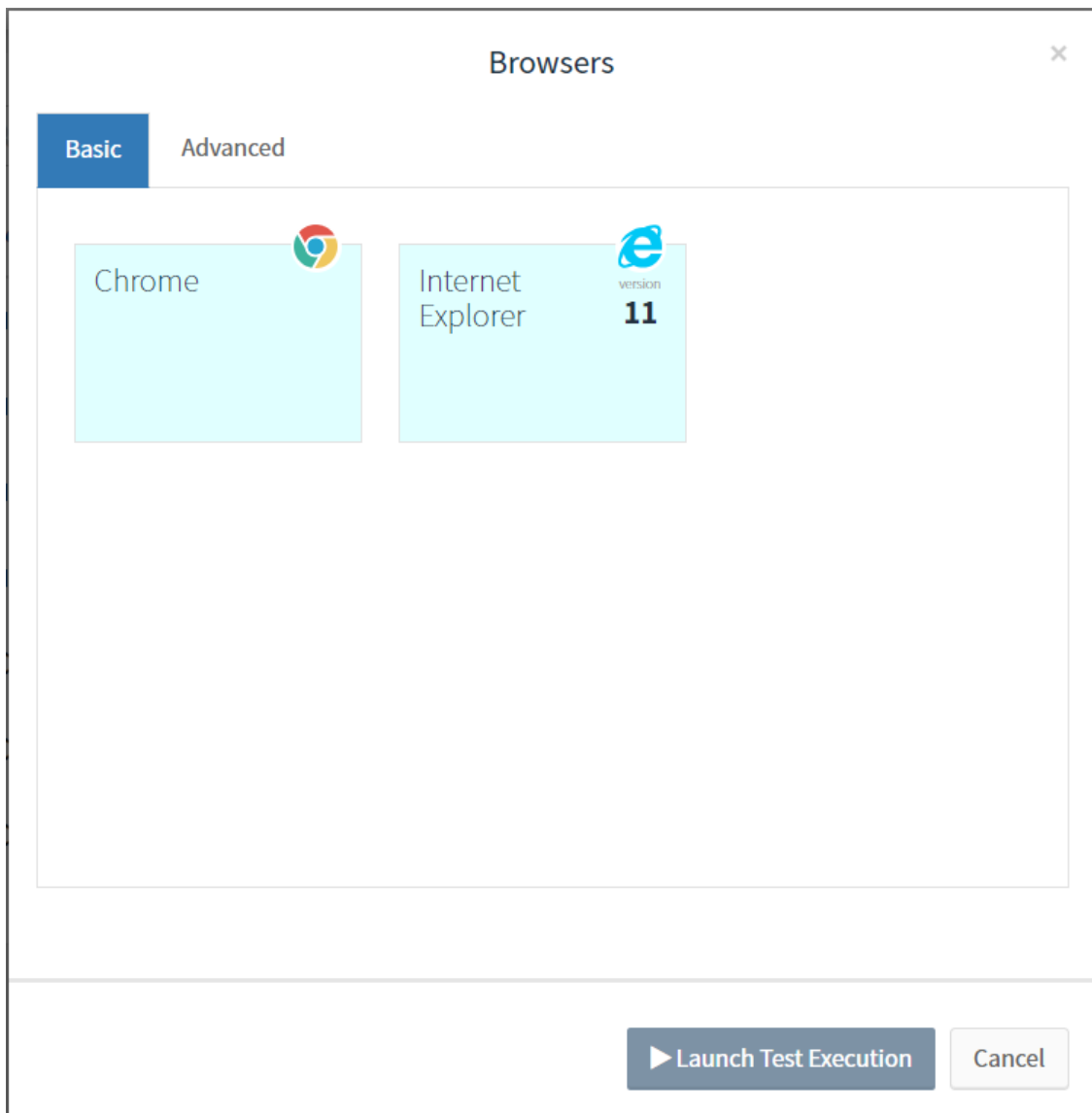
On the homepage or the page of a project, click on button `Launch test`, which opens a modal for selecting the browsers.



Select the desired browsers by clicking on them. If needed, click on the `Advanced` tab to configure a new one. To configure a browser, click on the corresponding icon and choose its properties. Then, click on `Add browser`.



The new browser is now available to be selected on the `Basic` tab.



After selecting all the browsers, click on `Launch Test Execution`.

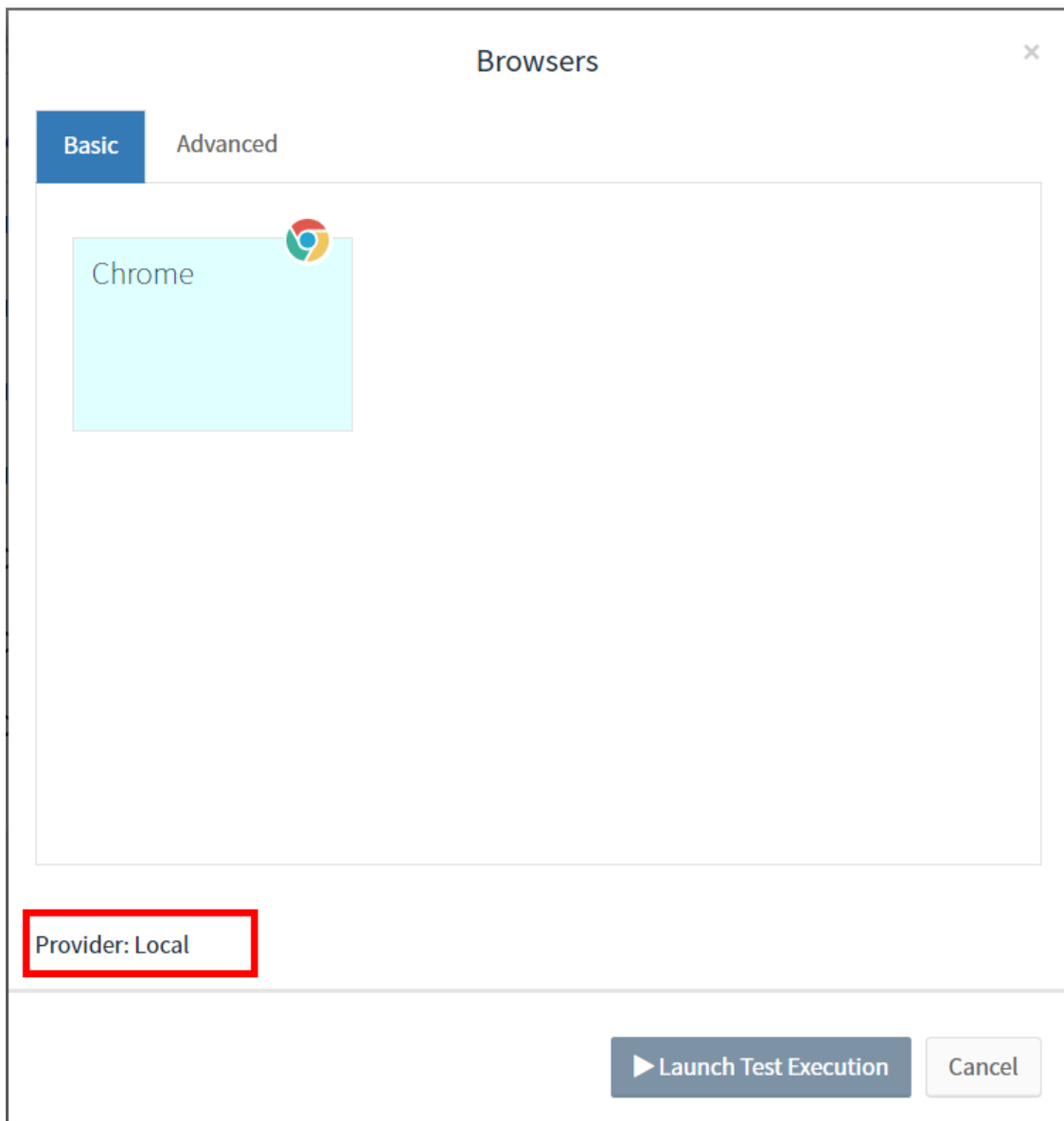


**Note:**

At the click on button `Launch test`, the Monitoring project don't open a modal for selecting the browsers. It launches a test immediately.

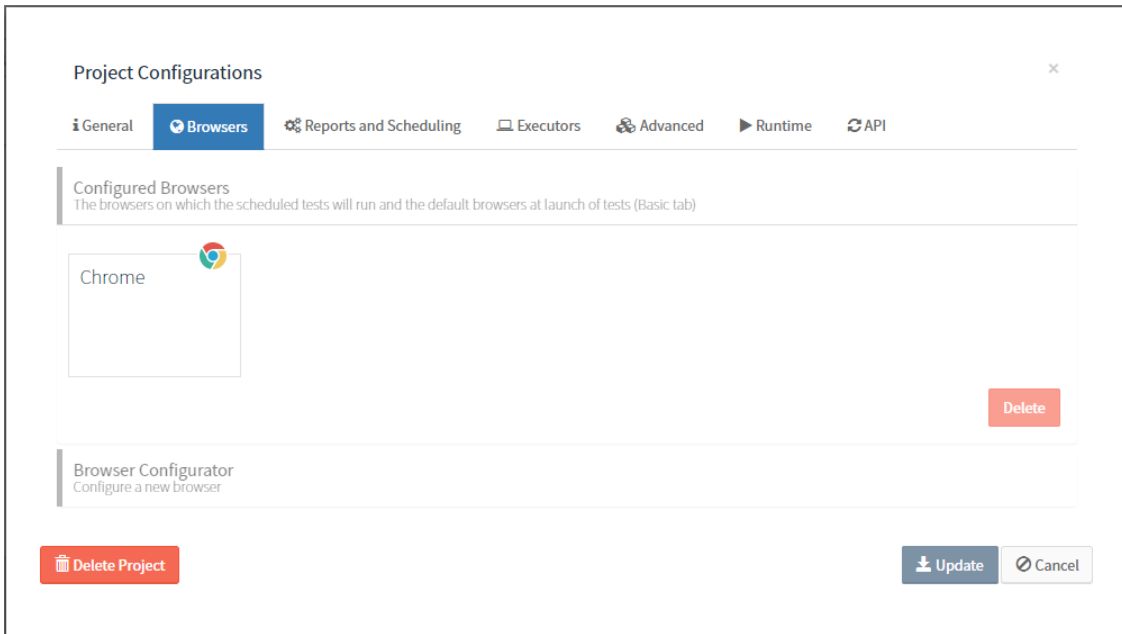
## Multiple Providers

With multiple providers, the Launch modal presents the provider configured for the project.

