



VILT Minium Manager

User Guide

Version 2.0



Contents

1. Introduction	1
2. Projects	2
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. . .	3
2.1.3. Click on a link and get the performance of the page	4
2.2. Setup a project	4
2.3. Project permissions	10
2.4. Organise projects by Labels or Groups	12
2.4.1. Manage Labels	12
2.4.2. Assign labels to the projects	13
2.4.3. Manage groups	14
Manage groups using LDAP authentication	15
Manage groups without LDAP authentication	15
2.4.4. Organise Projects	17
Clean project filters	18
2.5. Configuration profiles	19
2.6. Secret configuration properties	20
2.7. Delete a project	21
3. Check the available browsers	23
4. Project Statistics	24
5. Test executions	26
5.1. Launch test executions	26
5.1.1. Launch a test execution manually	26
Multiple Providers	28
5.1.2. Schedule test executions	29
Multiple Providers	31
5.1.3. REST API	31
5.2. Follow the progress of a test execution	34
6. Monitor test results	35
6.1. Feature overview	40
6.2. Reports	40
6.3. Daily Reports	41
7. Roles	44
7.1. Assign roles	44
7.2. License information	45
Minium Recorder User Guide	46
8. Configure Minium Developer	47
9. Launch Minium Recorder	48
10. Start recording	50
11. Record interactions	51
12. Stop recording	53

13. Import the recorded script in Minium Developer	54
---	-----------

1. Introduction

This document describes the user interface of the VILT - Minium Manager, version 2.0.

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.

There are two type of projects: Web application testing and Monitoring.

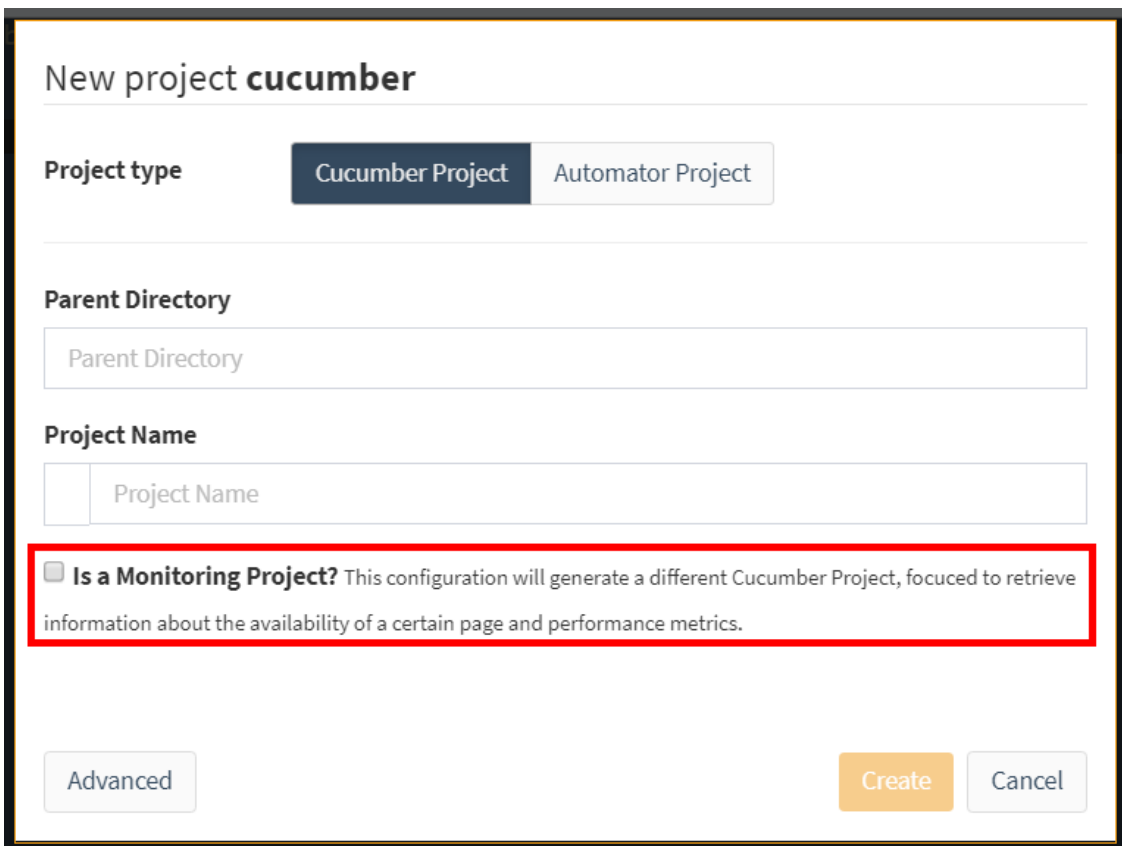
The Web application testing project is a Minium project that test a website in the same way a human would.

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 a 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.

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 in 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);
});

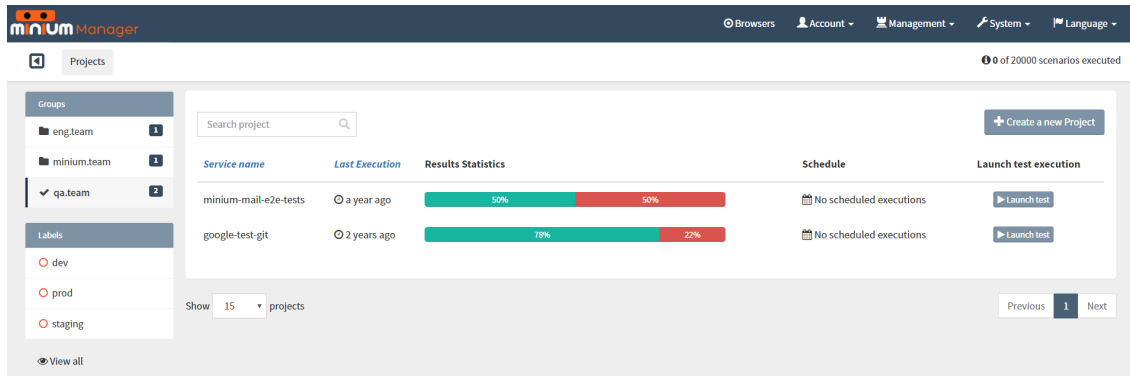
When(/^Search for "([\^"]*)" and check the results$/, function(search) {
  $("#cookies-accept-full a").click();
  $("#search-field").fill(search);
  $("#prebuscadorCabecera input[type='submit']").click();
  expect($(".search-result-block").to.exist());
  scenario.write(browser.getPerformance());
});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

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.

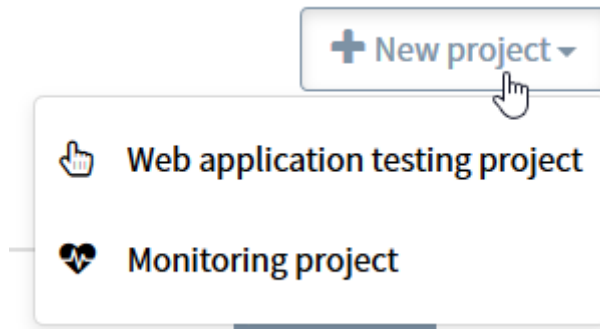
2.1.3. Click on a link and get the performance of the page

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



2.2. Setup a project

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



Then, fill the form:

Create or edit a Project ✕

General | Browsers | Reports and Scheduling | Advanced | Runtime | API

Name* ✓

Description

Labels + Add ▾

Repository

Type* Git Subversion

URL*

Username

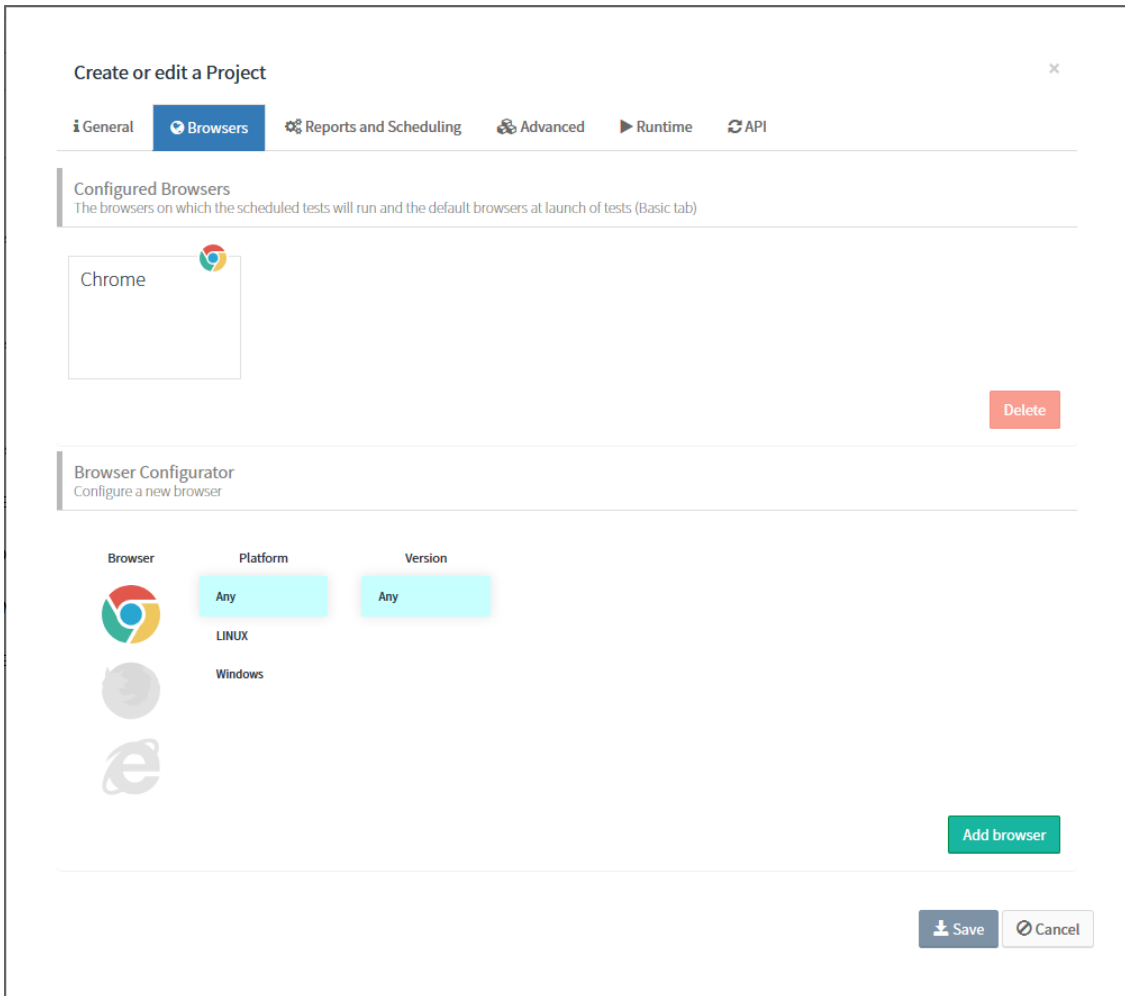
Password 👁

Branch

Save Cancel

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

Name	Display name of the project (e.g gmail-e2e-tests)
Description	Optionally provide a project description
Labels	Associated labels to a project in order to organize and filter projects.
Type	Type of SCM repository
Url	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): https://github.com/github/git.git //dolos/repos/gmail-e2e-tests
Username/Password	Credentials for authentication on the repository.
Branch	The name of the branch you want to execute (GIT only). The default value is <code>master</code>



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 projects.

Create or edit a Project ×

General
Reports and Scheduling
Advanced
Runtime
API

Emails

Recipients

Send only when there are test failures

Report formats PDF XLSX

Scheduling

Scheduler

Never run
 Every Day
 Every Week
 Every Month
 Every Day at midnight
 Custom

Scheduler Custom Configurations

Every: at :

Save
Cancel

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

Recipients	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".
Report formats	Set the format(s) that the report of execution will be sent to the email.
Scheduler	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.
Scheduler Custom Configurations	Set the schedule you want to execute the project and send the report of the execution with different levels of granularity.

Project Configurations
✕

Monitor page load performance. Make informed optimization armed with the size and load time of your applications.

General
Reports and Scheduling
Advanced
Runtime
API

Permissions

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

+ Append

Max number of test executions stored

URL Loading time treshhold seconds

Maximum time to load a page (in seconds).

Module

Delete Project
Update
Cancel

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

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

Create or edit a Project ×

General
Browsers
Reports and Scheduling
Advanced
▶ Runtime
API

Cucumber tags

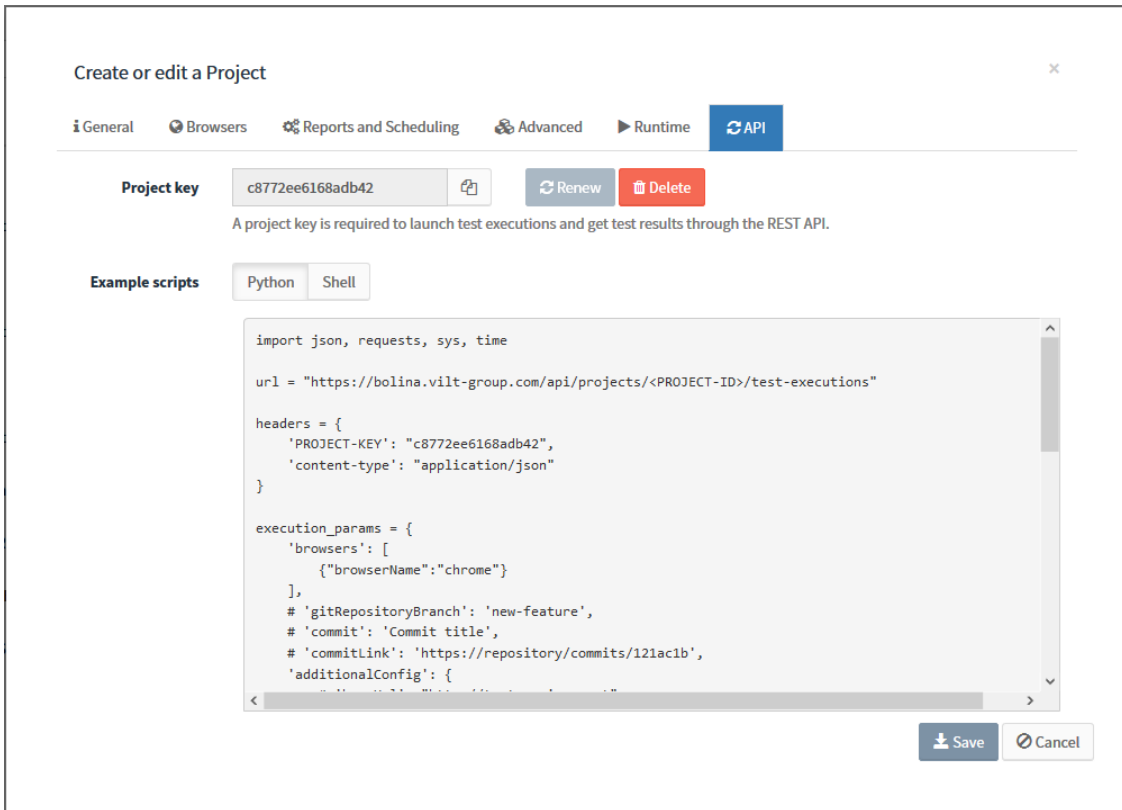
Additional profiles

Configuration properties 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.