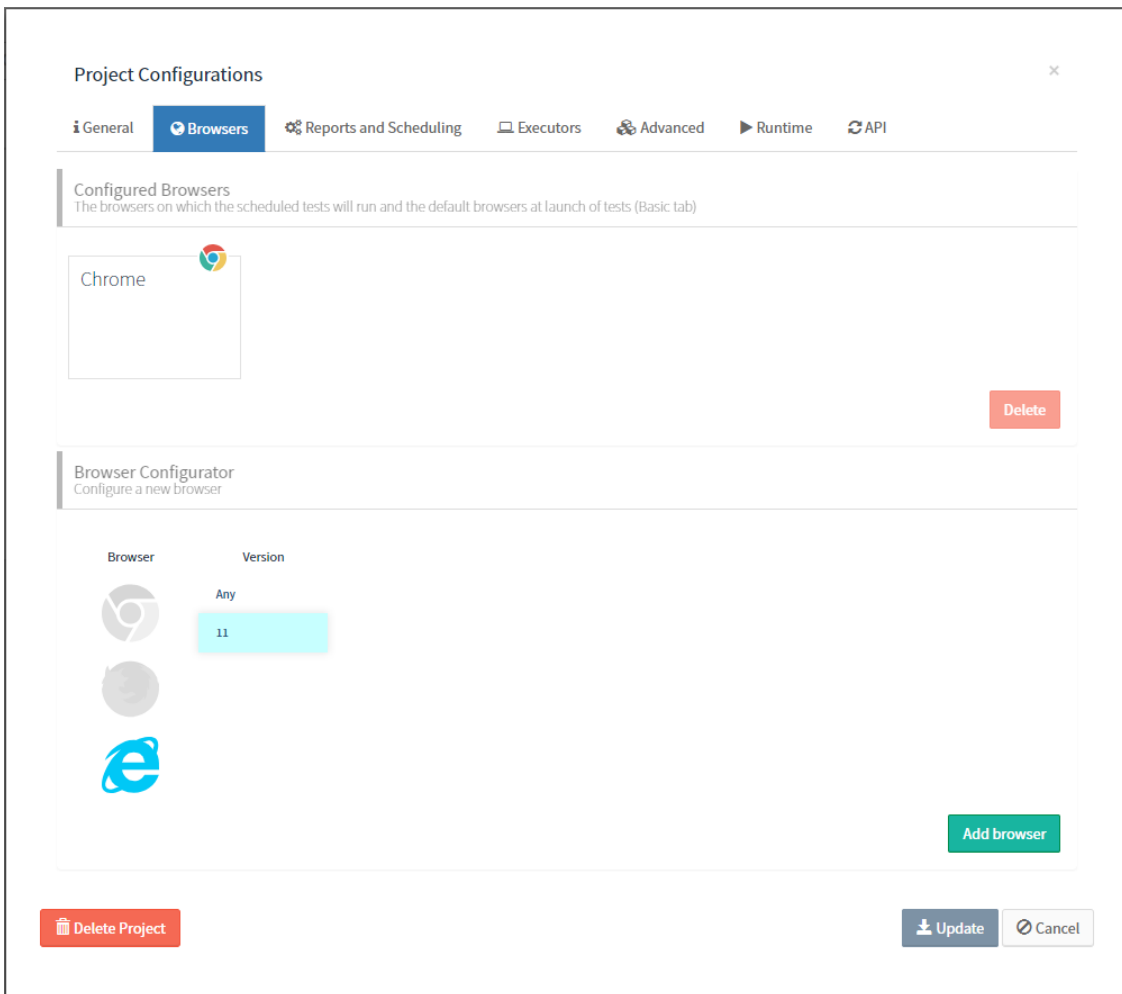


### 5.1.2. Schedule test executions

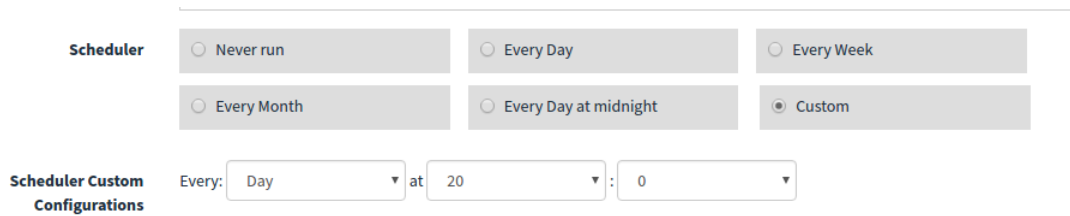
Go to the configurations of a project and open the `Browsers` tab to see the currently configured browsers.



Click on `Browser Configurator` and add more browsers if needed.



To schedule test executions on the configured browsers, switch to the `Reports` and `Scheduling` tab.



**Scheduler**

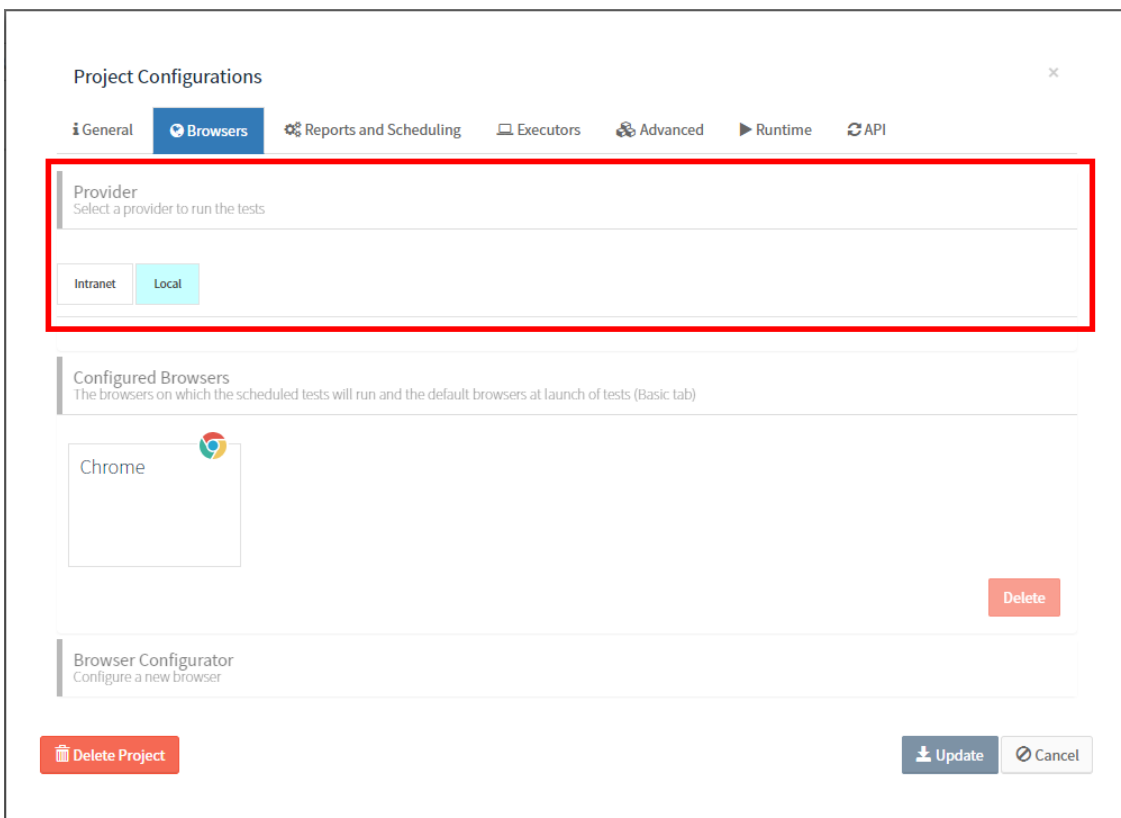
Never run  Every Day  Every Week

Every Month  Every Day at midnight  Custom

**Scheduler Custom Configurations** Every:  at  :

## Multiple Providers

With multiple providers, the `Browsers` tab has an extra field to configure the default provider.



Project Configurations

**Browsers** | Reports and Scheduling | Executors | Advanced | Runtime | API

**Provider**  
Select a provider to run the tests

Intranet  Local

**Configured Browsers**  
The browsers on which the scheduled tests will run and the default browsers at launch of tests (Basic tab)

Chrome

**Browser Configurator**  
Configure a new browser

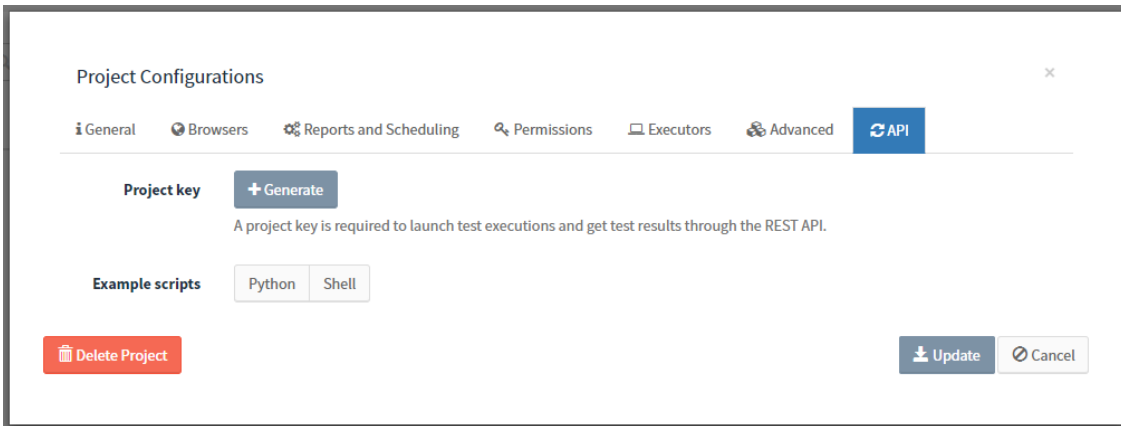


**Note:**

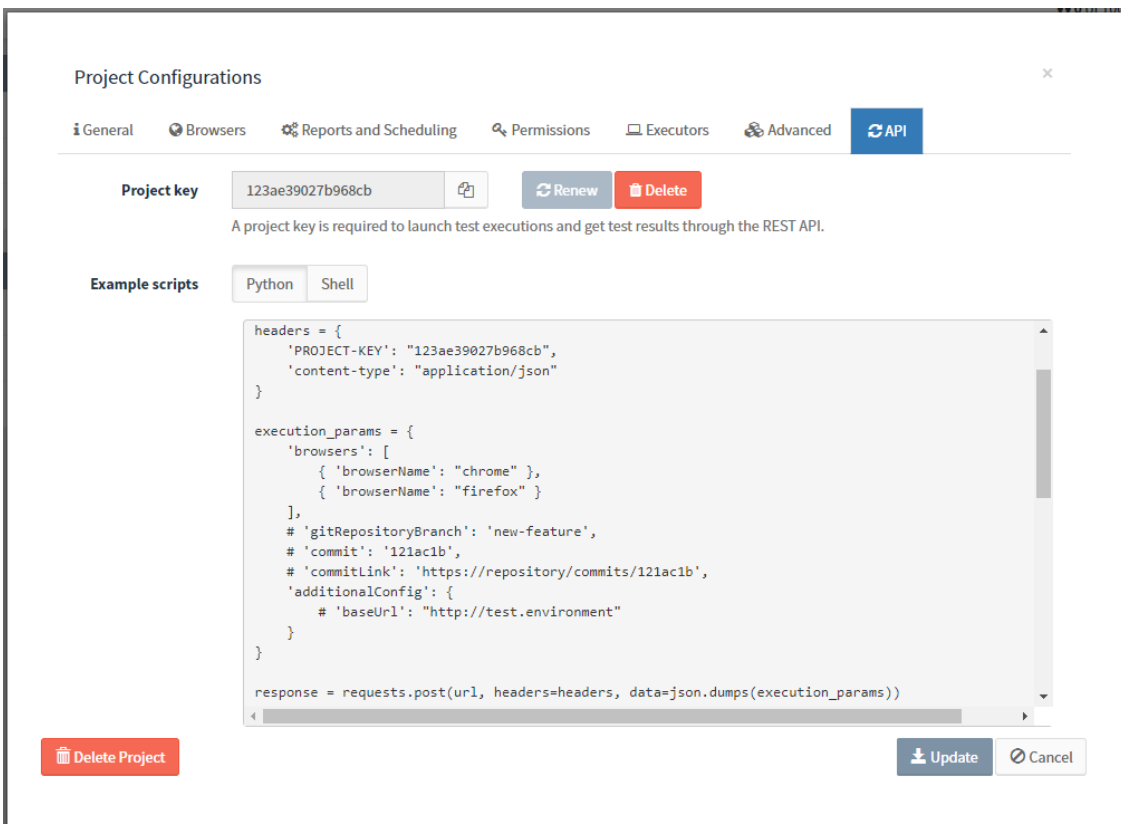
This configuration is not available at the Monitoring projects.

### 5.1.3. REST API

Minium Manager provides a REST API through which it is possible to launch test executions and get test results. In order to use it, an API key is required. API key's are specific of a project. To generate a key, go to the `API` tab on the configurations of a project and click on the `Generate` button.



Also, on the `API` tab are some example scripts. Once the project has an API key, the scripts will be fulfilled with the project-specific data and ready to use. The first part of the script shows how to launch a test execution.



A test execution can be launched by issuing a `POST` request to `api/projects/<PROJECT-ID>/test-executions` containing the `PROJECT-KEY` header set to the project key, and the `content-type` header set to `application/json`. The execution parameters are sent in JSON on the request body:

- **browsers**: list containing the capabilities of the browsers. Only the `browserName` is mandatory.
- **additionalConfig** (optional): configuration properties to be merged with the ones on the `application.yml` file of the project. Properties already defined on the

`application.yml` file are overridden.

- **gitRepositoryBranch** (optional): branch of the repository of the tests.
- **commit** (optional): the corresponding commit of the system under test, to be then included on the report of the execution.
- **commitLink** (optional): link to directly access the commit from Minium Manager.

To follow the progress of the execution until it finishes, poll the URL returned on the `Location` header of the response to the `POST` request.

```
url = response.headers['location']
response = requests.get(url, headers=headers)
while response.status_code == 404:
    time.sleep(5)
    response = requests.get(url, headers=headers)
if not response.ok: sys.exit(response.text)

execution = response.json()
while execution['state'] != "FINISHED":
    time.sleep(15)
    execution = requests.get(url, headers=headers).json()
```

While the tests are running, the response will be a JSON object containing the `state` field set to `RUNNING`, the name of the current browser and the corresponding progress in percentage, number of passed/failed and last finished feature/scenario. Example:

```
{
  "state": "RUNNING",
  "browser": "chrome",
  "progressInPercentage": 18,
  "failingScenariosCount": 1,
  "passingScenariosCount": 5,
  "feature": "Login",
  "scenario": "Successful login"
}
```

Once the execution finishes, the response will have `state FINISHED`, the global results and the number of passed/failed scenarios for each browser. Example:

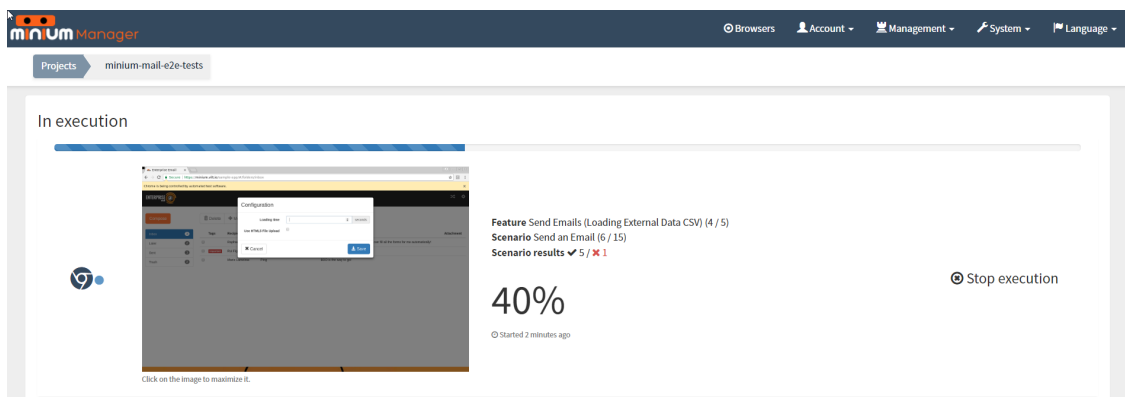
```

{
  "state": "FINISHED",
  "globalResults": {
    "totalScenarios": 20,
    "percentageOfPassingScenarios": 95,
    "passingScenarios": 19,
    "failingScenarios": 1
  },
  "browserResults": [
    {
      "browser": {
        "browserName": "chrome"
      },
      "passedScenarios": 10,
      "failedScenarios": 0
    },
    {
      "browser": {
        "browserName": "firefox"
      },
      "passedScenarios": 9,
      "failedScenarios": 1
    }
  ]
}

```

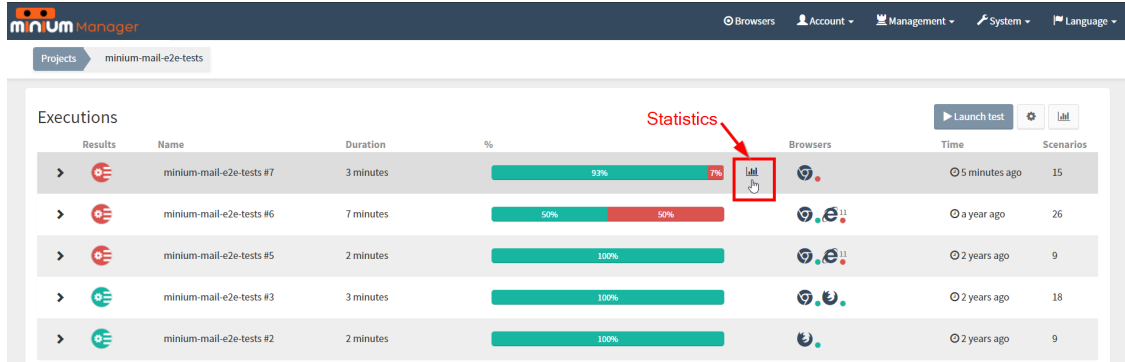
## 5.2. Follow the progress of a test execution

After launching a test execution, follow the progress in the project page. See the browser where the test is executing, the percentage, number of executed tests, the elapsed time and the test that is currently executing. Also, visualize the test be executed in the browser.



## 6. Monitor test results

In a project page, all the launched executions are showed:

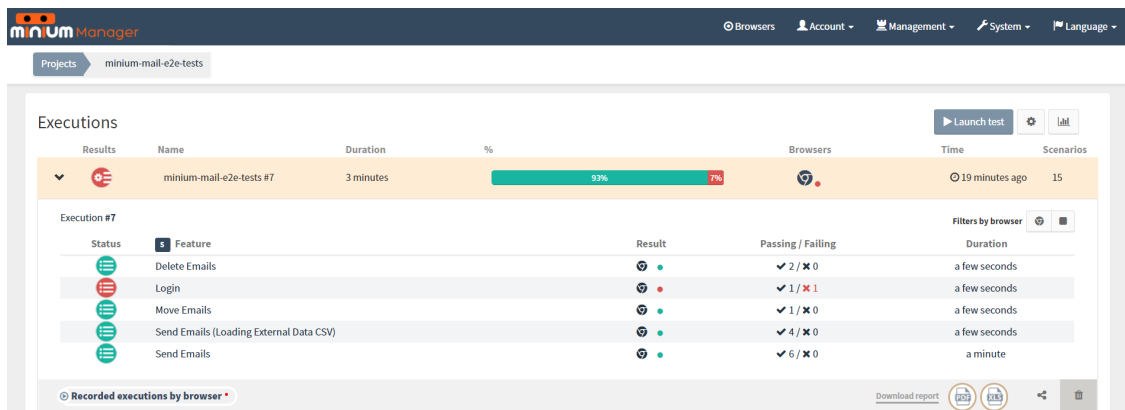


Results	Name	Duration	%	Browsers	Time	Scenarios
>	minium-mail-e2e-tests #7	3 minutes	93% 7%	🌐	🕒 5 minutes ago	15
>	minium-mail-e2e-tests #6	7 minutes	50% 50%	🌐 🌐	🕒 a year ago	26
>	minium-mail-e2e-tests #5	2 minutes	100%	🌐 🌐	🕒 2 years ago	9
>	minium-mail-e2e-tests #3	3 minutes	100%	🌐 🌐	🕒 2 years ago	18
>	minium-mail-e2e-tests #2	2 minutes	100%	🌐	🕒 2 years ago	9

For each execution it can be seen:

- result of the execution
- name
- duration of execution
- percentage of passed and failing scenarios
- statistics of the execution (shown when the mouse is over the execution)
- browser where the execution ran
- launch time
- number of scenarios of the execution

When expands a test execution a list of executed features, and their results can be seen:



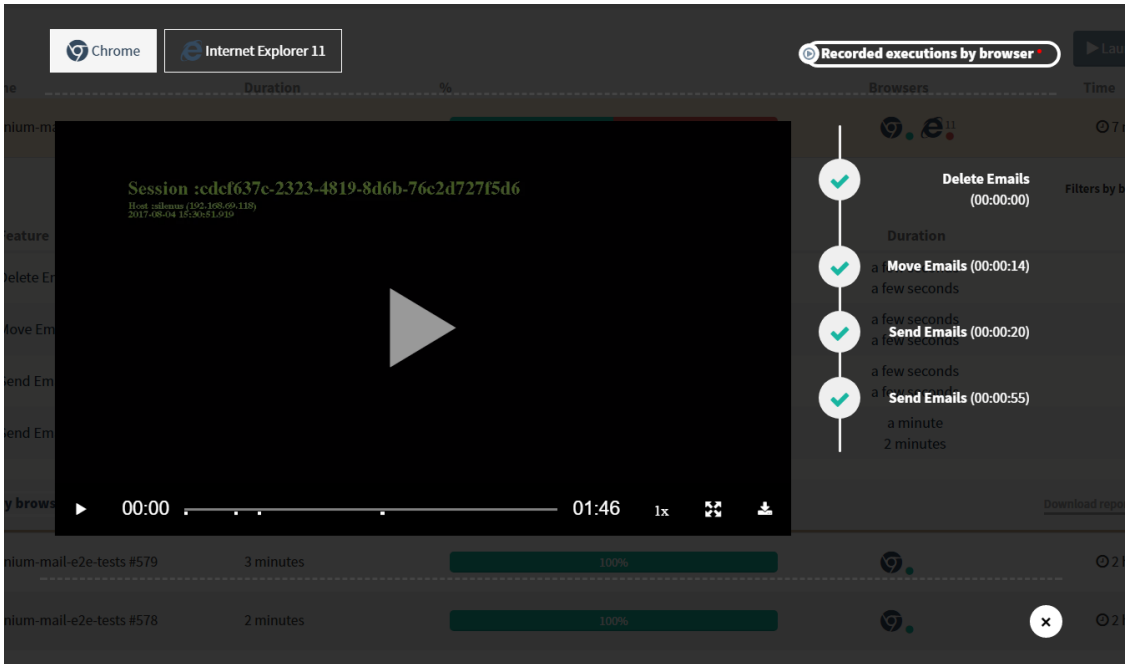
Status	Feature	Result	Passing / Failing	Duration
🟢	Delete Emails	🟢	2 / 0	a few seconds
🔴	Login	🔴	1 / 1	a few seconds
🟢	Move Emails	🟢	1 / 0	a few seconds
🟢	Send Emails (Loading External Data CSV)	🟢	4 / 0	a few seconds
🟢	Send Emails	🟢	6 / 0	a minute

For the execution expanded it can be seen:

- features tested
- the results detailed for each feature (with filters)
- videos of the execution

- download the report of the executions (MS excel and PDF)
- remove execution

Click at "Recorded executions by browser" to see the videos of the recorded executions:

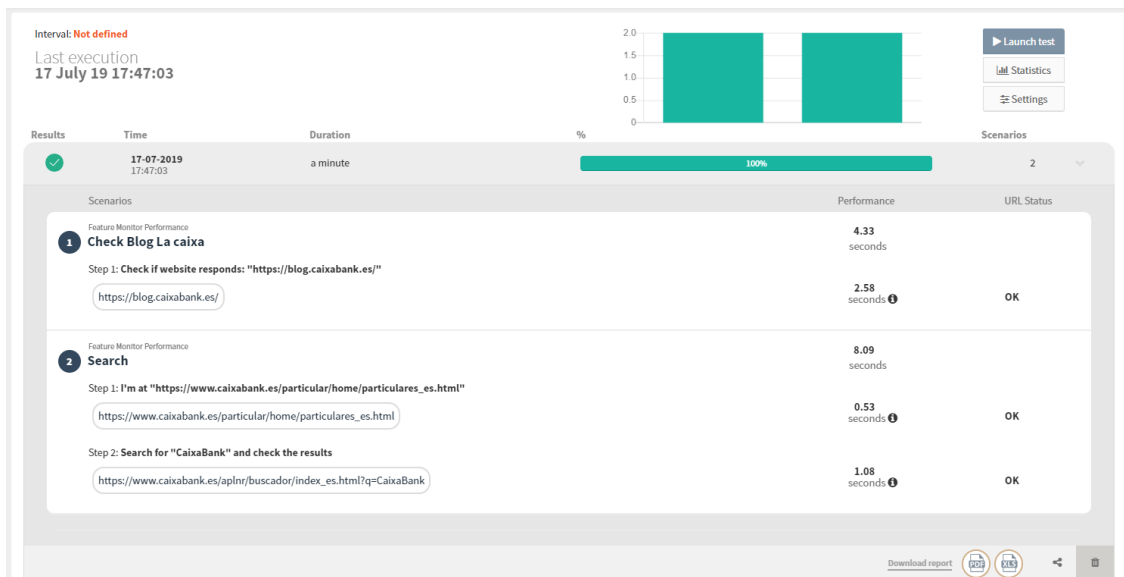


For the videos (at execution) it can be seen:

- videos for the browsers tested
- features to navigate through the video
- cue points when the feature starts on the video

### Monitoring project

The result of an execution of a Monitoring project is presented at the execution page:

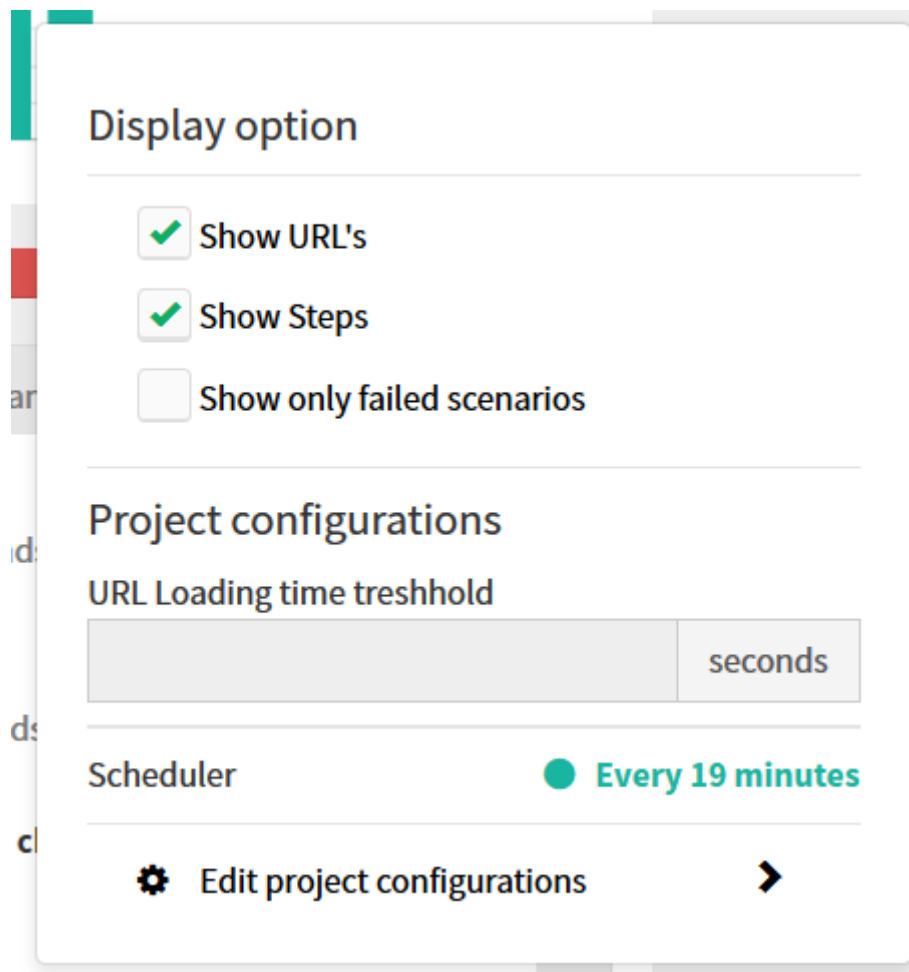


At execution page, we can see the time of each URL took to load. Also, we can see the URL status, and the time it took to execute the scenario.

### Display options

The information at the execution page can be filtered to show/hide some details.

To open the `Display options` click at `Settings`:



The display options allow us to:

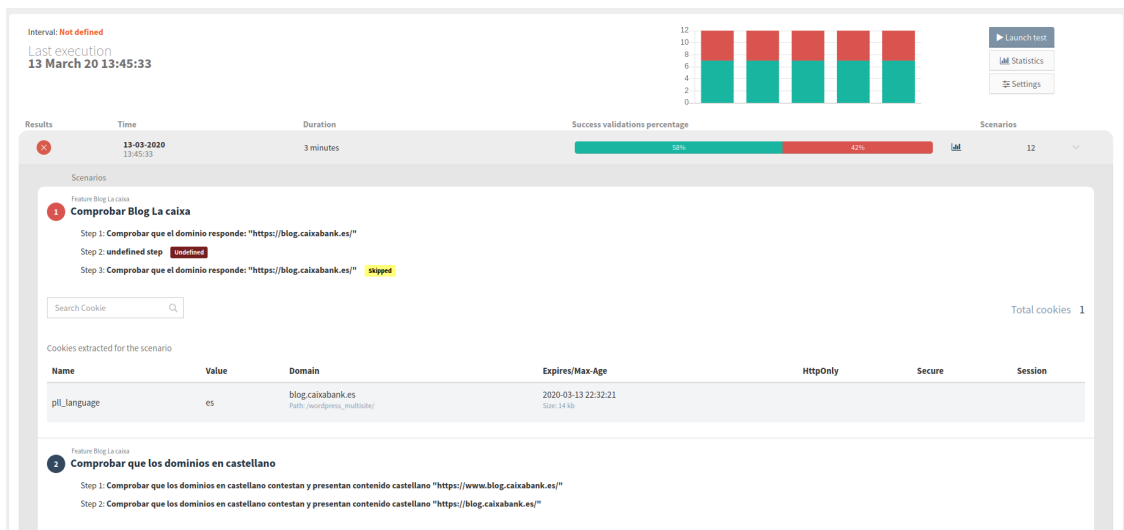
- Show/Hide the URL's
- Show/Hide the steps
- Show only the failed scenarios
- Consult the URL loading time threshold (This property is configured at the project configurations at the advanced tab and warns the user (at the execution page) when the loading time of a URL is higher than the loading time threshold)
- Consult the scheduler (The scheduled is configured at the project configurations at the reports and scheduling tab)

### Cookie project

The Cookie project is divided in two projects: Cookie Report project and Cookie Crawler project.

### Cookie Report project

The result of an execution of a Cookie Report project is presented at the execution page:



The screenshot shows the execution page of a test scenario. At the top, it indicates the interval is 'Not defined' and the last execution was on 13 March 20 at 13:45:33. A bar chart shows the success rate of validations, with a progress bar indicating 58% success and 42% failure. Below this, a table lists the results for two scenarios. The first scenario, 'Comprobar Blog La caixa', has three steps: Step 1 (successful), Step 2 (undefined), and Step 3 (skipped). A search bar for cookies is present, showing 'Total cookies 1'. A table lists the extracted cookies:

Name	Value	Domain	Expires/Max-Age	HttpOnly	Secure	Session
plL_language	es	blog.caixabank.es Path: /wordpress_multisite/	2020-03-13 22:32:21 Size: 14 kb			

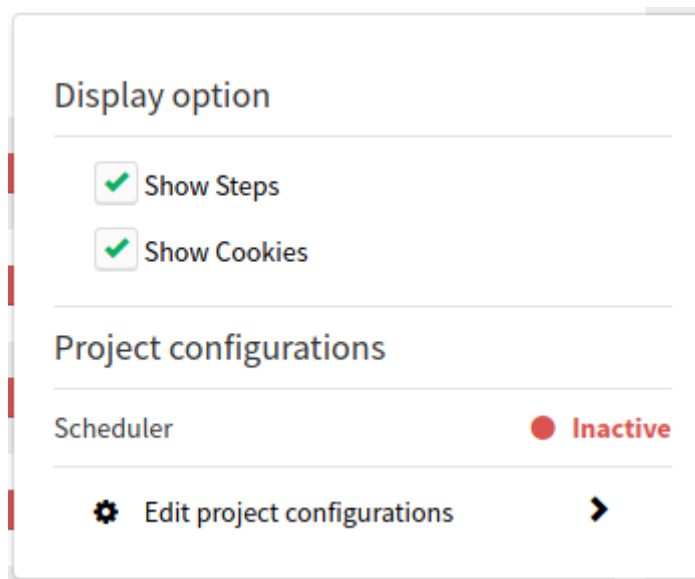
The second scenario, 'Comprobar que los dominios en castellano', has two steps: Step 1 (successful) and Step 2 (successful).

At execution page, we can see at the end of each scenario a table (with a filter) of all browser cookies at the end of a scenario.

## Display options

The information at the execution page can be filtered to show/hide some details.

To open the `Display options` click at `Settings`:





The 'Display option' dialog shows the following settings:

- Show Steps
- Show Cookies

Project configurations

Scheduler ● Inactive

 Edit project configurations 

The display options allow us to:

- Show/Hide the steps
- Show/Hide the cookies table
- Consult the scheduler (The scheduled is configured at the project configurations at the reports and scheduling tab)

## Cookie Crawler project

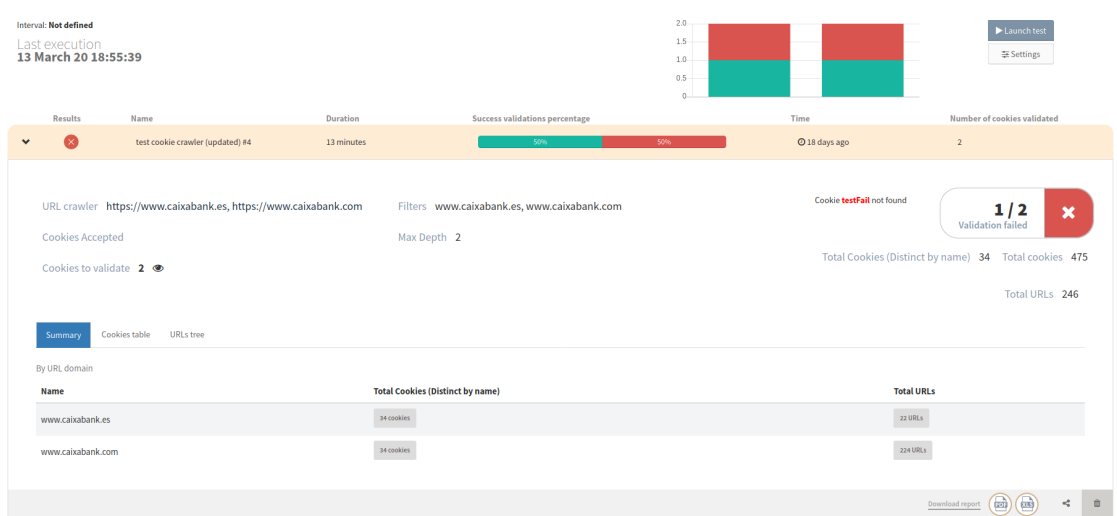
The Cookie Crawler project generates a report of the all browser cookies found during the crawl of the websites.

For each execution it can be seen:

- URL crawler - The URL(s) that was crawled
- Filters - Domains filtered
- Cookies Accepted - Tells whether the cookie warning was accepted or not
- Cookies to validate - Cookies validated during the crawl (click at the eye to see the validations in a table)
- Max Depth - Depth of the crawl
- Validation failed - The cookies not found at the report
- Total Cookies (Distinct by name) - Number of different cookies found at the report distinct by name
- Total cookies - Number of different cookies found at the report
- Total URLs - Number of different URLs crawled

There are several tabs to present execution information: Summary, Cookie table and URLs tree.

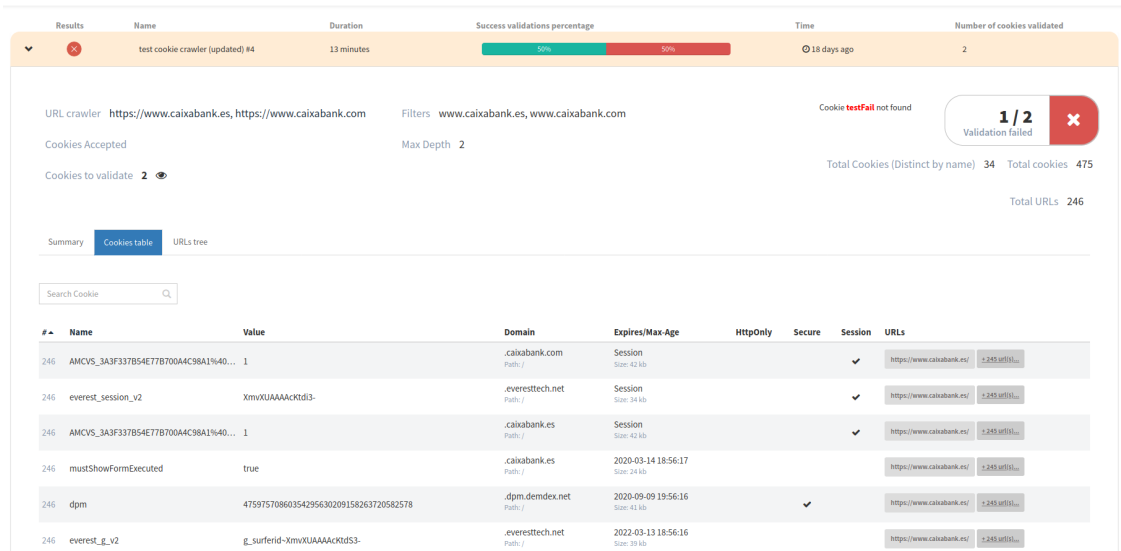
**Summary:**



This table presents the information By URL domain.