Property	Value	
<input type="text"/>	<input type="text"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>

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

Cucumber tags	Use this option to tell Minium Manager that only run features or scenarios that have certain tags.
Additional Profiles	Enable any of the profiles that are configured in the <code>./config/application.yml</code> project file. See Configuration profiles for more details.
Configuration properties	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.



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

Project key	Generate a project key (required to launch test executions and get test results through the REST API).
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.

2.3. 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`.

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"/>	

+ Append

Max number of test executions stored

Module

Delete Project

Update
Cancel

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



Note:

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

bob ×

alice ×

+ Append

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

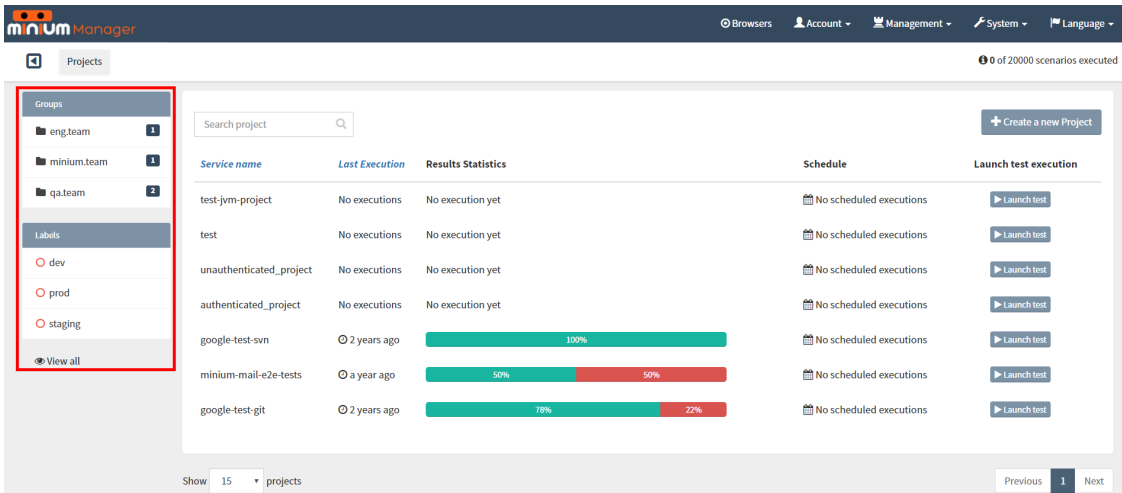
Module

Delete Project

Update
Cancel

2.4. Organise projects by Labels or Groups

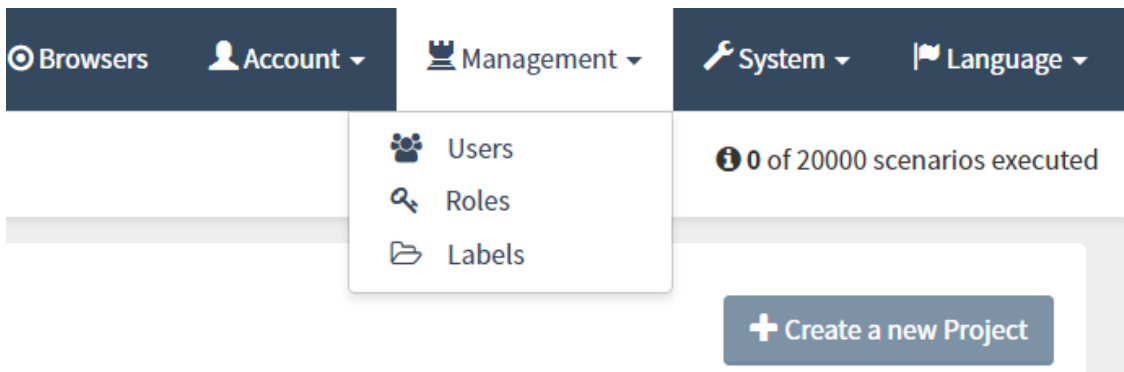
Labels were meant to be used only to filter projects in the Projects page providing a useful way to organize the projects.



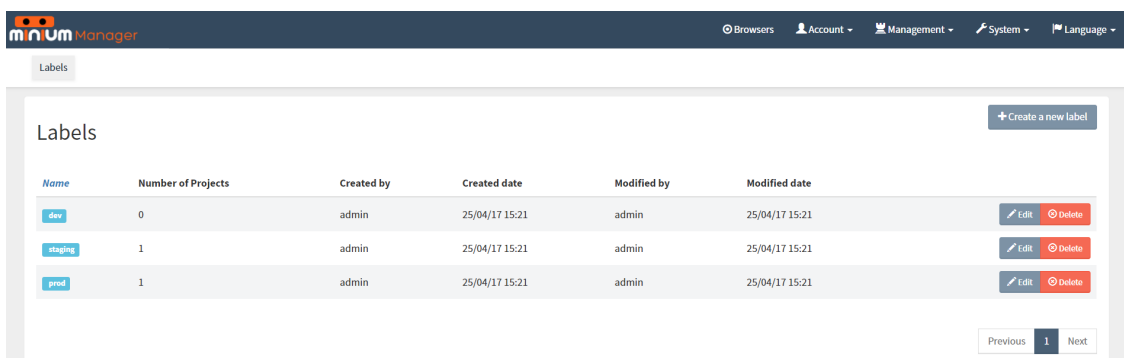
Service name	Last Execution	Results Statistics	Schedule	Launch test execution
test-jvm-project	No executions	No execution yet	No scheduled executions	Launch test
test	No executions	No execution yet	No scheduled executions	Launch test
unauthenticated_project	No executions	No execution yet	No scheduled executions	Launch test
authenticated_project	No executions	No execution yet	No scheduled executions	Launch test
google-test-svn	2 years ago	100%	No scheduled executions	Launch test
minium-mail-e2e-tests	1 year ago	50%	No scheduled executions	Launch test
google-test-git	2 years ago	78%	No scheduled executions	Launch test

2.4.1. 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.

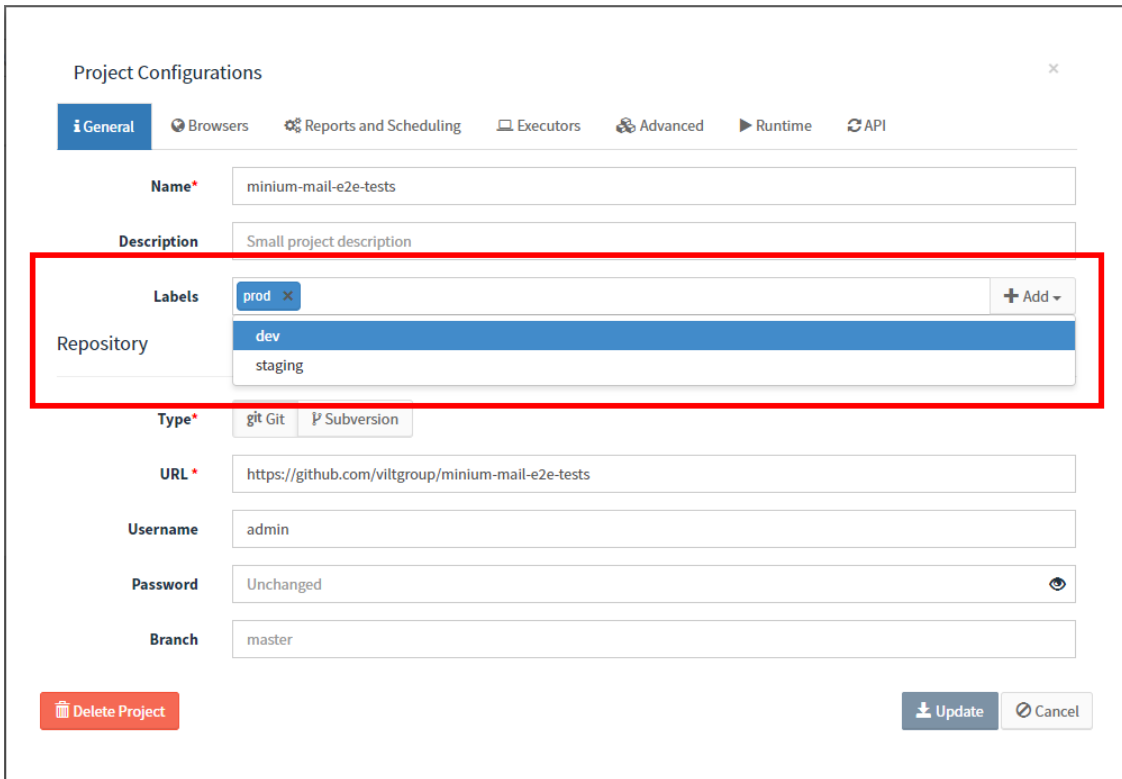


To create a new label, click on `Create a new label`. To delete or edit an existing label, click on `Edit` or `Delete` respectively.

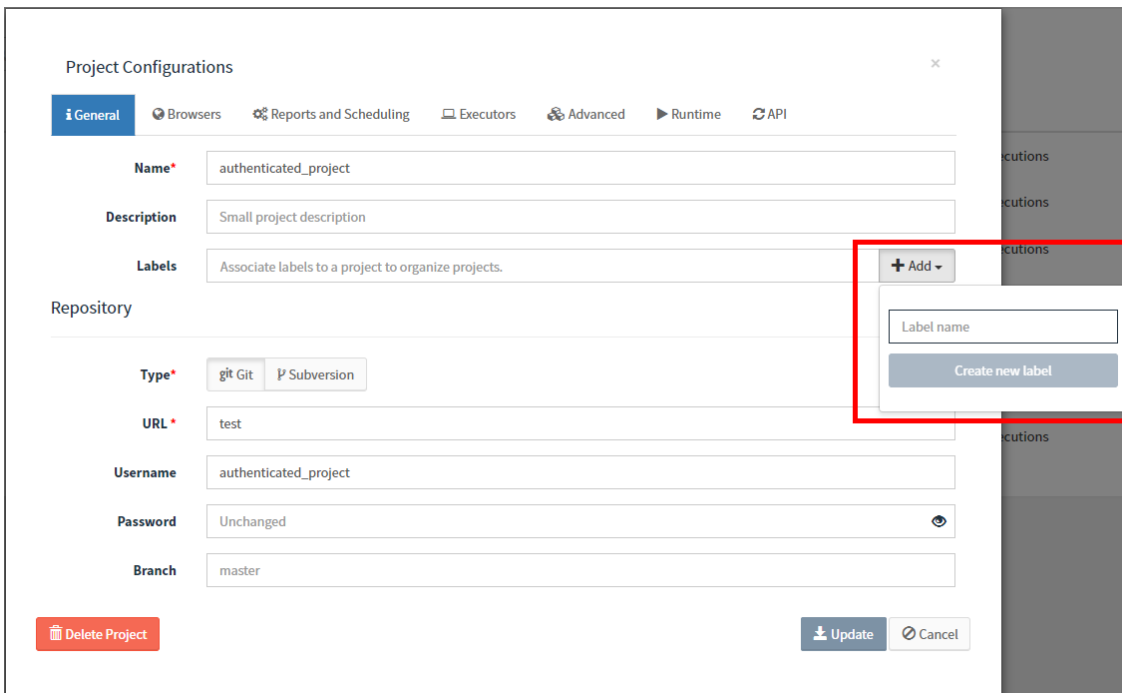


2.4.2. Assign labels to the projects

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



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



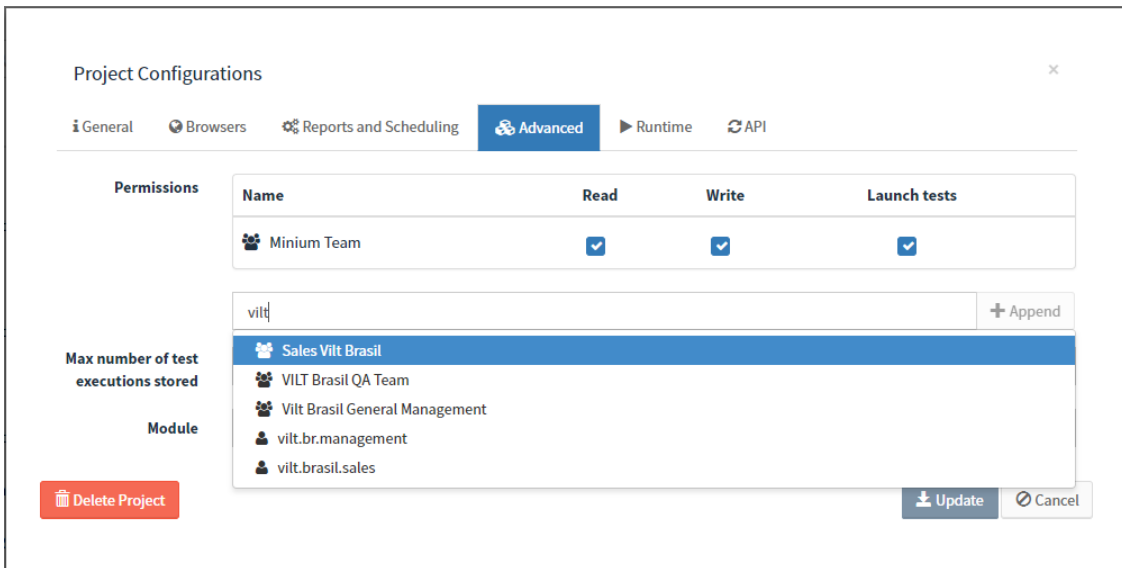
2.4.3. Manage groups

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 a 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 dropdown for groups.