For the screenshot above we can see that for the domain `www.caixabank.es`, we found 34 cookies (Distinct by name) in a total of 22 URLs.

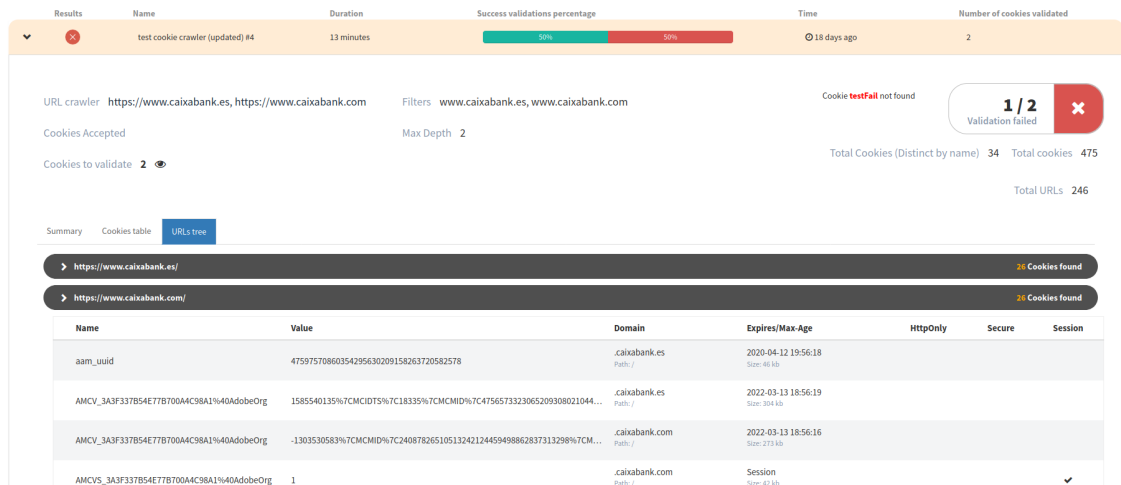
**Cookie table:**



#	Name	Value	Domain	Expires/Max-Age	HttpOnly	Secure	Session	URLs
246	AMCVS_3A3F337B54E77B700A4C98A1%40...	1	.caixabank.com	Session Size: 42 kb			✓	https://www.caixabank.es/   +245 urls...
246	everest_session_v2	XmvXUAAAACKd3-	everesttech.net	Session Size: 24 kb			✓	https://www.caixabank.es/   +245 urls...
246	AMCVS_3A3F337B54E77B700A4C98A1%40...	1	.caixabank.es	Session Size: 42 kb			✓	https://www.caixabank.es/   +245 urls...
246	mustShowFormExecuted	true	.caixabank.es	2020-03-14 18:56:17 Size: 24 kb				https://www.caixabank.es/   +245 urls...
246	dpm	47597570860354295630209158263720582578	dpm.demdex.net	2020-09-09 19:56:16 Size: 11 kb		✓		https://www.caixabank.es/   +245 urls...
246	everest_g_v2	g_surferid-XmvXUAAAACKd3-	everesttech.net	2022-03-13 18:56:16 Size: 29 kb				https://www.caixabank.es/   +245 urls...

This table presents the all the cookie's information retrieved at the crawler for the URL(s) crawled, and the URL(s) where the cookie was found.

### URLs tree:



Name	Value	Domain	Expires/Max-Age	HttpOnly	Secure	Session
aam_uid	47597570860354295630209158263720582578	caixabank.es	2020-04-12 19:56:18 Size: 46 kb			
AMCV_3A3F337B54E77B700A4C98A1%40AdobeOrg	1585540135%7CMCIDT5%7C18335%7CMCMID%7C4756573323065209308021044...	.caixabank.es	2022-03-13 18:56:19 Size: 394 kb			
AMCV_3A3F337B54E77B700A4C98A1%40AdobeOrg	-1303530583%7CMCMID%7C24087826510513242124459498862837313298%7CM...	caixabank.com	2022-03-13 18:56:16 Size: 273 kb			
AMCVS_3A3F337B54E77B700A4C98A1%40AdobeOrg	1	.caixabank.com	Session Size: 42 kb			✓

This view shows the URLs tree generated by the crawl, with the cookies found for each URL.

### Display options

The information at the execution page can be filtered to show/hide some details.

To open the `Display options` click at `Settings`:

### Display option

---

Show new cookies only (Tree view)

Show all cookie information (Table view)

---

### Project configurations

Max Depth

Accept Cookies

Number of Validation(s)



---

Scheduler ● Inactive

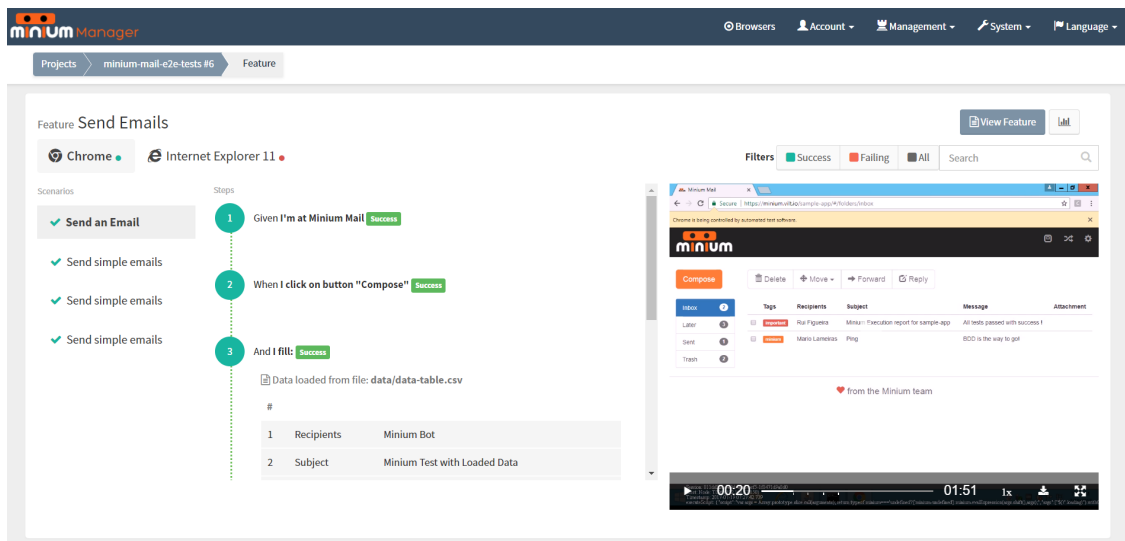
---

⚙ Edit project configurations ➤

The display options allow us to:

- Show new cookies only (Tree view) - Only shows the new cookies at the URLs tree in the child levels. The default is show every cookie (even the repeated ones).
- Show all cookie information (Table view) - Shows all the cookie information (this is the fields: domain, Expires/Max-Age, HttpOnly, Secure and Session). The default is to show the Name, Value and URLs aggregated.
- Max Depth - The Max depth configured at project configurations
- Accept Cookies - Shows if accept cookies are checked or not at project configurations
- Number of Validation(s) - Shows the number of validations configured at project configurations
- Consult the scheduler (The scheduled is configured at the project configurations at the reports and scheduling tab)

On the feature (for web application testing) view check the scenarios of this particular feature:

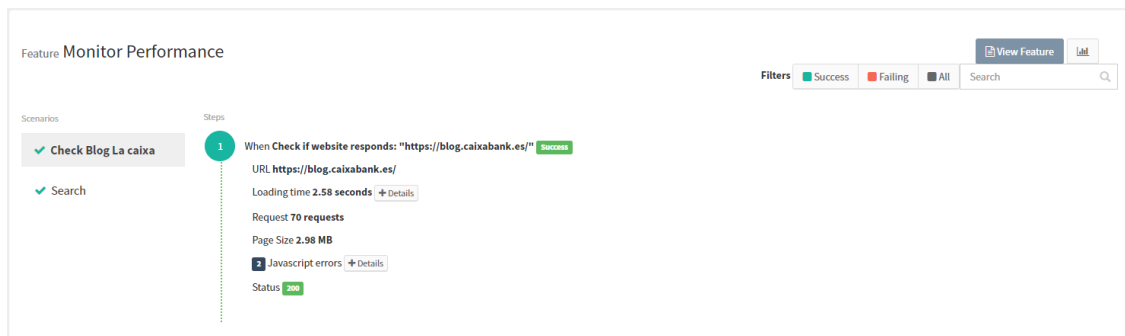


When expands a scenario see all the steps and results. Is easily switch to the results of another browser. More functionalities:

- **Screenshots** - Minium Manager provides screenshots of the application at the moment of the failure. So it possible see the state of the application in the moment of failure.
- **Error messages**
- **Links to the feature or step file for each step.**
- **Evolution chart for the projects**
- **Video with the cue points when the scenarios start**

## Monitoring project

Feature page of a Monitoring project scenario:



At the feature page, we can see more details about the page load performance:

- The url loaded
- The loading time with the times of the Backend Performance, Frontend Performance and DOM Content Loading
- The number of requests
- The Javascript Errors (if any)

- The URL status code

## Cookie project

Feature page of a Cookie Report project scenario:

Feature **Blog La caixa**

Filters: Success, Failing, All

Scenarios

- ✗ Comprobar Blog La caixa
- ✓ Comprobar que los dominios en castellano
- ✓ Comprobar que los dominios en catalan
- ✓ Comprobar carga toda entera en una página única.
- ✓ Comprobar que se listan posts
- ✓ Comprobar que se listan posts
- ✓ Comprobar que se carga la nueva noticia.
- ✓ Buscar en el buscador
- ✗ Comprobar checkstatus
- ✗ Comprobar checkstatus
- ✗ Comprobar checkstatus

Steps

- 1 Cuando Comprobar que el dominio responde: "https://blog.caixabank.es/" **Success**
- 2 Cuando undefined step **Undefined**
- 3 Cuando Comprobar que el dominio responde: "https://blog.caixabank.es/" **Skipped**

Search Cookie

Total cookies 1

Cookies extracted for the scenario

Name	Value	Domain	Expires/Max-Age	HttpOnly	Secure	Session
pll_language	es	blog.caixabank.es Path: /wordpress_multisite/	2020-03-13 22:32:21 Site: 14 kb			

We can see at the end of each scenario a table (with a filter) of all browser cookies.



### Note:

The Feature overview page is not available to the Cookie Crawler project.

## 6.1. Feature overview

Overview of all the features, by clicking on the button View Feature.

Feature **Send Emails**

Filters: Success, Failing, All

Scenarios

- ✓ Send an Email
- ✓ Send simple emails
- ✓ Send simple emails
- ✓ Send simple emails

Steps

- 1 Given I'm at Minium Mail **Success**
- 2 When I click on button "Compose" **Success**
- 3 And I fill: **Success**

Data loaded from file: data/data-table.csv

#	Recipients	Minium Bot
1	Recipients	Minium Bot
2	Subject	Minium Test with Loaded Data

View Feature


View the result of each step in each browser that the feature ran.

### Delete Emails

Background:  
  Given I'm at Minium Mail

Scenario: Delete an email  
  Given an email with Subject "Minium Can!" exists  
  When I delete an email with Subject "Minium Can!"  
  And I navigate to section "Trash"  
  Then I should see an email with:

Subject	Minium Can!
Recipients	Minium Bot

Scenario: Delete an email from trash  
  Given I'm at section "Trash"  
  And an email with Subject "Phasellus vitae interdum nulla." exists  
  When I delete an email with Subject "Phasellus vitae interdum nulla."  
  Then I shouldn't see an email with:

Subject	Minium Can!
---------	-------------

Close

## 6.2. Reports


Receive the reports of each test execution, in PDF and/or Excel formats, via email. Alternatively, download them directly in Minium Manager. To choose the formats to receive via email, go to the configurations of the project and expand the advanced configurations:


### Emails

**Recipients** Email addresses to receive reports of the executions

Send only when there are test failures



**Report formats**

 PDF











 XLSX

To download a report, expand the results of a test execution and click in `Download report`:



#### Executions

Results	Name	Duration	%	Browsers	Time	Scenarios
	minium-mail-e2e-tests #7	3 minutes	93% <span style="color: red;">7%</span>		26 minutes ago	15

Execution #7

Status	Feature	Result	Passing / Failing	Duration
	Delete Emails		2 / 0	a few seconds
	Login		1 / 1	a few seconds
	Move Emails		1 / 0	a few seconds
	Send Emails (Loading External Data CSV)		4 / 0	a few seconds
	Send Emails		6 / 0	a minute

Download report

## 6.3. Daily Reports

The daily report is a custom report with your favorite projects that will be sent to your email with the status of the last completed execution in the last 24 hours.

Before being able to configure a daily report, first you need to set the email configured at [Account > Settings](#).



**Note:**

If Minium Manager is using LDAP, please contact your administrator to set your email at LDAP.

To create a daily report configuration, open the [Account](#) menu and click on [Daily Reports](#).

Choose the projects to be included in your custom daily report:

Projects

Choose the projects to be included in your custom daily report.

Search for projects   - Filter By Group -  - Filter By Label -

<input type="checkbox"/>	Name <input type="button" value="v"/>	Groups	Labels
<input type="checkbox"/>	authenticated_project		
<input type="checkbox"/>	google-test-git	<input type="button" value="qa.team"/>	<input type="button" value="staging"/>
<input type="checkbox"/>	google-test-svn		
<input type="checkbox"/>	minium-mail-e2e-tests	<input type="button" value="minium.team"/> <input type="button" value="eng.team"/> <input type="button" value="qa.team"/>	<input type="button" value="prod"/>
<input type="checkbox"/>	test		
<input type="checkbox"/>	test-jvm-project		
<input type="checkbox"/>	unauthenticated_project		

Show  Projects

Project(s) selected: Total selected: 0

None

After, schedule the time the email will be sent, with a summary of the past 24h.

### Scheduling

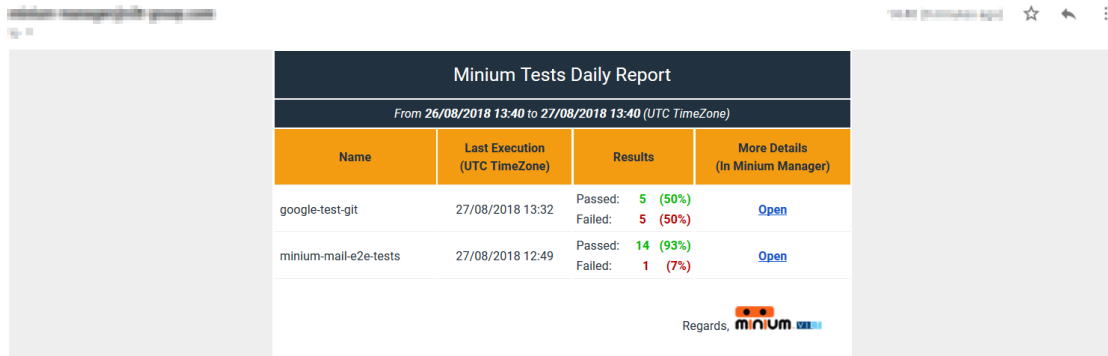
The report will be sent at the time you configure, with a summary of the past 24h.

Every:  at  :


Finally, click on the button "Create/Update Daily Report Configuration" to create/update the

daily reports:

Daily report of the past 24h (with errors) Inbox x



Minium Tests Daily Report			
From 26/08/2018 13:40 to 27/08/2018 13:40 (UTC TimeZone)			
Name	Last Execution (UTC TimeZone)	Results	More Details (In Minium Manager)
google-test-git	27/08/2018 13:32	Passed: 5 (50%) Failed: 5 (50%)	<a href="#">Open</a>
minium-mail-e2e-tests	27/08/2018 12:49	Passed: 14 (93%) Failed: 1 (7%)	<a href="#">Open</a>

Regards, 

To delete the configuration, click on the button "Delete Daily Report Configuration".

The daily report was configured to be sent at **06:00** with the status of the last completed execution in the last 24 hours for the following projects:

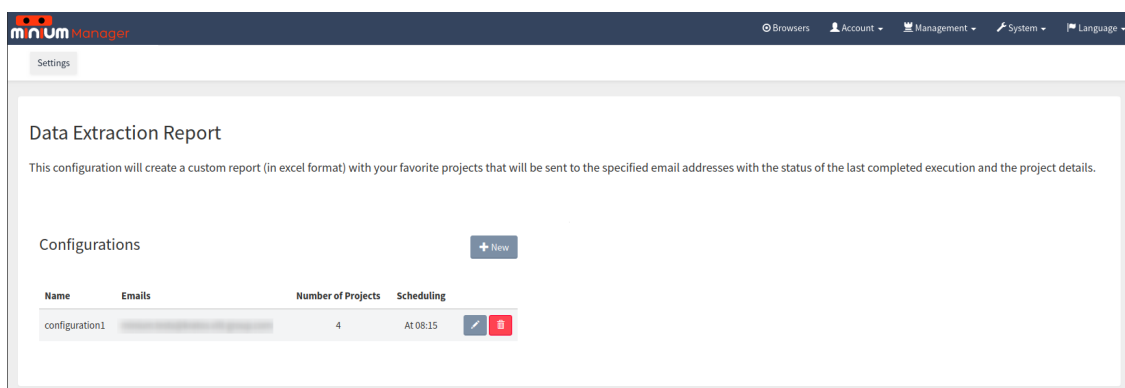
- authenticated\_project
- google-test-git
- google-test-svn
- minium-mail-e2e-tests
- test
- test-jvm-project
- unauthenticated\_project

Change Daily Report Configuration Delete Daily Report Configuration

## 6.4. Data Extraction Reports

The Data extraction report is a custom report (in excel format) with your favorite projects that will be sent to the specified email addresses with the status of the last completed execution, and the project details.

To create and/or consult the data extraction reports configurations, open the **Management** menu and click on **Data Extraction Report**.



**Data Extraction Report**

This configuration will create a custom report (in excel format) with your favorite projects that will be sent to the specified email addresses with the status of the last completed execution and the project details.

Configurations + New

Name	Emails	Number of Projects	Scheduling
configuration1	<a href="#">[Email Address]</a>	4	At 08:15 <span>[Edit]</span> <span>[Delete]</span>

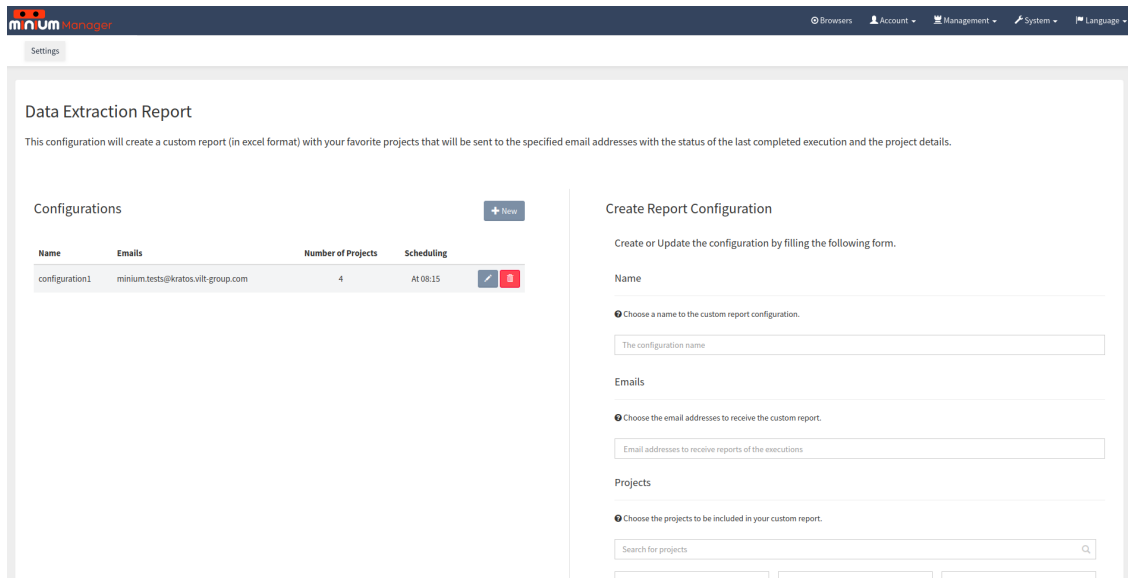
At the data extraction report page, you can see the current configurations created at the list

Configurations.

You can edit (by clicking in the **pencil** button) or delete (by clicking on the **trash** button) the configurations listed at the list `Configurations`.

Also, you can create a new configuration by clicking on the button `+ New`.

When you click the button `+ New`, a form will appear (similar to the edit form) on the right side of the page:



To create a new configuration, fill the form by add a new configuration name (unique value), the emails that will receive the configuration, choose the projects to be included in the report and finally, schedule the time the email will be sent, with the report.

**Note:**



To choose the projects, you have filters available to help. You can filter the project by name, type, group and label.

To save the configuration, click on the button `+ Save`.

The email received will list a summary of the projects selected (separated by type), and will have the report (in excel format) attached.

**Note:**



The subject of the email received is "Data Extraction Report".

## 7. Roles

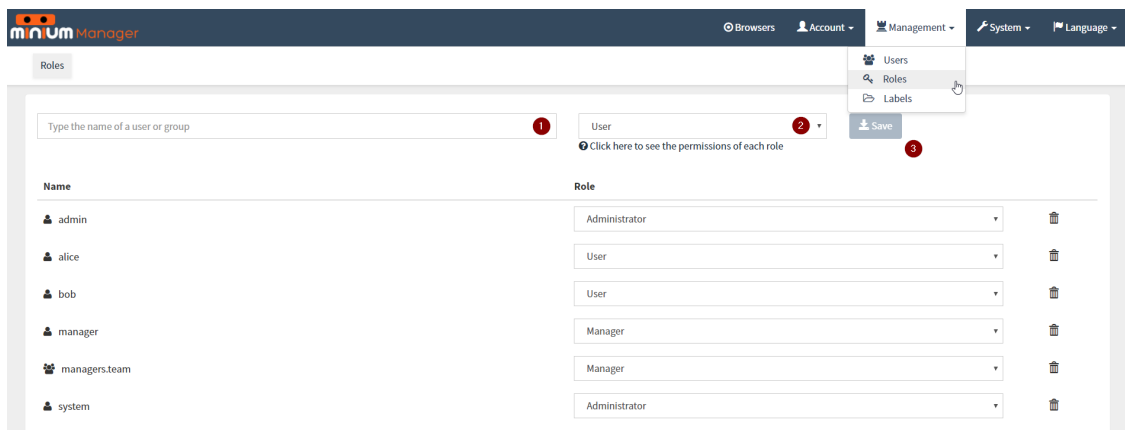
Roles allow granting different privileges to the users and the groups. The privileges associated with each role are described in the table below:

	User	Manager	Administrator
View projects	✓	✓	✓
Launch test executions	✓	✓	✓
Configure projects	✓	✓	✓
Create projects		✓	✓
Manage project labels		✓	✓
Manage users			✓
Manage roles			✓

### 7.1. Assign roles

To change the role of a user or group, navigate to **Management > Roles** at the navigation bar and:

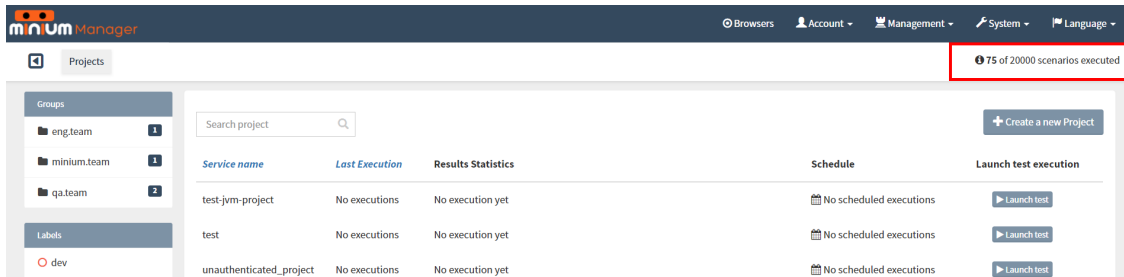
1. Introduce the name of the user or group in the search bar.
2. Select a role.
3. Click on **Save** button.