Permissions	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

Buttons: Delete Project, Update, Cancel

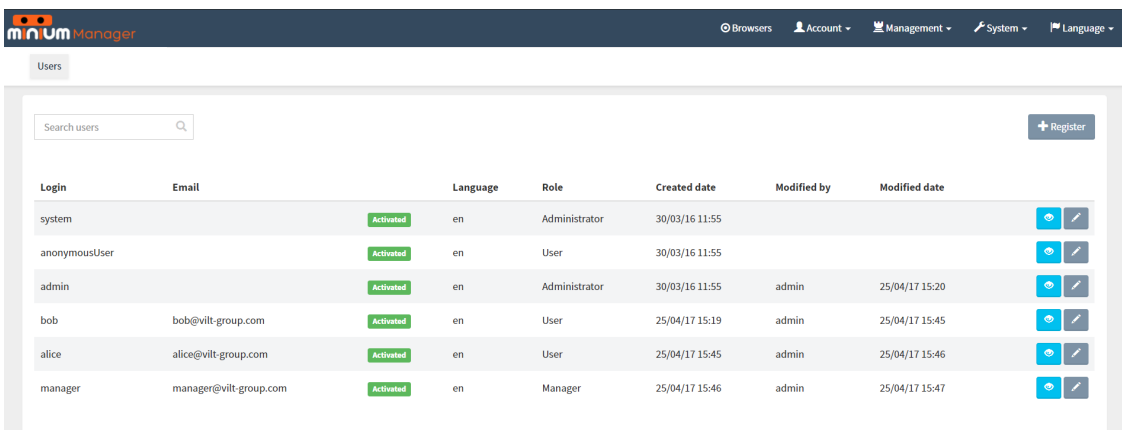
Manage groups without LDAP authentication

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



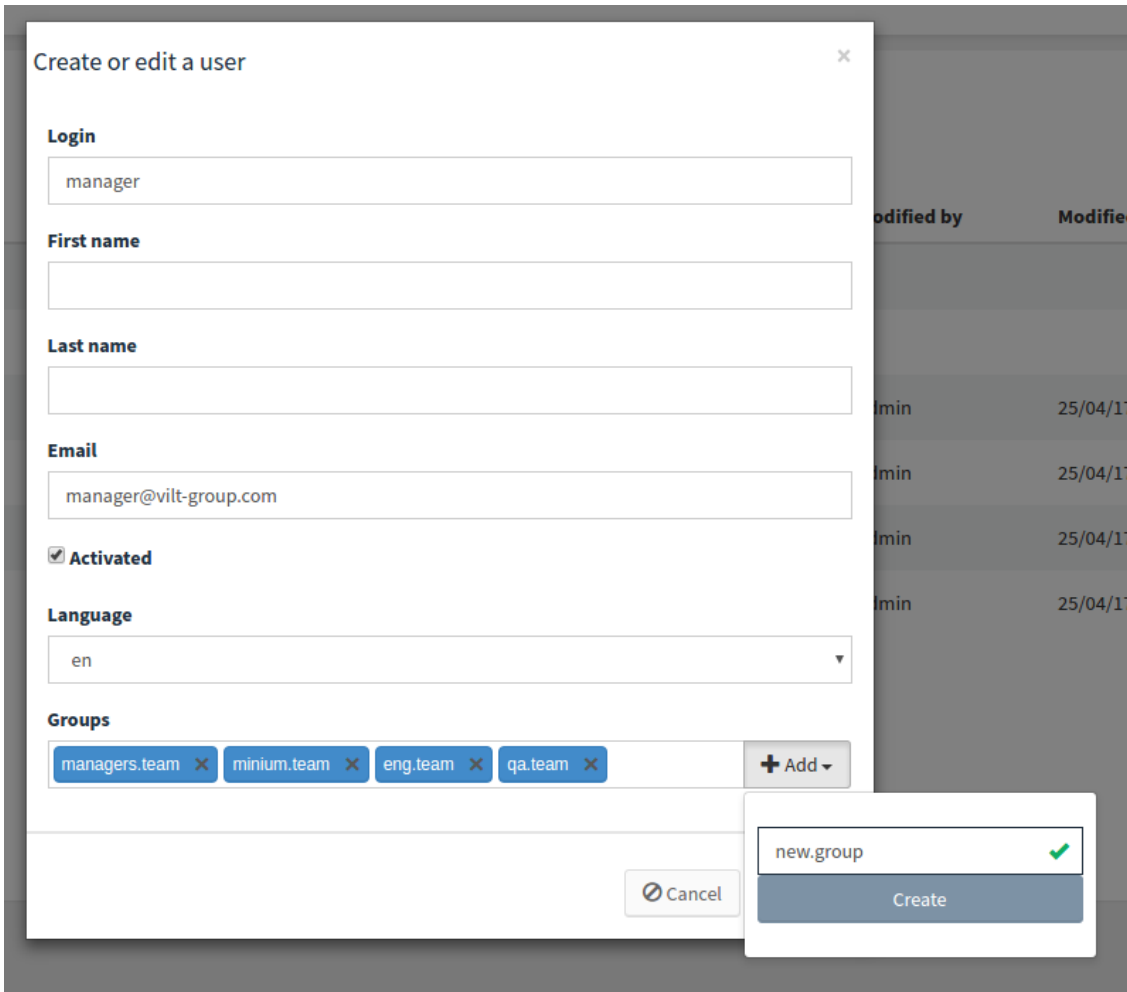
Note:

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

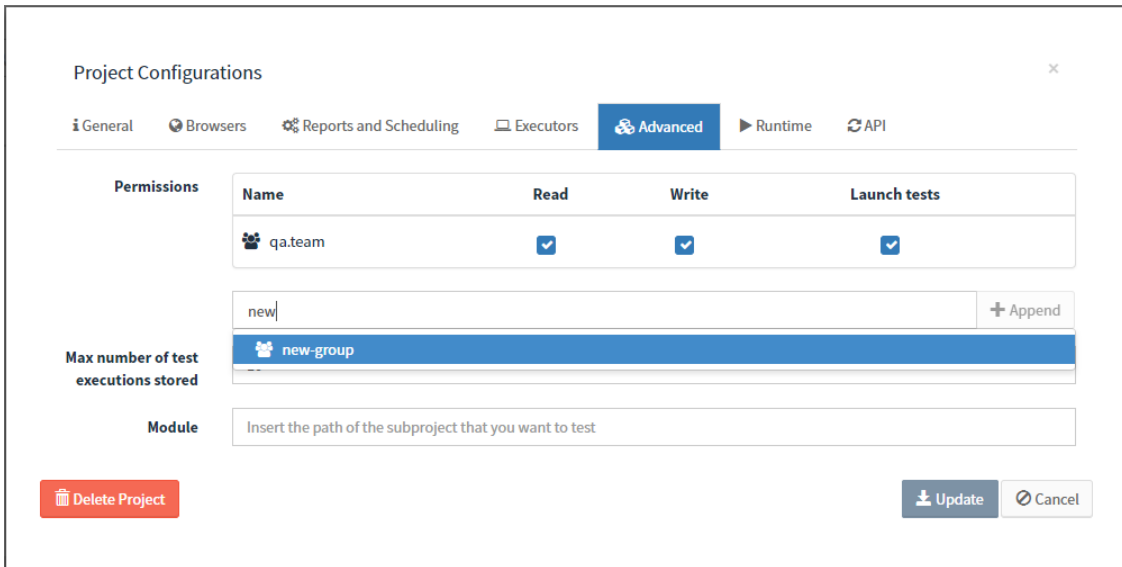


The screenshot shows the 'Users' management page. It includes a search bar, a '+ Register' button, and a table of users.

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

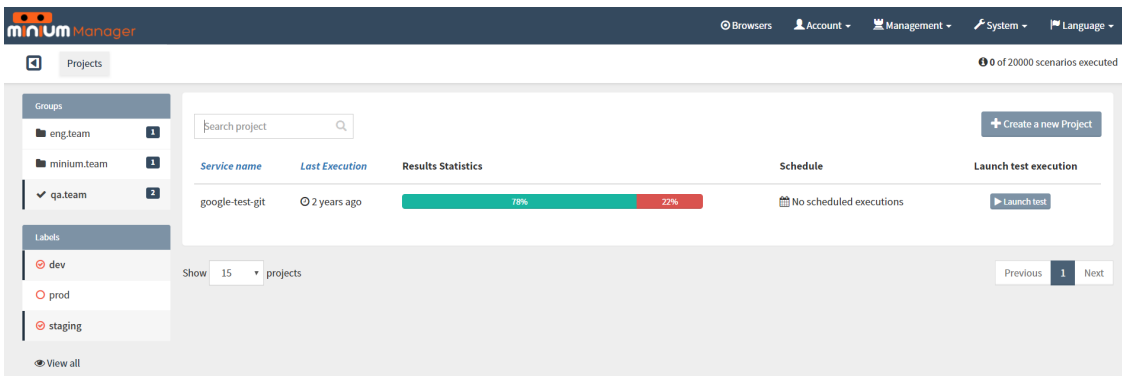


If desired add permissions to the group recently created in the project configuration. Below the permission table, start typing the name of the group created.



2.4.4. Organise Projects

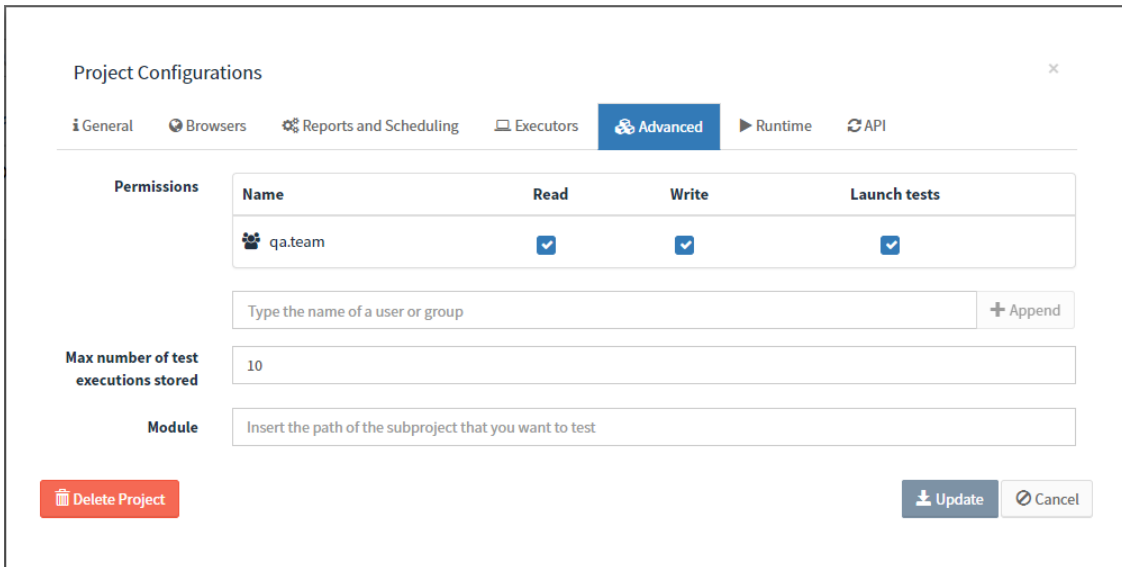
Using **groups** and **labels** to organise the projects selecting a group and labels from the sidebar.



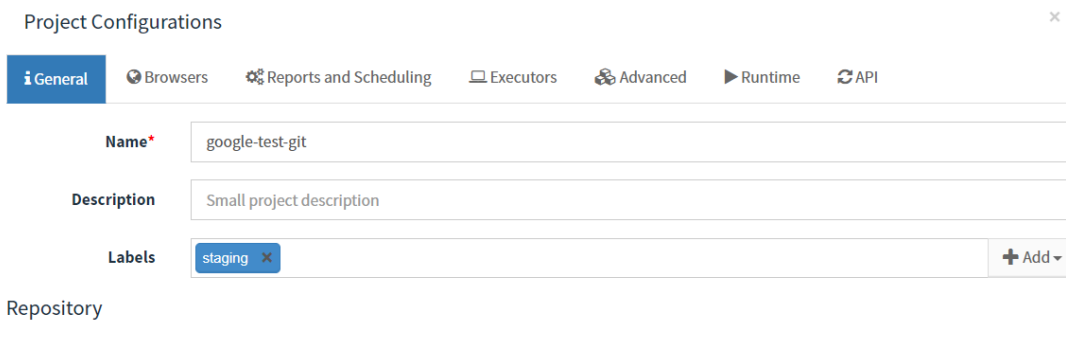
Select one **group** at a time, but select multiple **labels** and combine both.

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 this filters applied, exists one project shown in the project list.

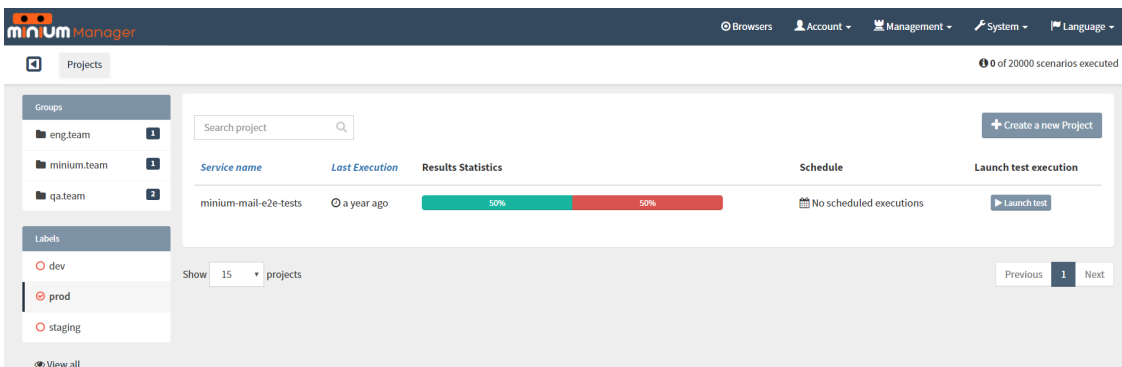
This means that the one project belong to the **group qa.team** (see it in the figure below).



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

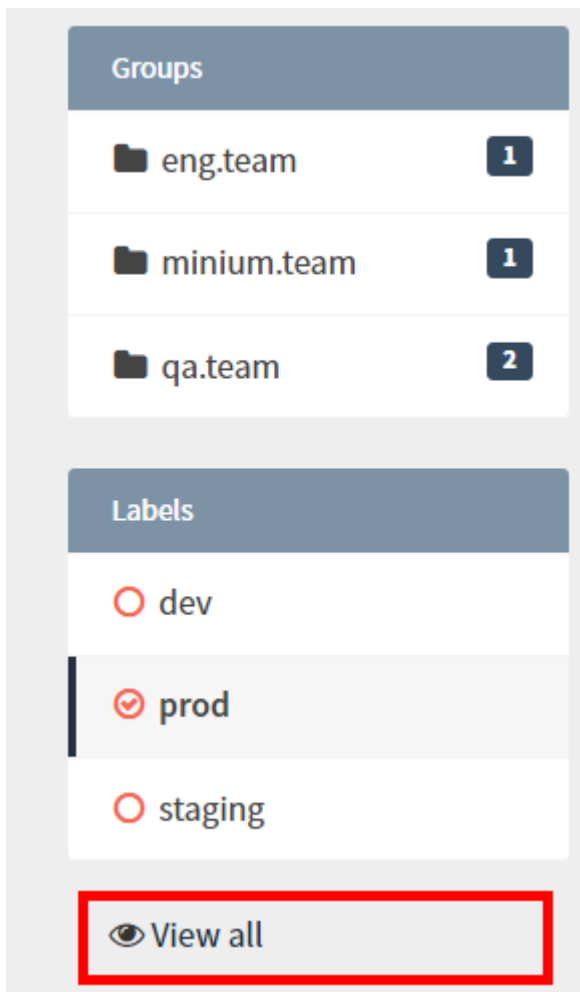


Select a **group** or **one or more label**, like shown in the figure below. In this specific case, are selected only the **label prod**, without selecting any group. It means that all the projects in the list have the **label prod** associated.



Clean project filters

By clicking on the button **View all**, it will remove all the selected filter.



2.5. Configuration profiles

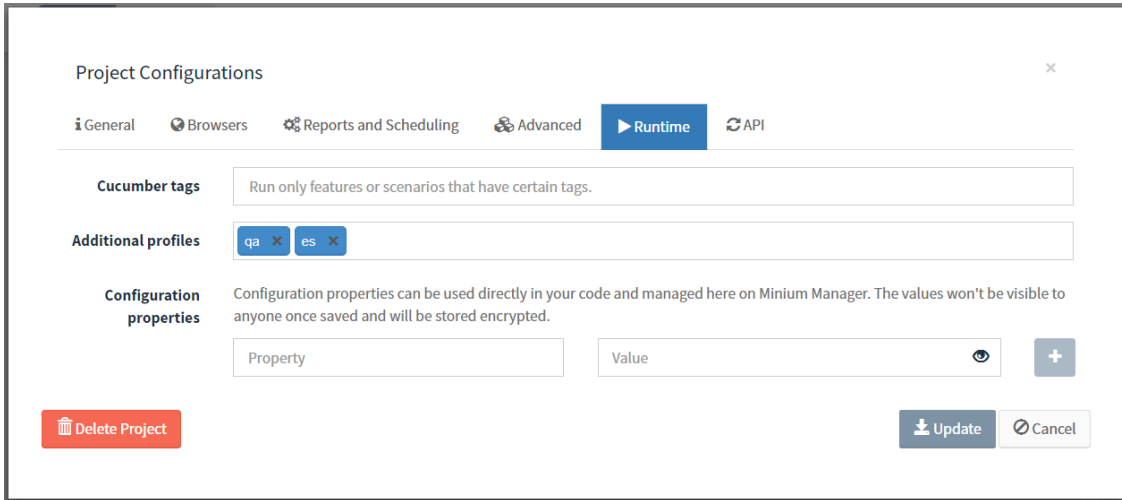
Configuration profiles can be defined on the `config/application.yml` of a project and used to load configuration properties that vary based on the environment being tested. Such configuration could be for example the URL of the application or the language:

```
# 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**:



2.6. Secret configuration properties

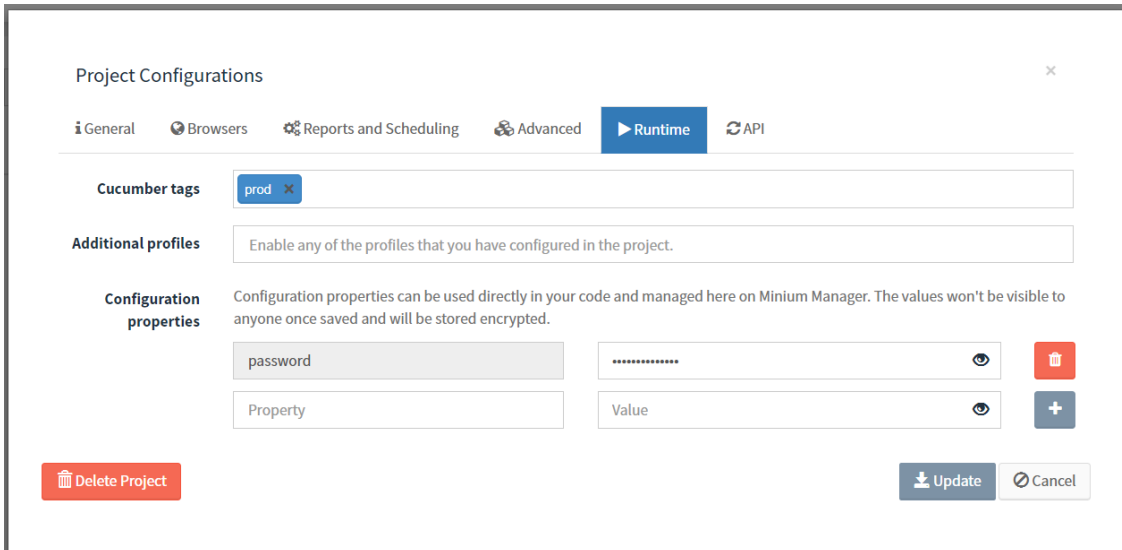
Sensitive configuration properties that cannot be included on the repository of the tests can be safely stored on Minium Manager. They can be used just like the configuration properties defined on the `config/application.yml` file of the project. As an example, consider that some tests need to be executed 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:



Project Configurations

General Browsers Reports and Scheduling Advanced **Runtime** API

Cucumber tags: prod

Additional profiles: Enable any of the profiles that you have configured in the project.

Configuration properties: 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.

password	👁	🗑
Property	Value	👁	+


Delete Project Update Cancel

2.7. Delete a project

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

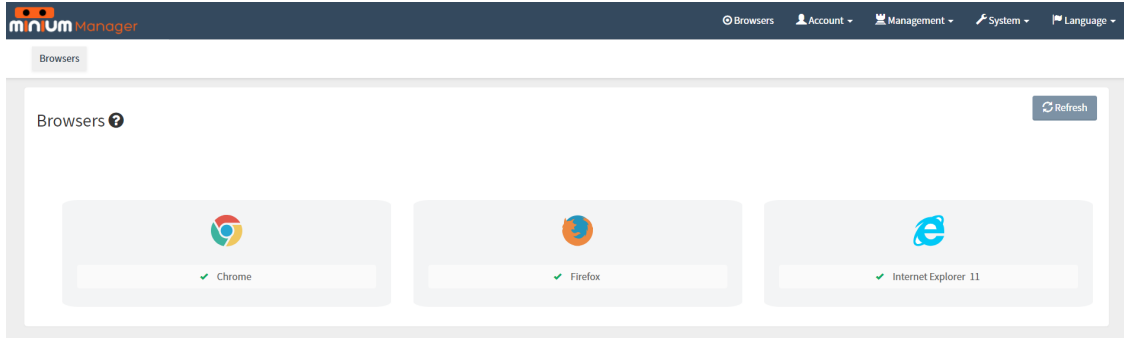
Sh

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



3. Check the available browsers

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

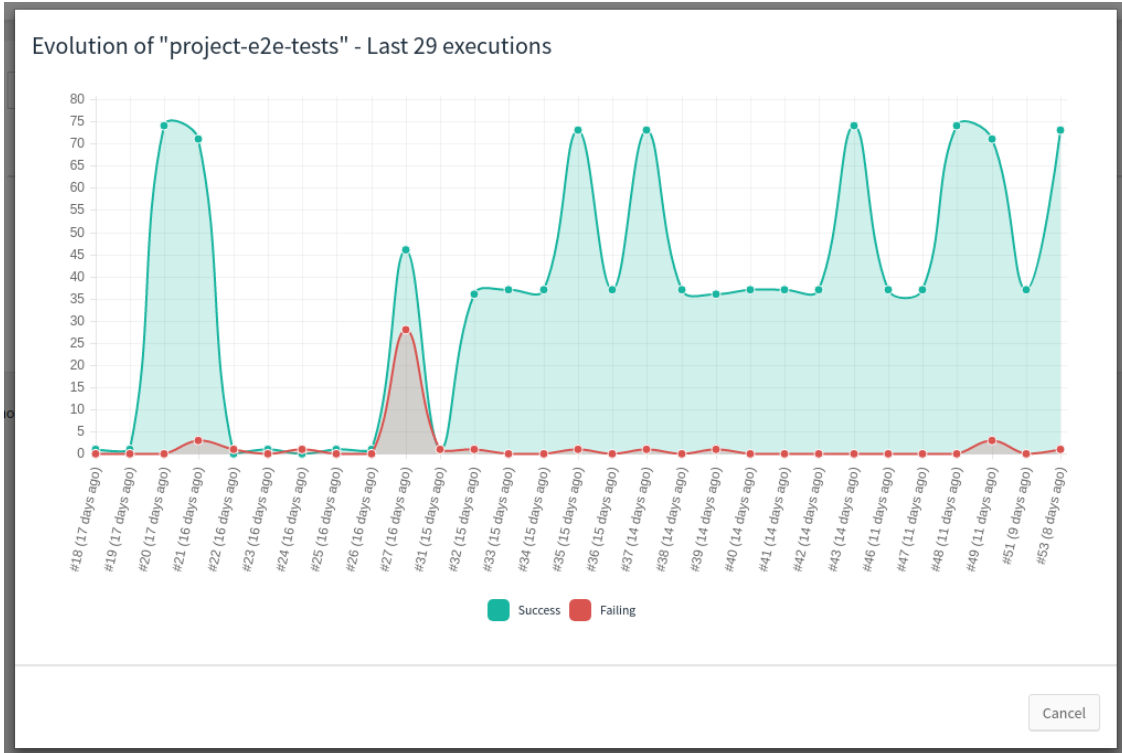


In the figure above, we only have one provider available, but we can have two or more providers configured (e.g. "BrowserStack" and/or "Selenium Grid Extras"). The provider configured has the browsers Chrome, Firefox and Internet Explorer available.

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.



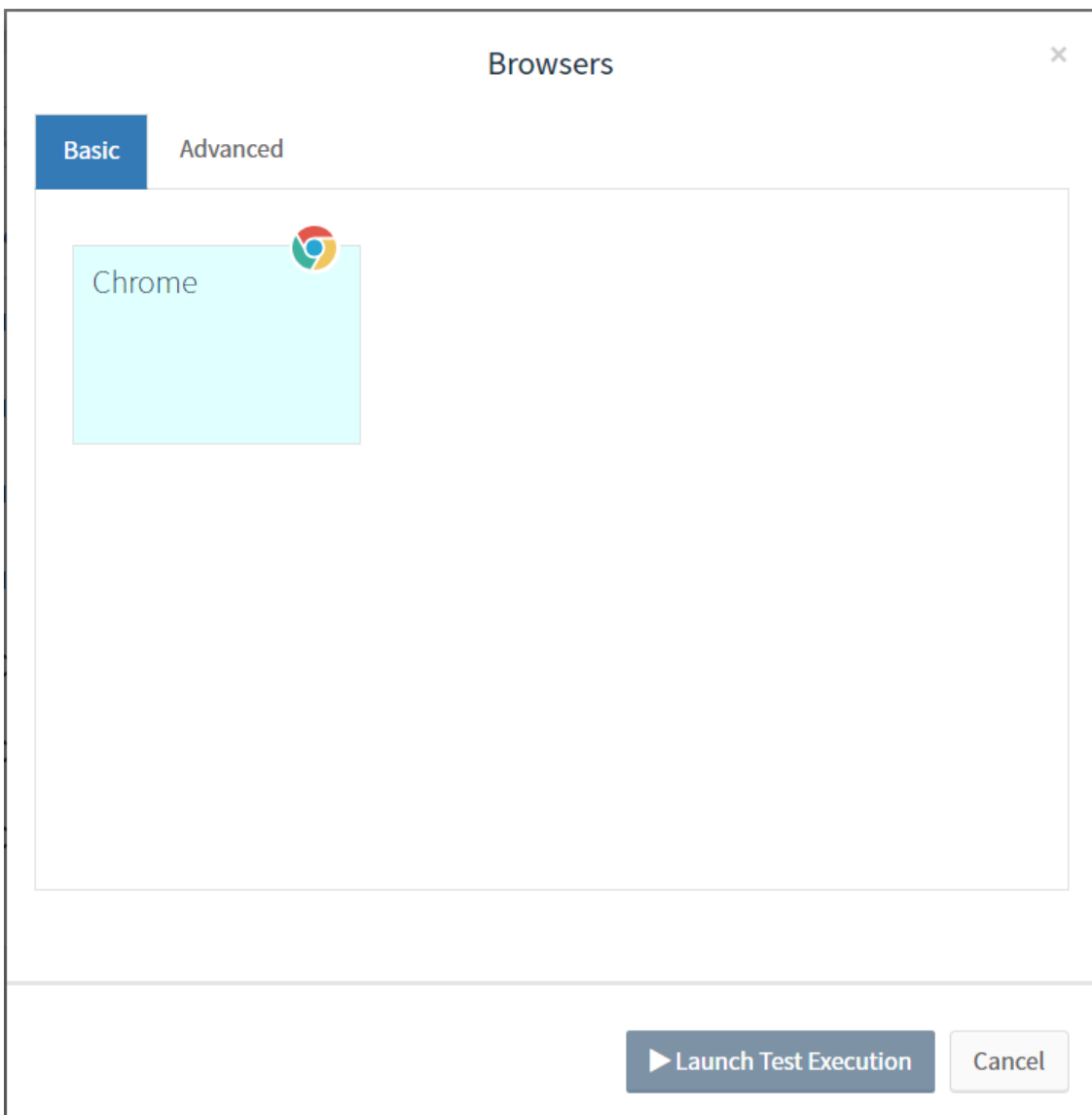
5. Test executions

5.1. Launch test executions

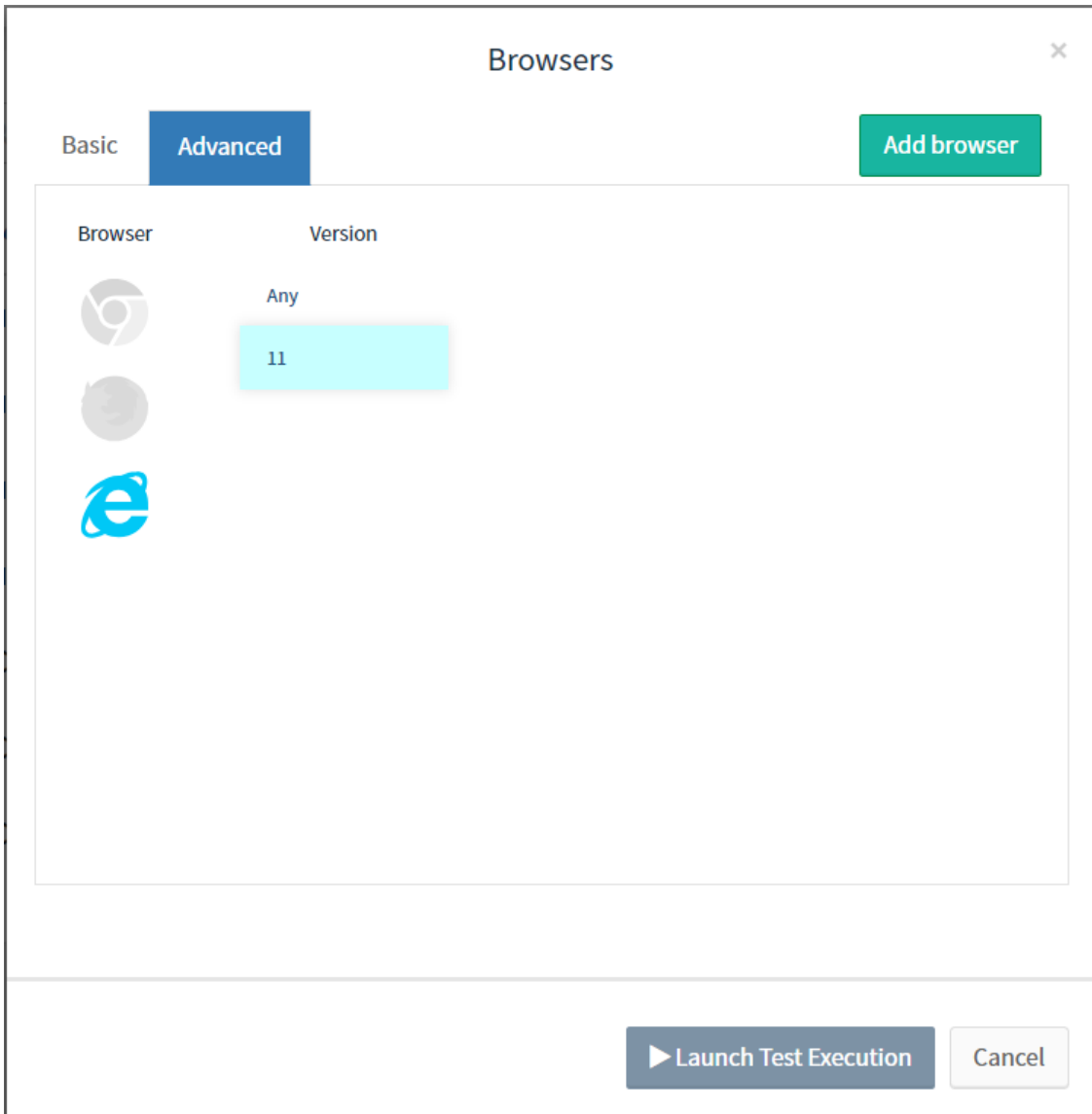
A test execution will execute the tests present in a project on one or more different browsers. There are alternative ways for launching test executions.

5.1.1. Launch a test execution manually

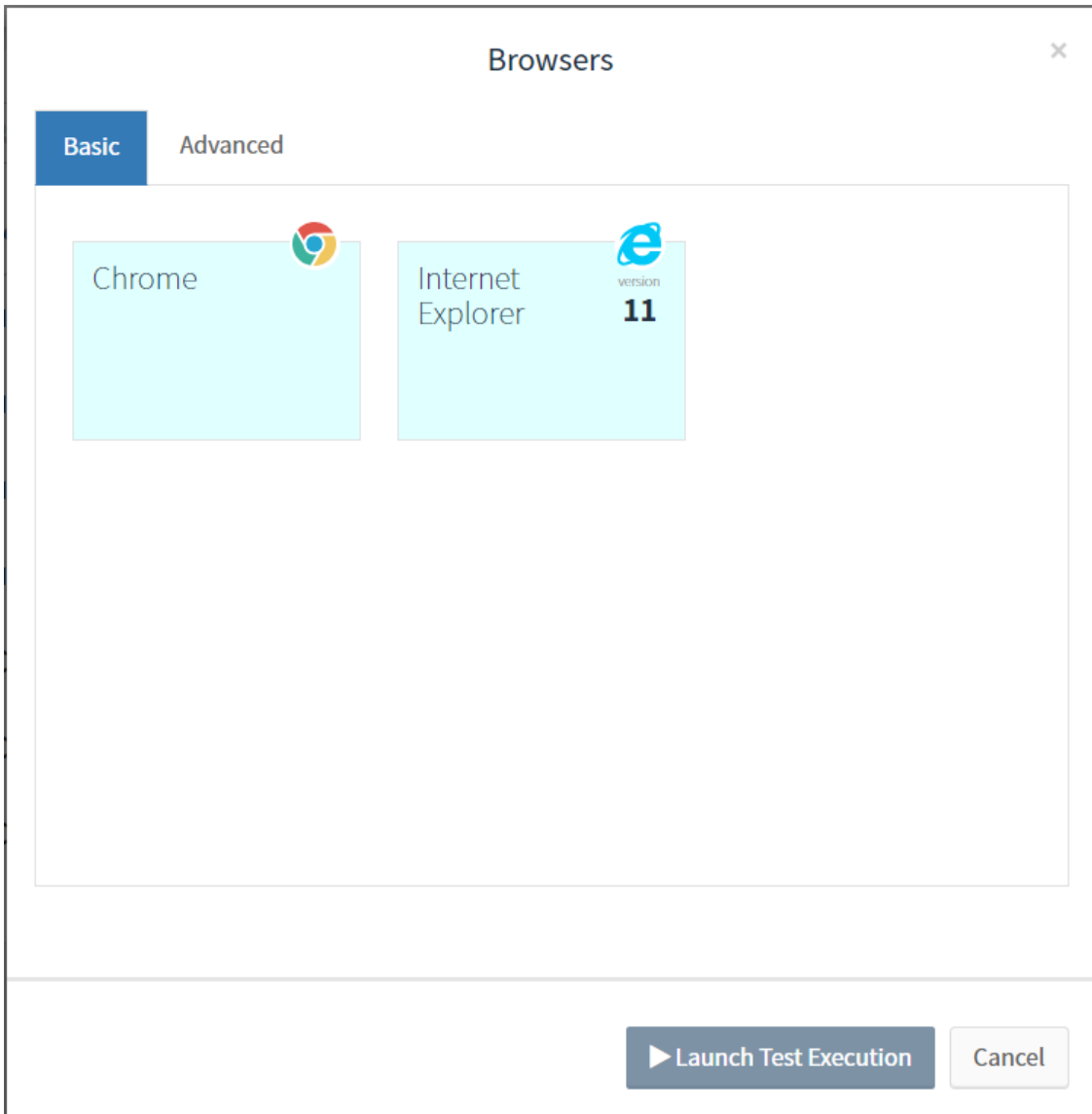
On the homepage or the page of a project, click on button `Launch test`, which opens a modal for selecting the browsers.



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`.

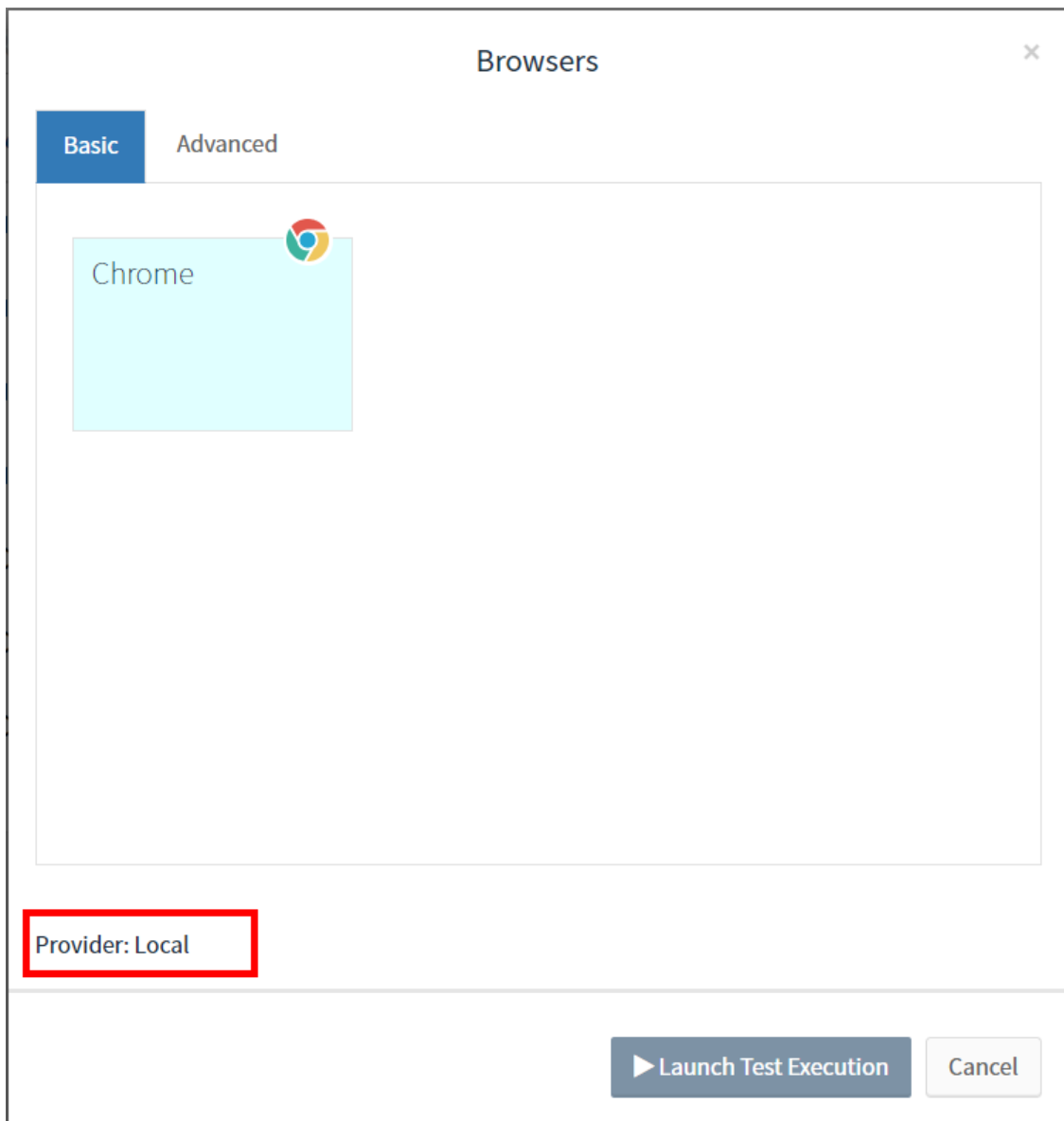
Note:



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

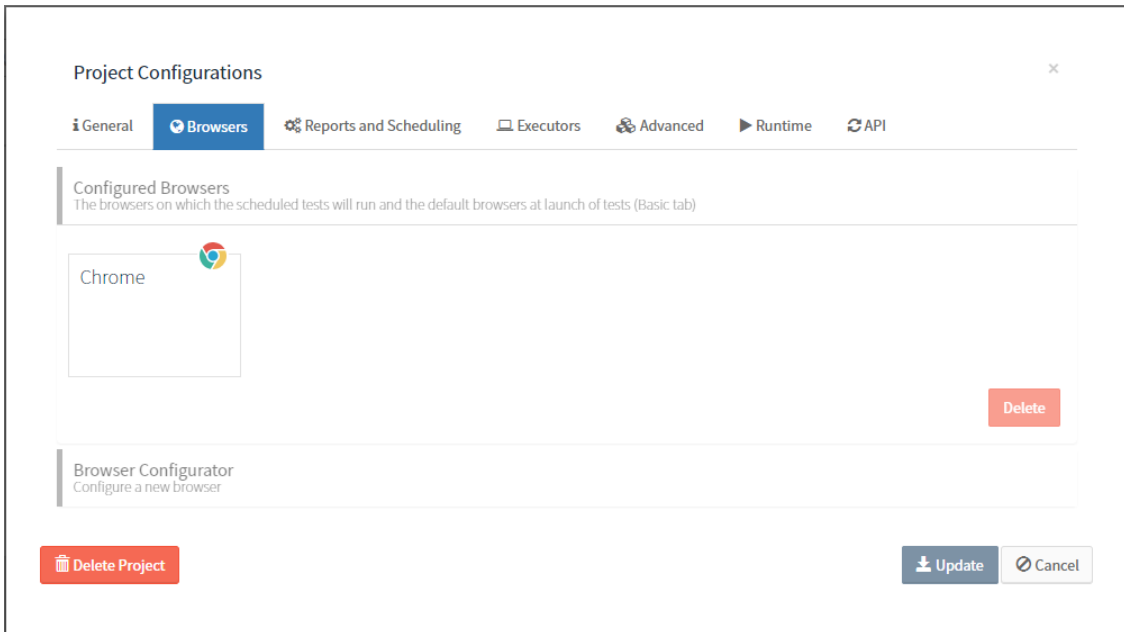
Multiple Providers

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

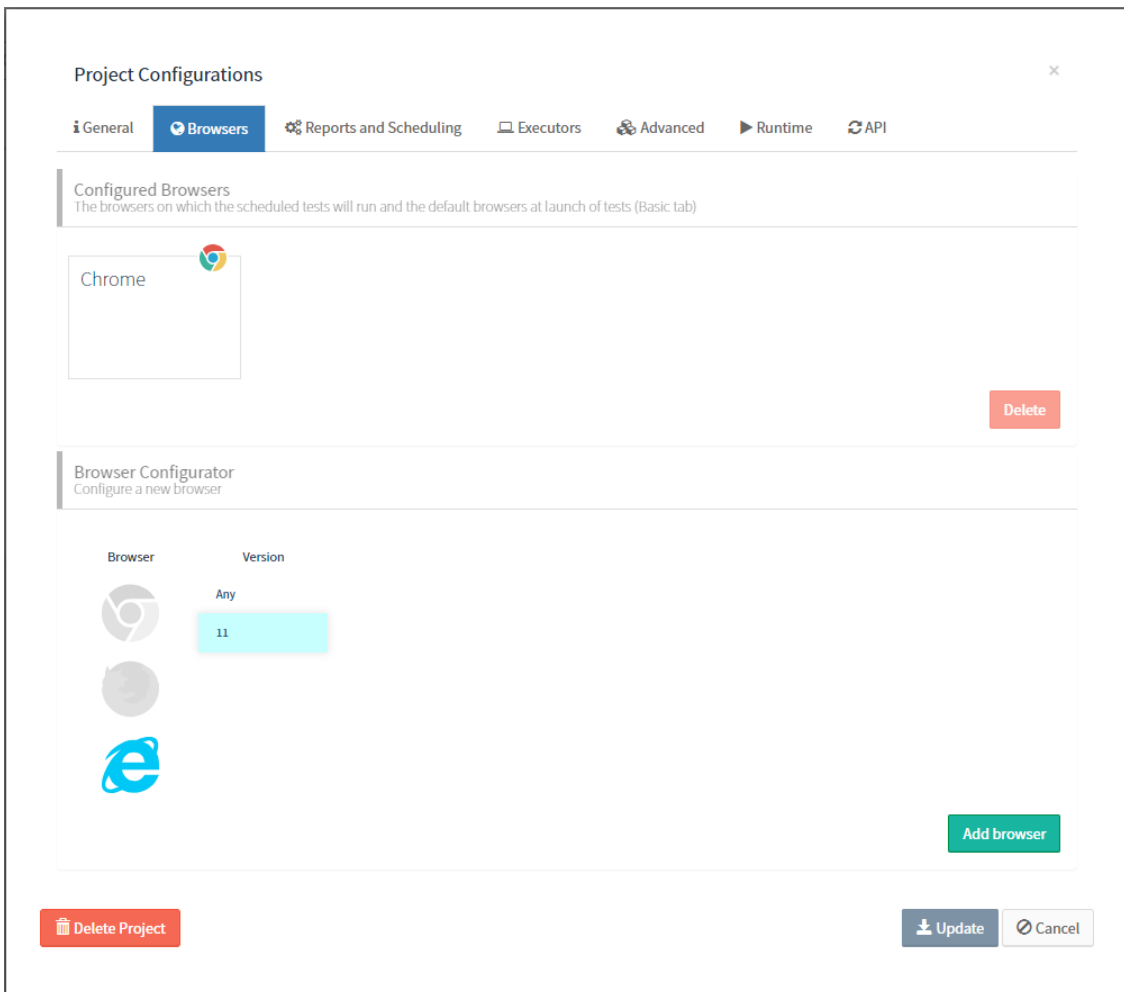


5.1.2. Schedule test executions

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

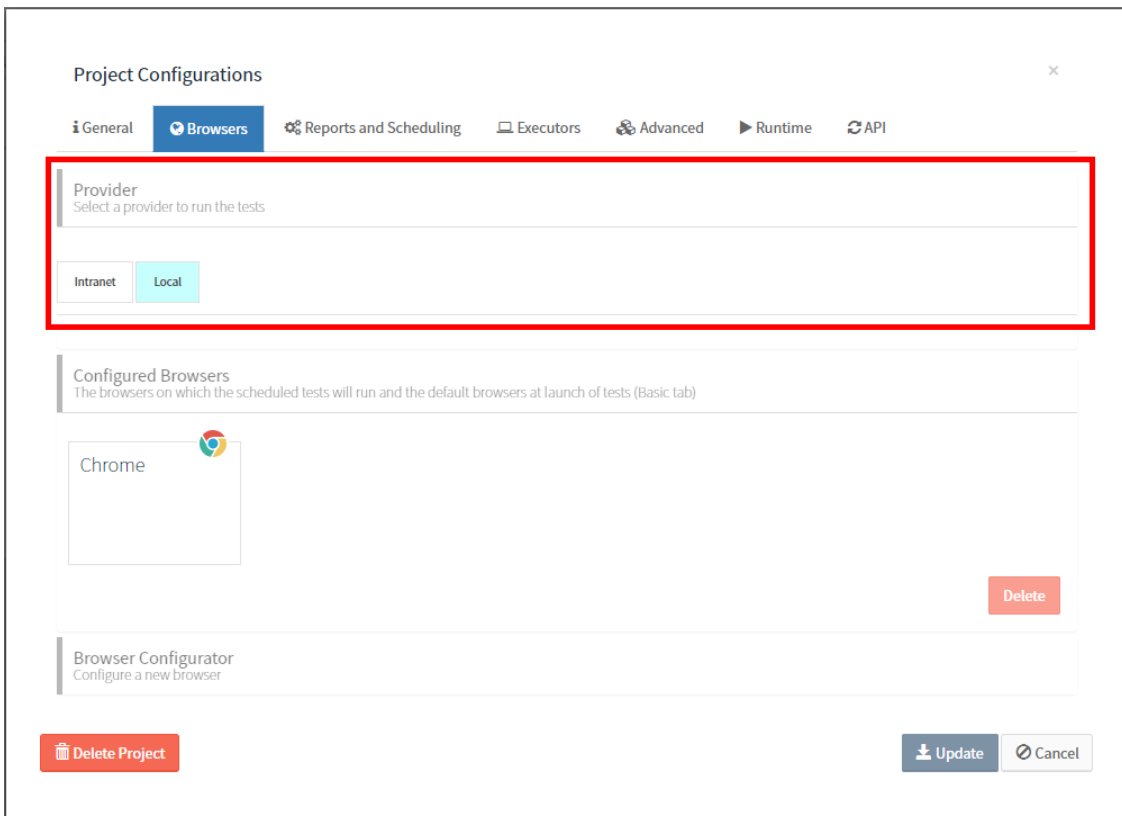
Never run
 Every Day
 Every Week
 Every Month
 Every Day at midnight
 Custom

Scheduler Custom Configurations

Every: at :

Multiple Providers

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



Project Configurations

General
 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 Delete

Browser Configurator
Configure a new browser

Delete Project
Update
Cancel

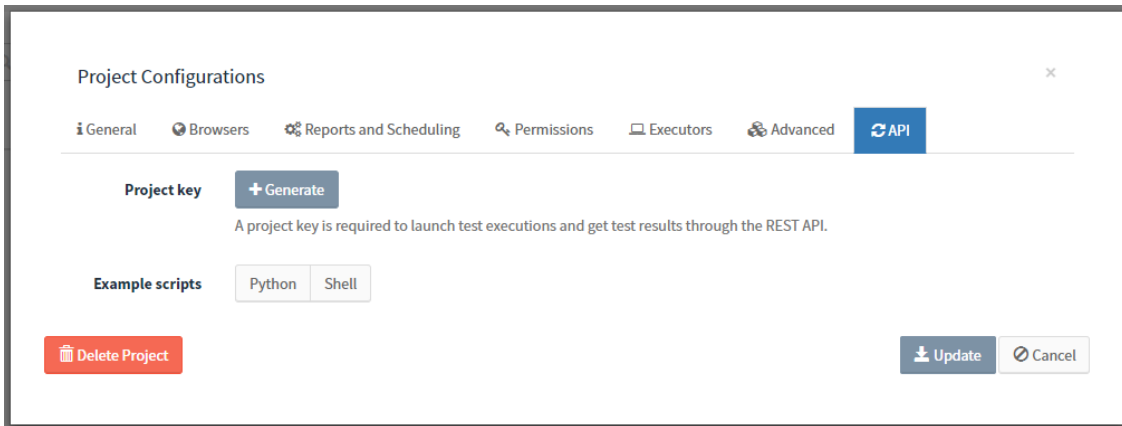


Note:

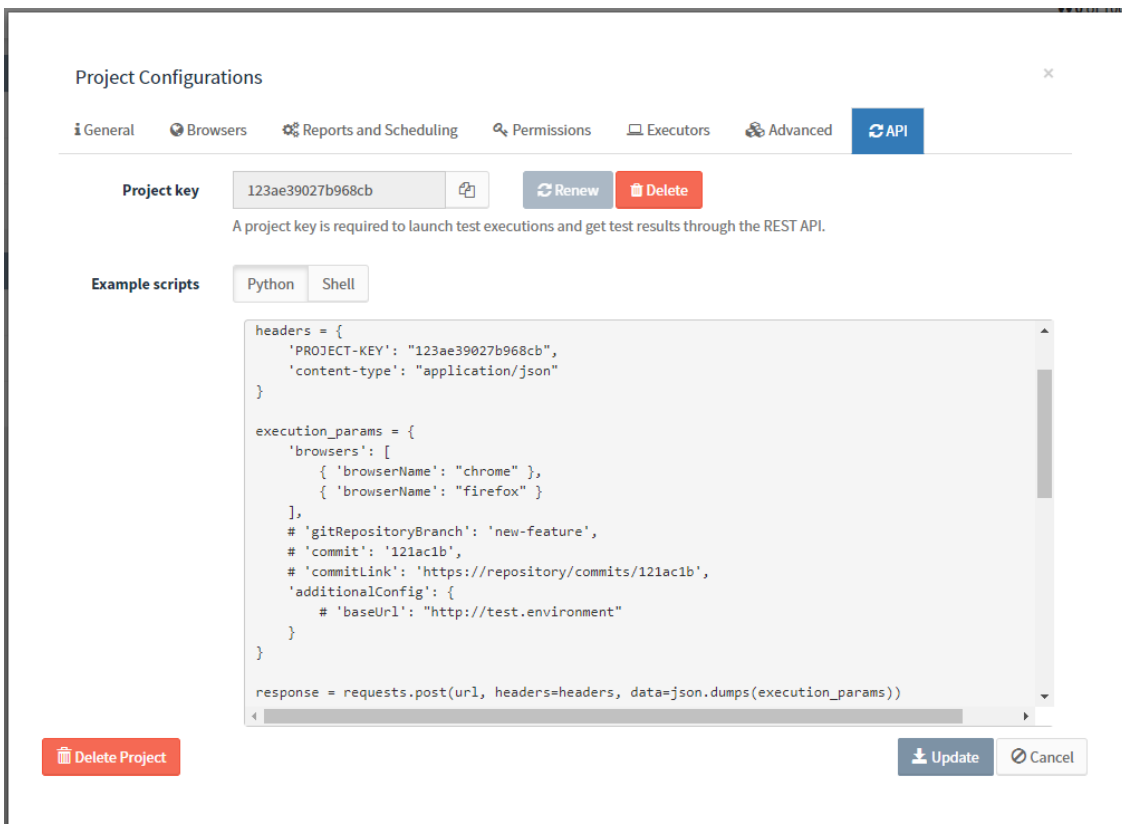
This configuration is not available at the Monitoring projects.

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 keys 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:

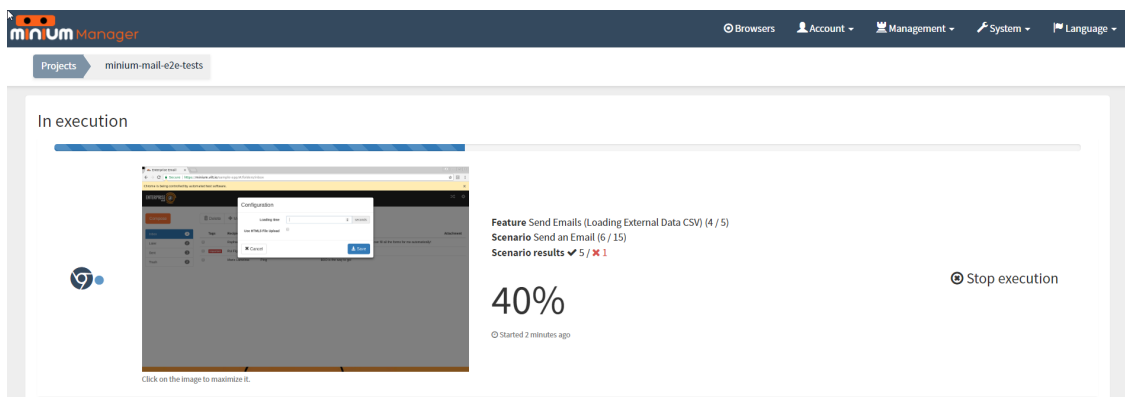
```
{
  "state": "RUNNING",
  "browser": "chrome",
  "progressInPercentage": 18,
  "failingScenariosCount": 1,
  "passingScenariosCount": 5,
  "feature": "Login",
  "scenario": "Successful login"
}
```

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
    }
  ]
}
```

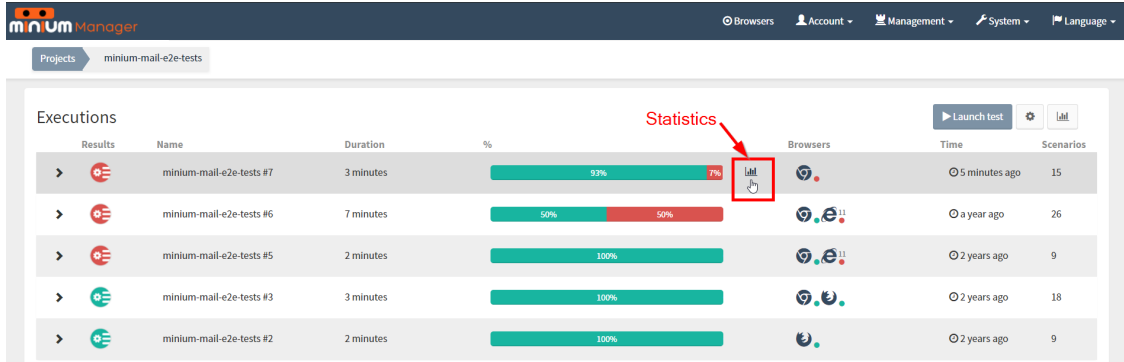
5.2. Follow the progress of a test execution

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

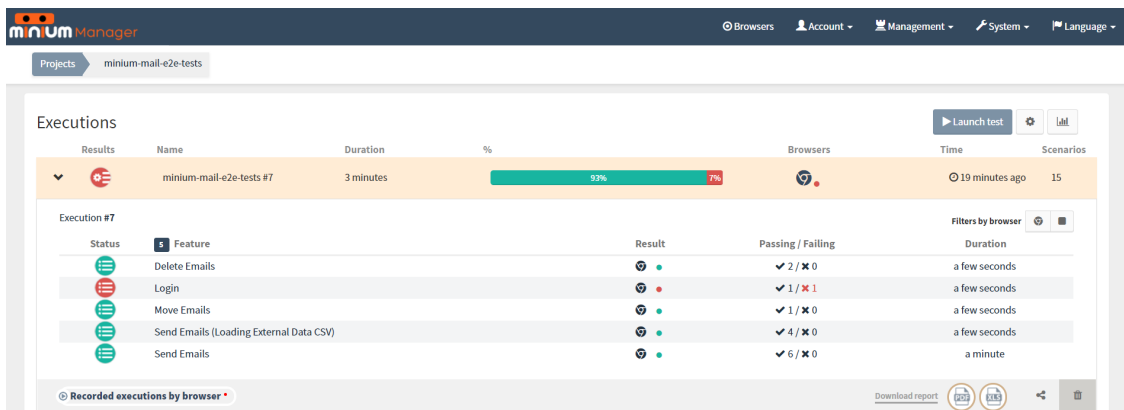
In a project page, all the launched executions are showed:



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:

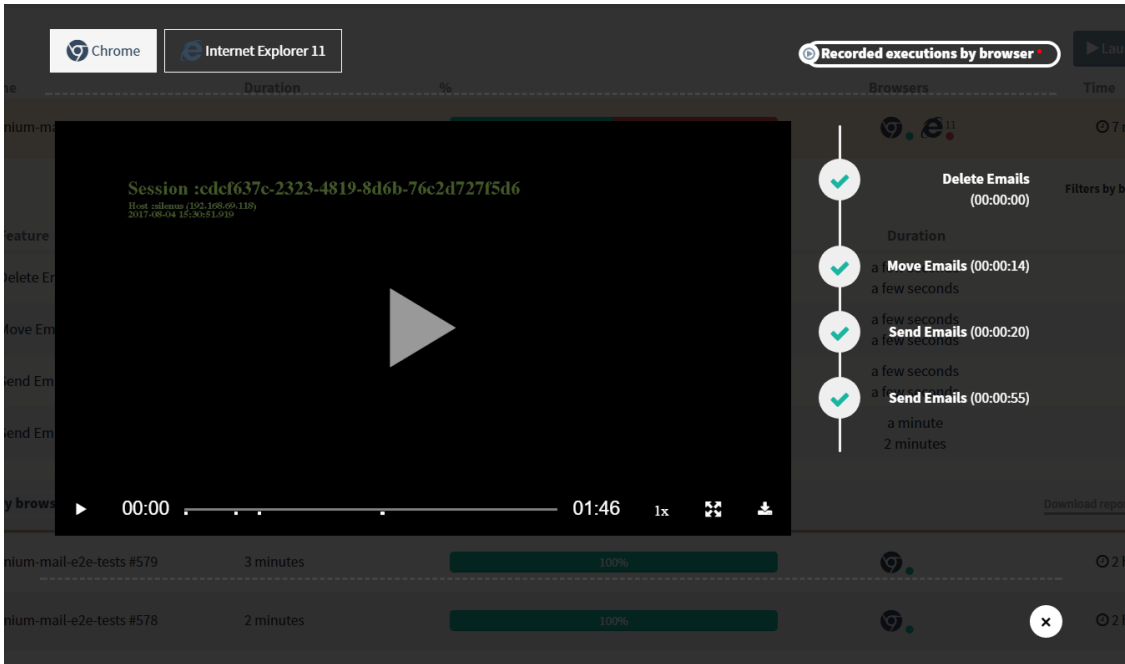


For the execution expanded it can be seen:

- 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:

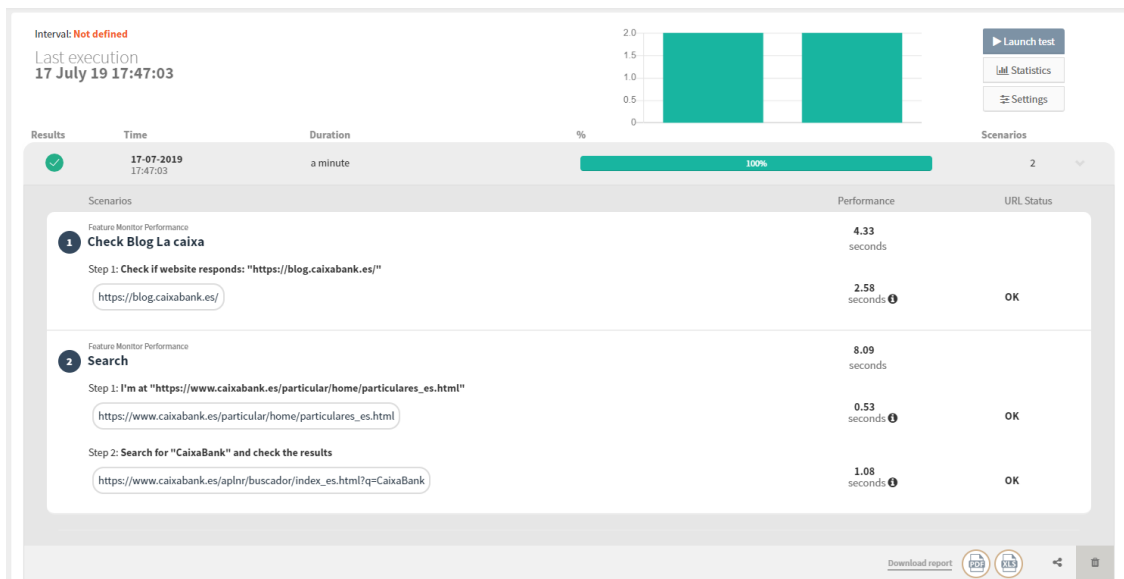


For the videos (at execution) it can be seen:

- 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:

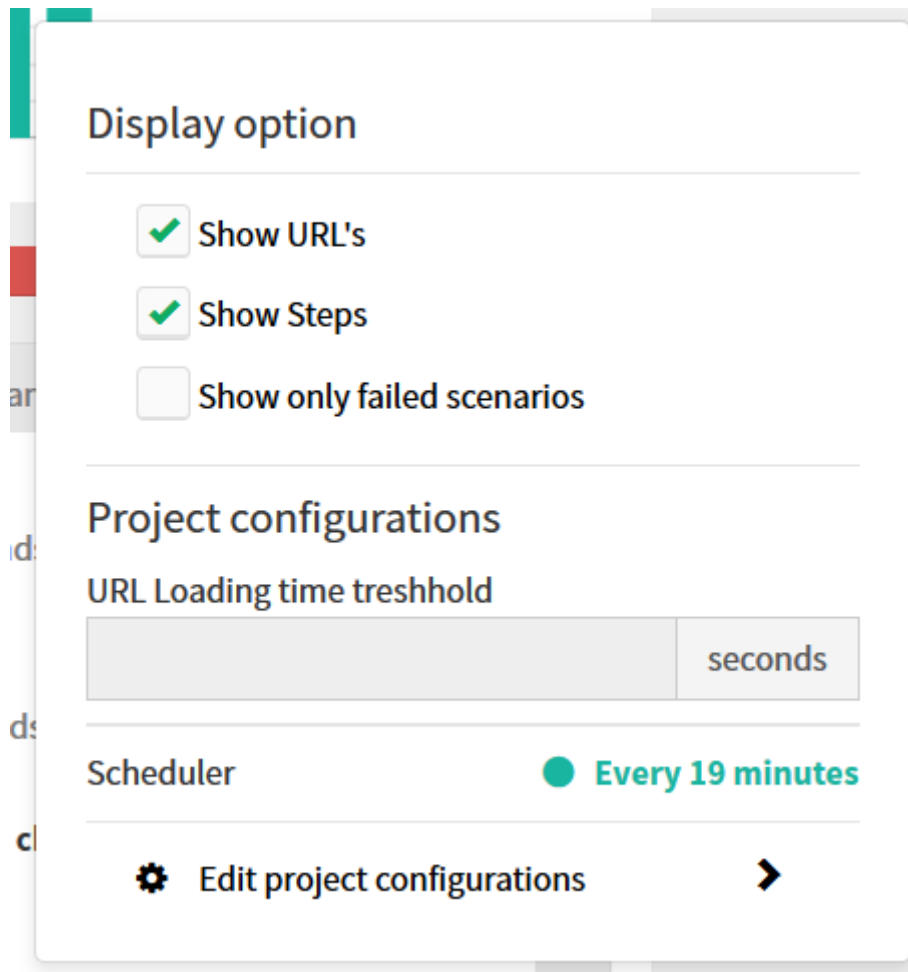


At execution page, we can see the time of each URL took to load. Also we can see the URL status and the time that took to execute the scenario.

Display options

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

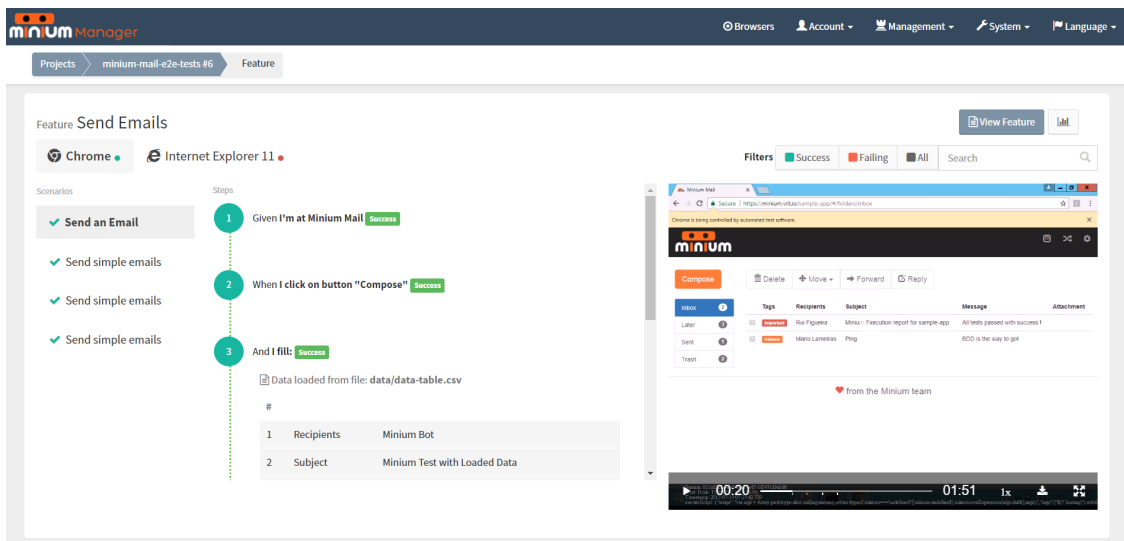
To open the `Display options` click at `Settings`:



The display options allow us to:

- Show/Hide the URL's
- Show/Hide the steps
- Show only the sailed 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 an 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)

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

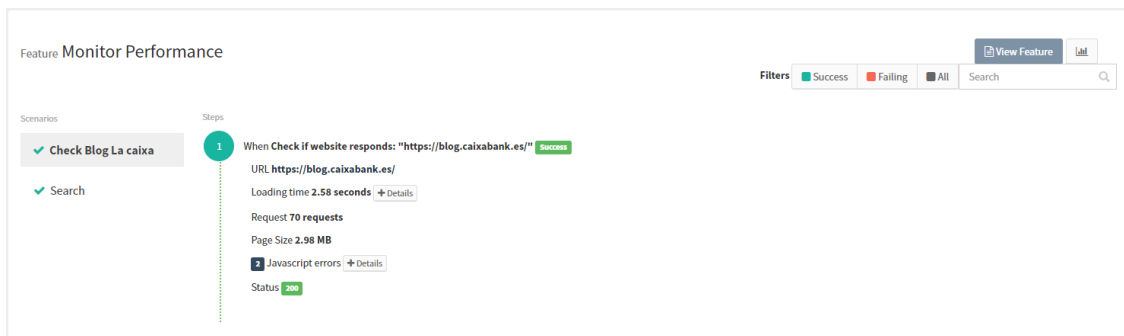


When expands a scenario see all the steps and results. Is easily switch to the results of another browser. More functionalities:

- **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:



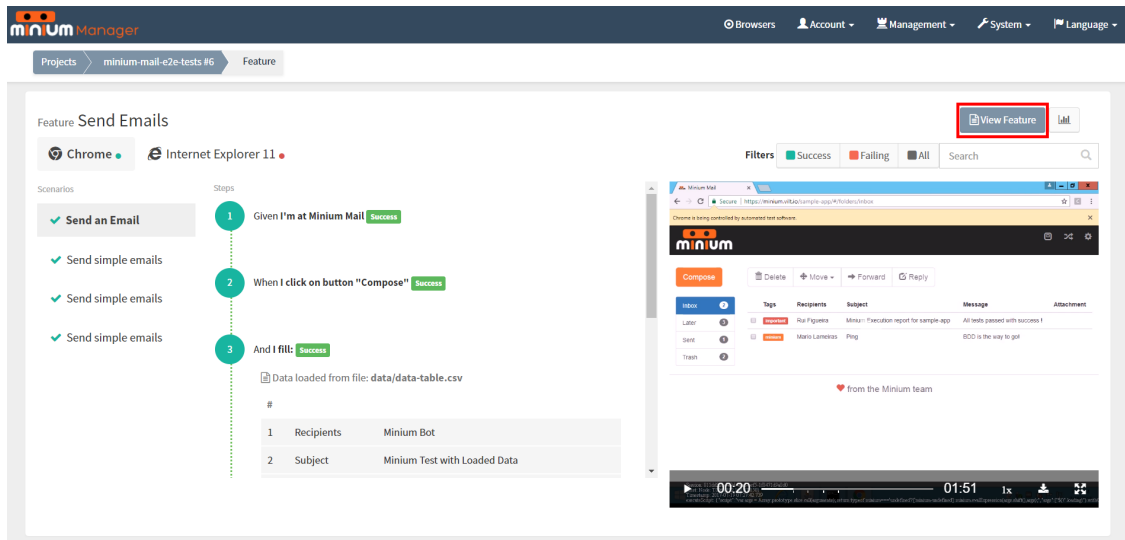
At the feature page, we can see more details about the page load performance:

- 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

6.1. Feature overview

Overview of all the features, by clicking on the button 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:

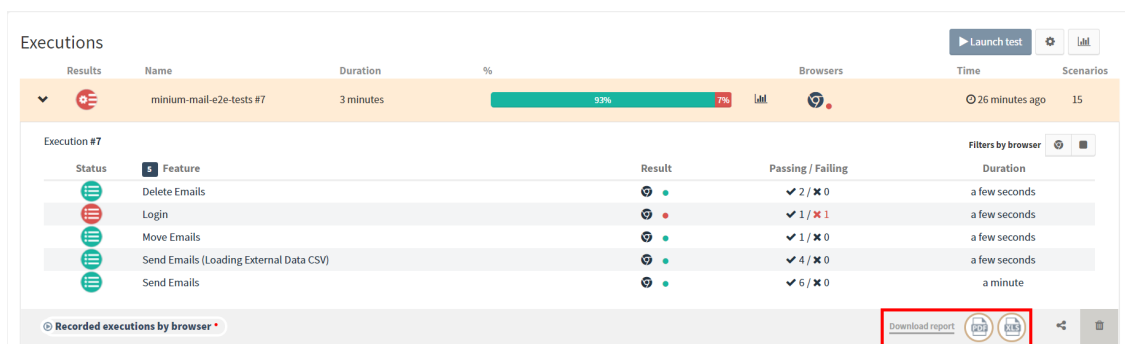
Emails

Recipients

Send only when there are test failures

Report formats

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



The screenshot shows the 'Executions' section of the Minium Manager interface. A test execution named 'minium-mail-e2e-tests #7' is shown with a 93% completion rate. Below the execution summary, a table lists the test results for 'Execution #7'. At the bottom right of the table, a 'Download report' button is highlighted with a red box, along with PDF and XLSX icons.

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

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

Previous Next

Project(s) selected:

Total selected: 0

None

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

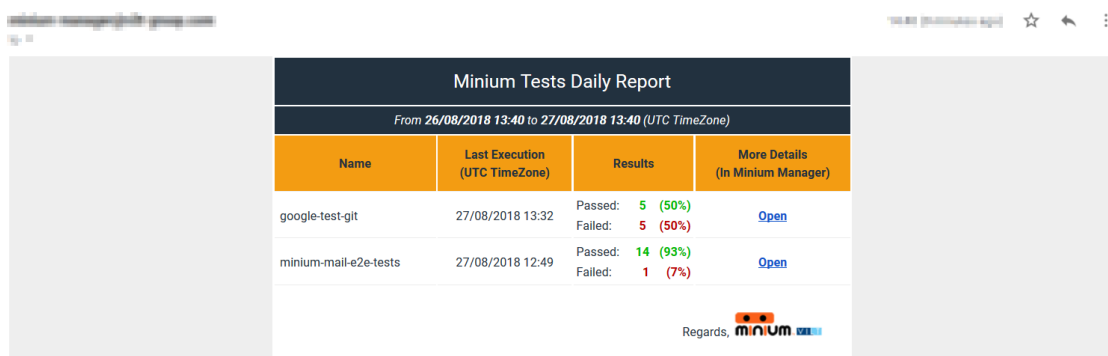
Scheduling

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)



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%)	Open
minium-mail-e2e-tests	27/08/2018 12:49	Passed: 14 (93%) Failed: 1 (7%)	Open

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](#)

7. Roles

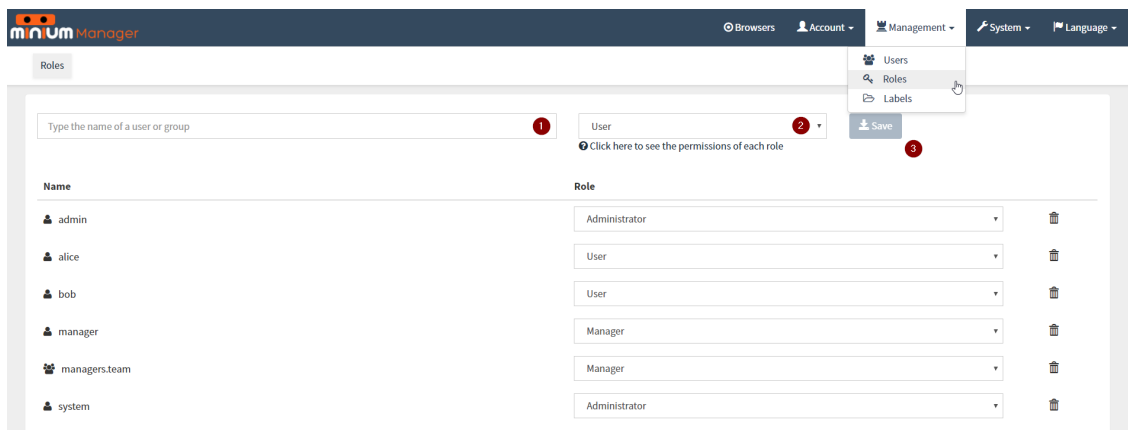
Roles allows grant different privileges to the users and the groups. The privileges associated with each role are described in the table below:

	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:

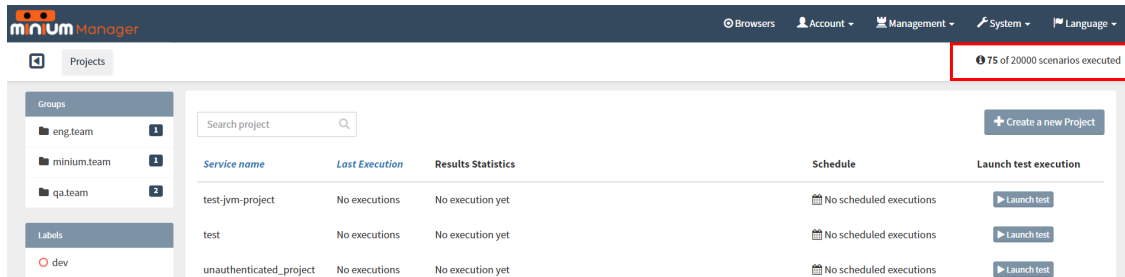
1. Introduce the name of the user or group in the search bar.
2. Select a role.
3. Click on **Save** button.



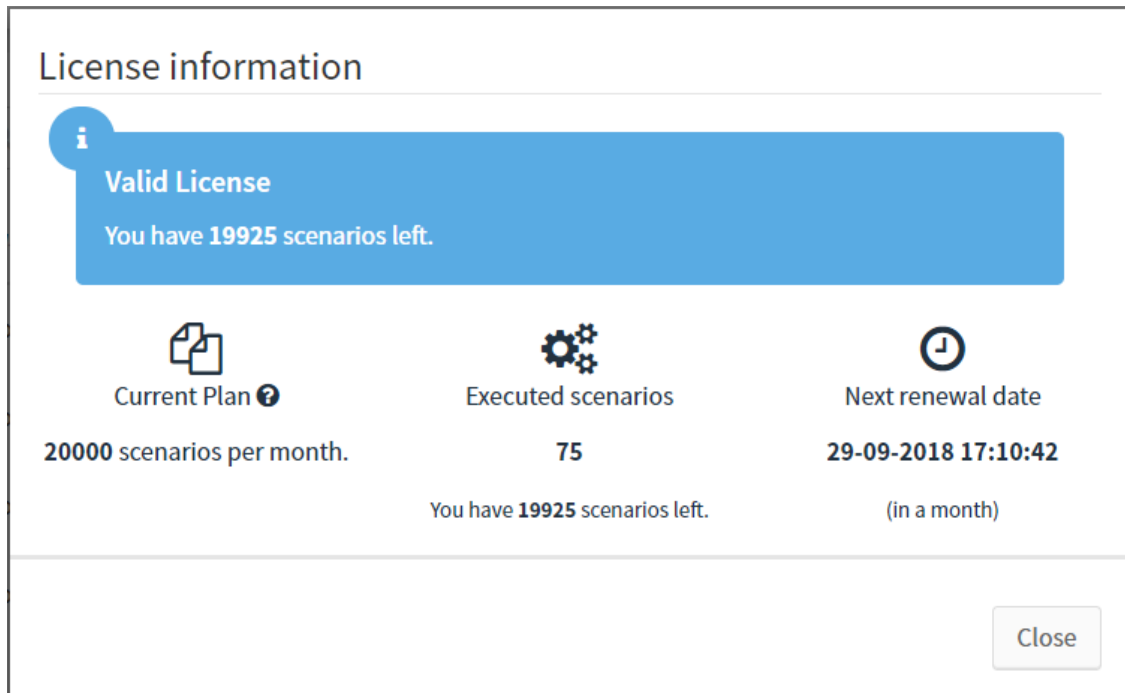
The screenshot displays the 'Roles' management page in the VILT Minium Manager. At the top, there is a navigation bar with 'Management' selected. Below it, a search bar is labeled 'Type the name of a user or group' (1). To the right of the search bar is a dropdown menu for selecting a role (2), currently showing 'User'. Below the search bar is a list of users and groups: admin, alice, bob, manager, managers.team, and system. To the right of this list is a table where each user's current role is displayed in a dropdown menu, and a trash icon is visible next to each entry. A 'Save' button (3) is located at the bottom right of the interface.

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



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



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

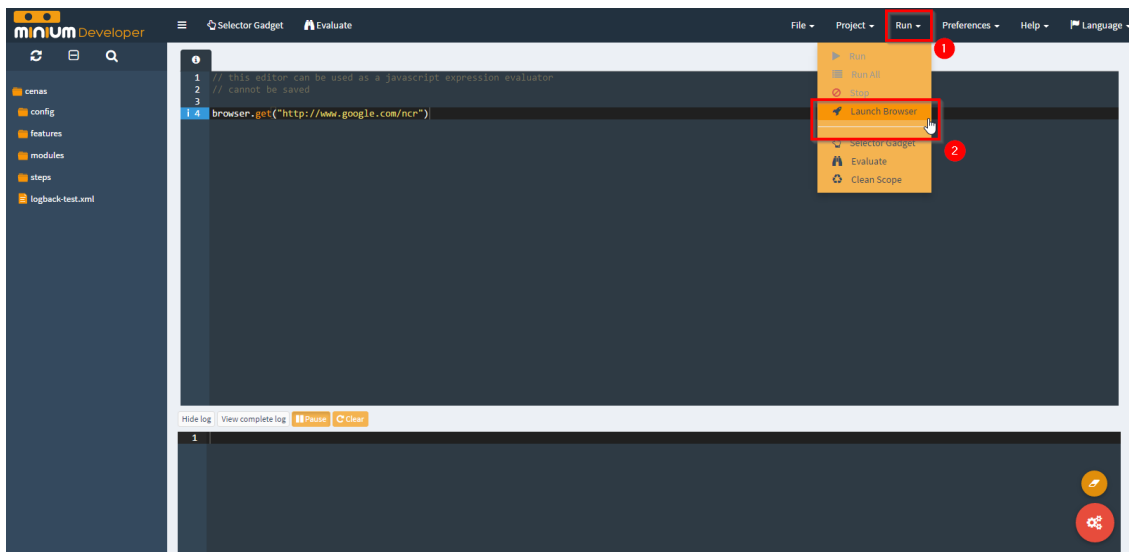
Launch **Minium Recorder** through **Minium Developer**.

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