The screenshot displays the 'Roles' management page in the VILT Minium Manager. At the top, there is a navigation bar with 'Management > Roles' selected. Below the navigation bar, there is a search bar labeled 'Type the name of a user or group' (1). To the right of the search bar is a dropdown menu for selecting a role (2), with a 'Save' button (3) below it. The main content area is divided into two columns: 'Name' and 'Role'. The 'Name' column lists users: admin, alice, bob, manager, managers.team, and system. The 'Role' column shows the assigned role for each user: Administrator, User, User, Manager, Manager, and Administrator. Each user entry has a trash icon to its right.

## 7.2. License information

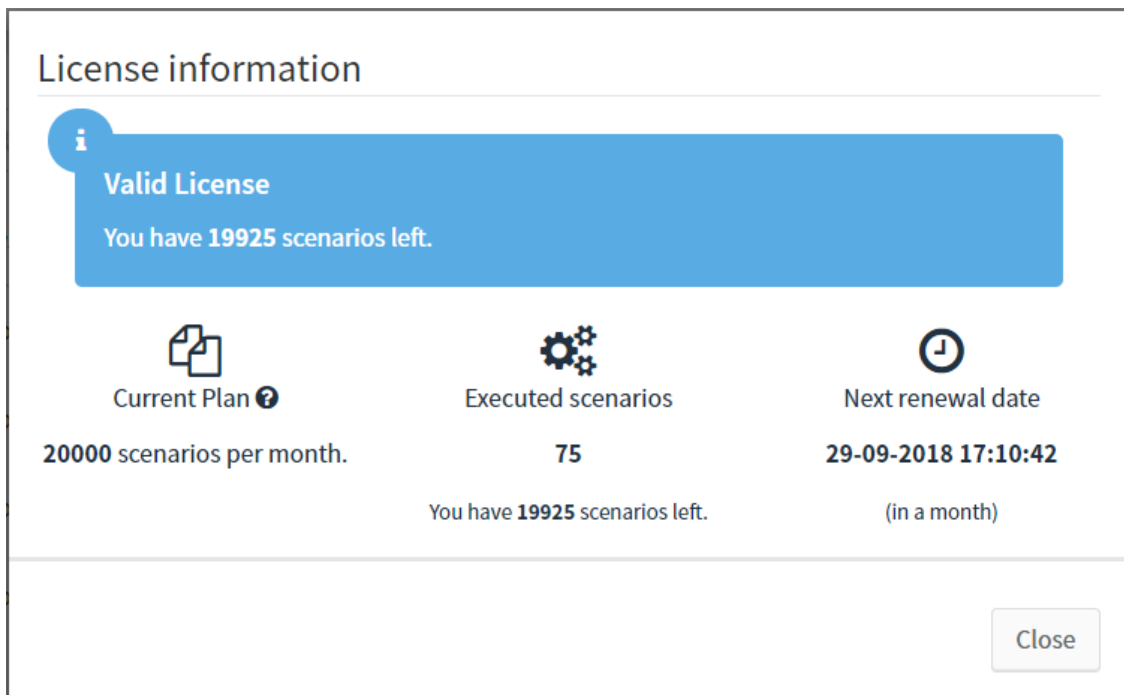
At the top right of the project list page exists the license information label, that gives the number of executed scenarios in this month, and the total of scenarios per month that to the current license



The screenshot shows the Minium Manager interface. At the top right, a red box highlights the license information label: "75 of 20000 scenarios executed". Below this, there is a table of projects with columns for Service name, Last Execution, Results Statistics, Schedule, and Launch test execution.

Service name	Last Execution	Results Statistics	Schedule	Launch test execution
test-jvm-project	No executions	No execution yet	No scheduled executions	Launch test
test	No executions	No execution yet	No scheduled executions	Launch test
unauthenticated_project	No executions	No execution yet	No scheduled executions	Launch test

By clicking on the license information label, it will open a modal with more detailed information about one license.



The modal window displays the following information:

- Valid License**: You have **19925** scenarios left.
- Current Plan**: 20000 scenarios per month.
- Executed scenarios**: 75. You have **19925** scenarios left.
- Next renewal date**: 29-09-2018 17:10:42 (in a month).

A "Close" button is located at the bottom right of the modal.

## Minium Recorder User Guide

**Minium Recorder** has been developed to ease and accelerate the creation of Minium tests. With the **Minium Recorder**, is no longer needed to write all the code for the step definitions by hand. Simply tell to the **Minium Recorder** to start recording the interactions with the browser, perform the actions that correspond to the defined step and an automation script is immediately available in **Minium Developer**.

But what really makes Minium Recorder different in relation to other automation script recorders is that, for each element that interacts, Minium Recorder will not generate just one CSS selector. Instead, is possible choose among a list of alternative expressions for each element that take advantage of the great Minium filtering methods that are used like for example `withLabel`, `below`, `rightOf`, etc, to build more readable and resilient expressions.

## 8. Configure Minium Developer

**Minium Recorder** is already configured at **Minium Developer Prime**, but you can configure an existing **Minium Recorder**.

To configure **Minium Developer** with the **Minium Recorder** extension, first you need to download the **Minium Recorder** extension.

Navigate to the folder where **Minium Developer** is installed, create a folder named "**extensions**", copy the **Minium Recorder** extension file to the new folder and rename the file to "minium-recorder.crx".

After, edit the `config/application-{windows,linux,macos}.yaml` file in order to add the following configuration (at the chrome webdriver):

```
minium:
  developer:
    webdrivers:
      - name: chrome
      ...
    chromeOptions:
      extensions:
        - ${app.home:./}/extensions/minium-recorder.crx
      preferences:
        devtools:
          preferences:
            panel-tabOrder: "{\"chrome-
extension://ggfeclafoeoejognlebilidgmgdlogMiniumRecorder\":10,\"elements
\":20,\"console\":30,\"sources\":40,\"network\":50,\"timeline\":60,\"heap_
profiler\":70,\"resources\":80,\"security\":90,\"audits\":100}"
```

## 9. Launch Minium Recorder

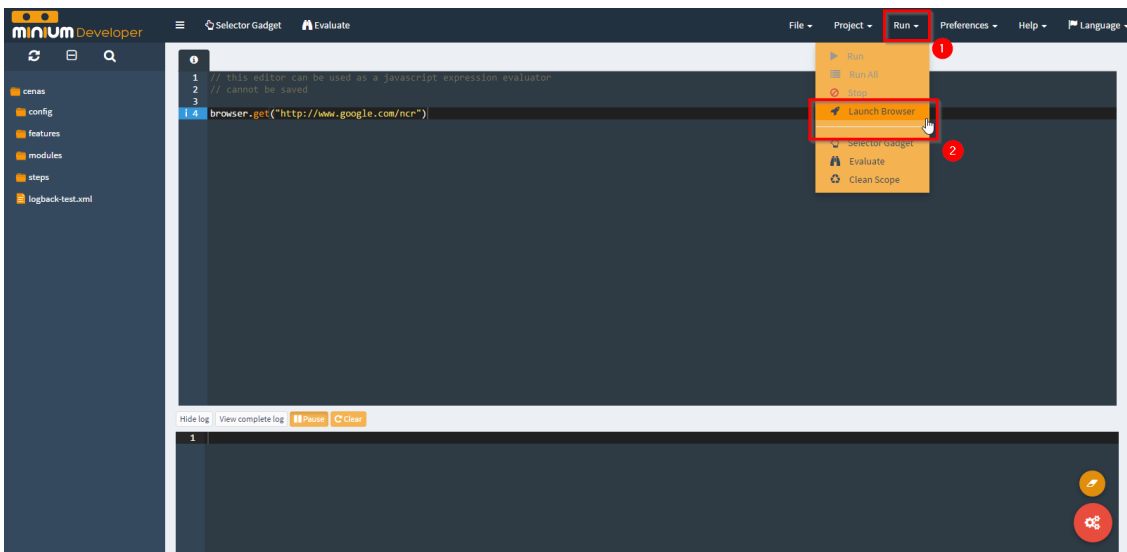
Launch **Minium Recorder** through **Minium Developer**.

Open **Minium Developer** and add the following command to the editor:

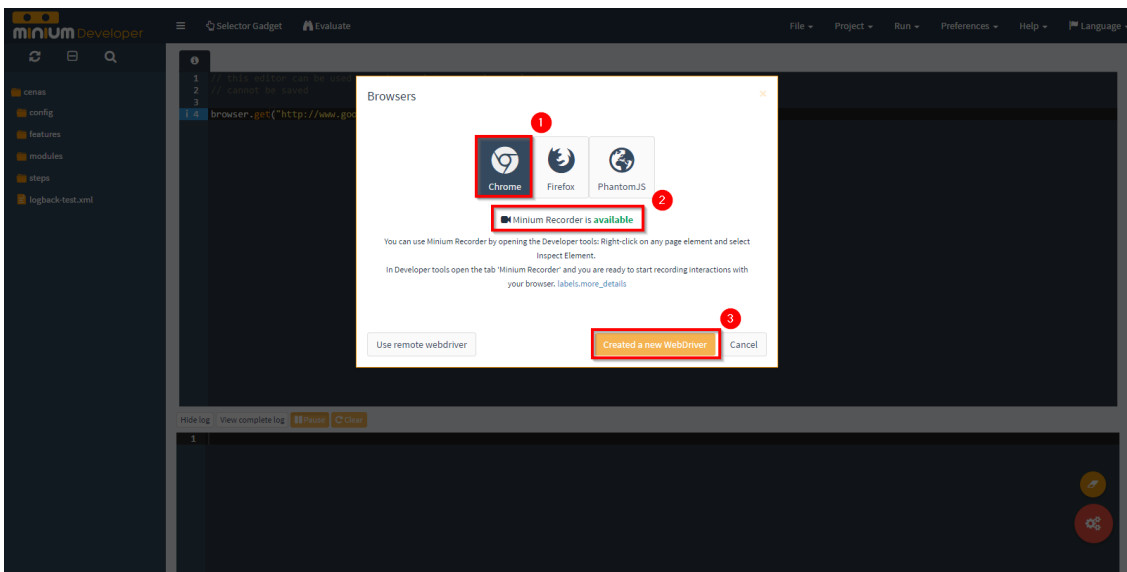
```
browser.get("http://www.google.com/ncr")
```

This will load the google search engine website, when the instance of Chrome is launched.

To launch Minium Recorder, use the (Ctrl + Enter) shortcut or select **Run > Launch Browser (1)** and **(2)** at the top-right menu:

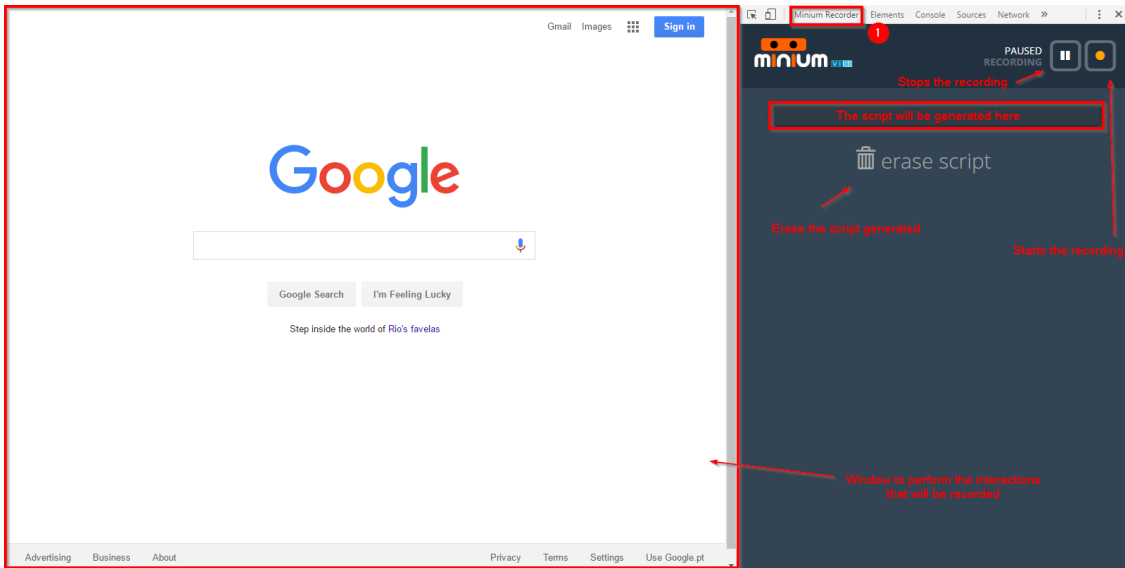


Select the **Chrome browser (1)**, check if **Minium Recorder is available (2)** and click **Created a new WebDriver (3)**:



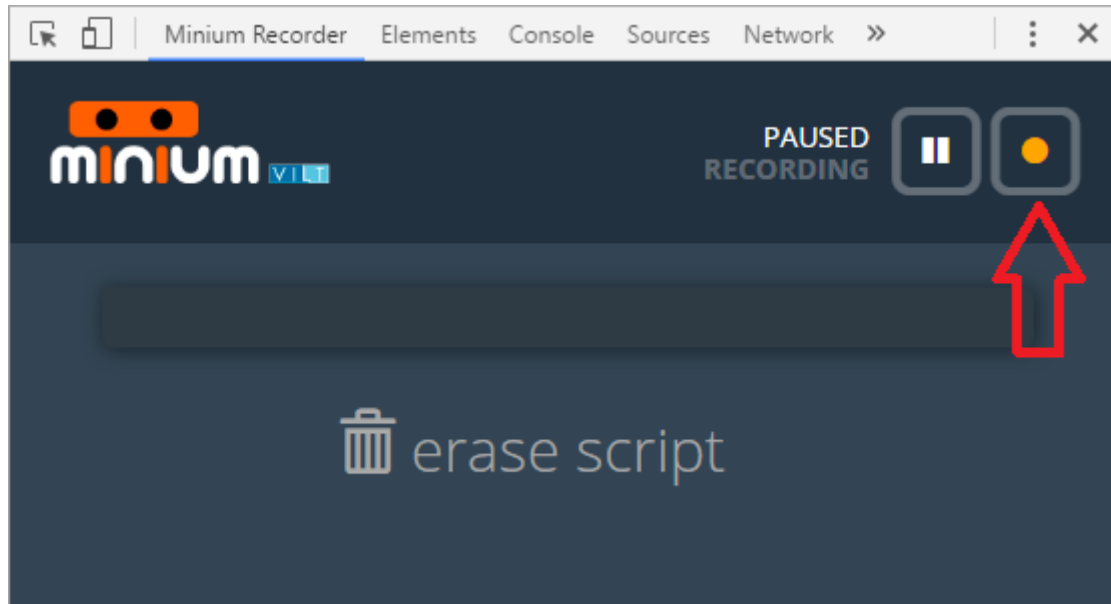
A new instance of Chrome will be launched with the Minium Recorder extension installed. In the instance of Chrome, open the `Developer tools` by using the (F12) shortcut (use another [shortcut](#)) or select `More Tools > Developer tools`.

After the `Developer tools` of the instance Chrome is opened, select the `Minium Recorder` tab (1):



## 10. Start recording

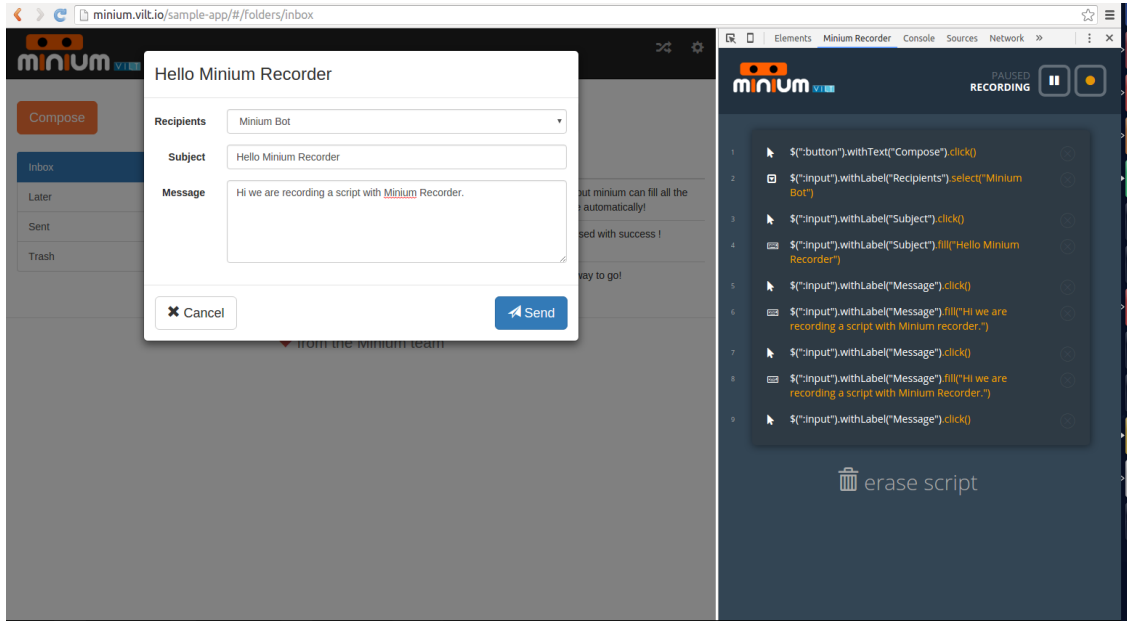
To start recording, click on the following button:



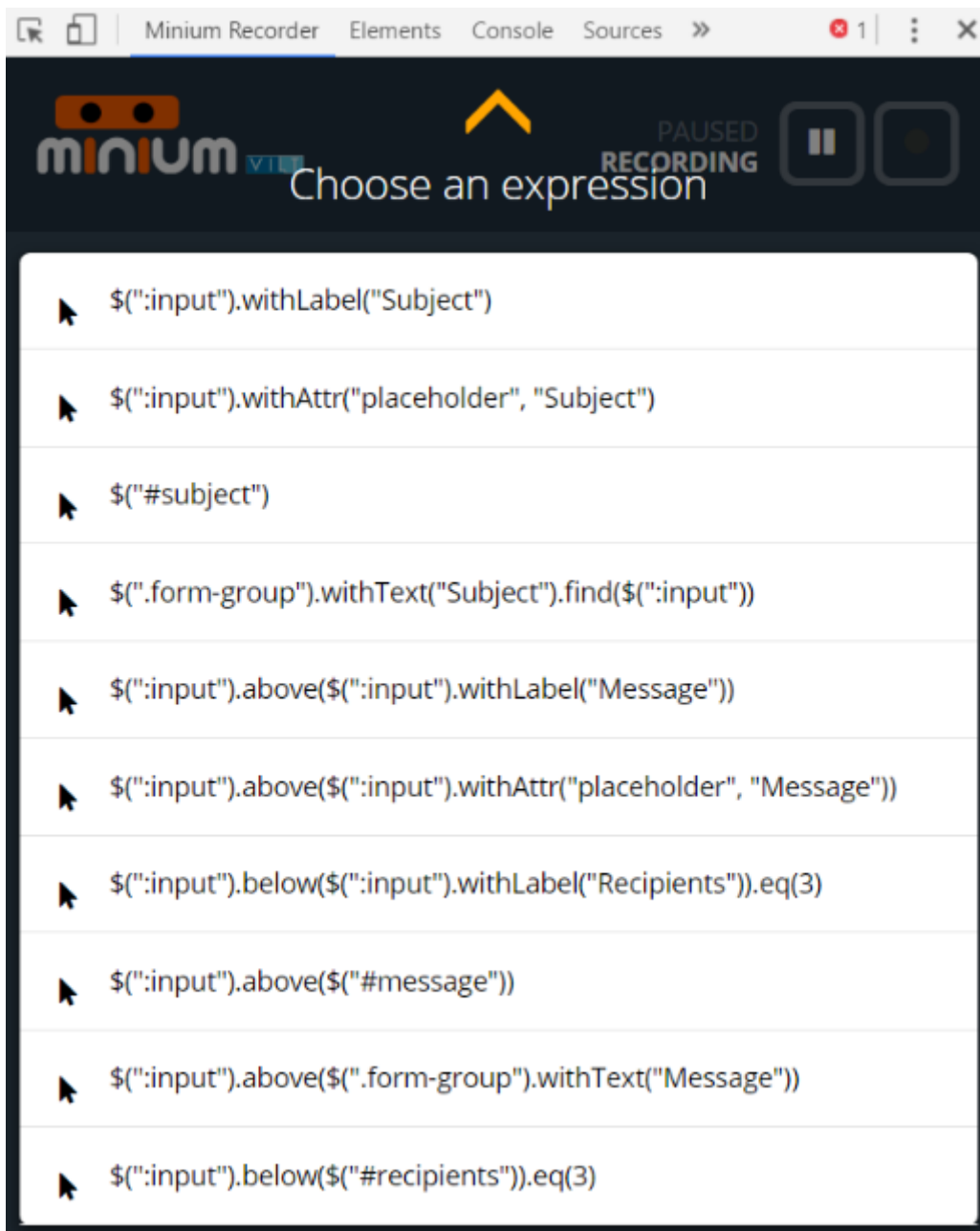
From now on, every interaction performed with the browser will be recorded.

## 11. Record interactions

For every action performed on the browser, like clicking on an element or filling a text input field, Minium Recorder will automatically generate the corresponding code. If typed an URL in the address bar, Minium Recorder will also generate the code to load that URL.



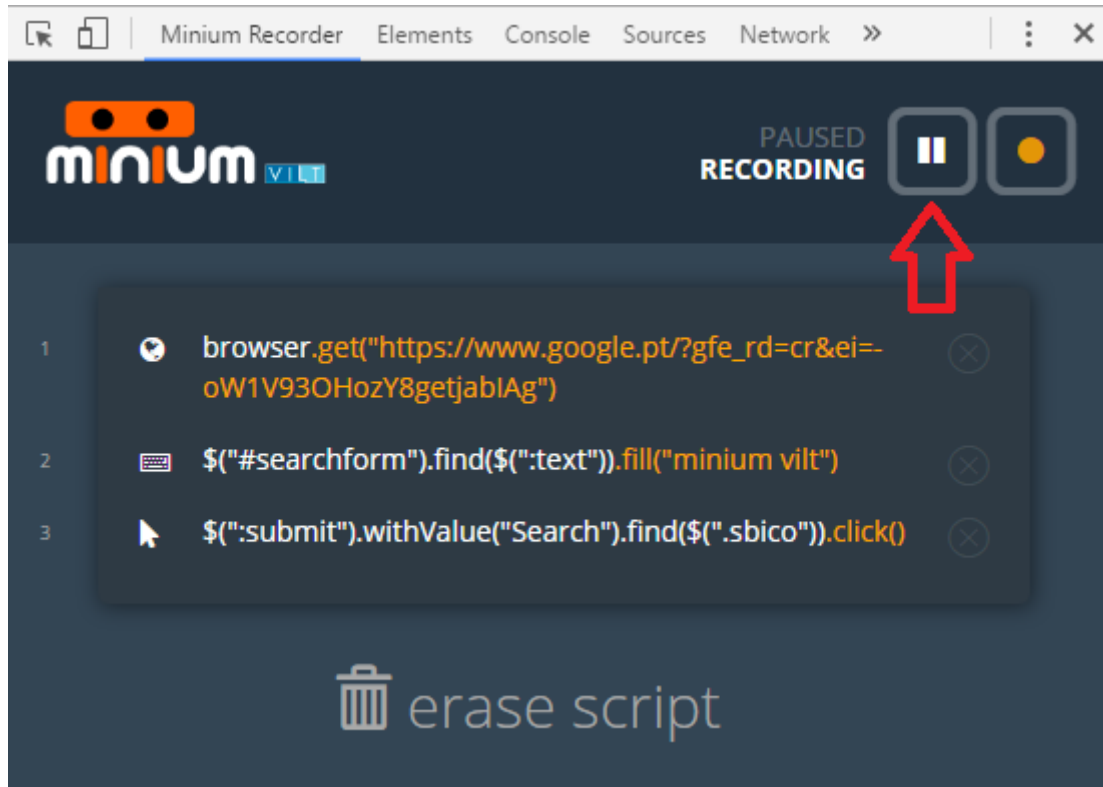
To try other expressions, click on the expression and a list of alternative expressions will be displayed:



To select an expression, just click on it.

## 12. Stop recording

Click on the pause button to stop a recording.



## 13. Import the recorded script in Minium Developer

To import a script in Minium Developer, right-click on the editor area, where is intended to put the script, and choose the option `Import recorded script`:

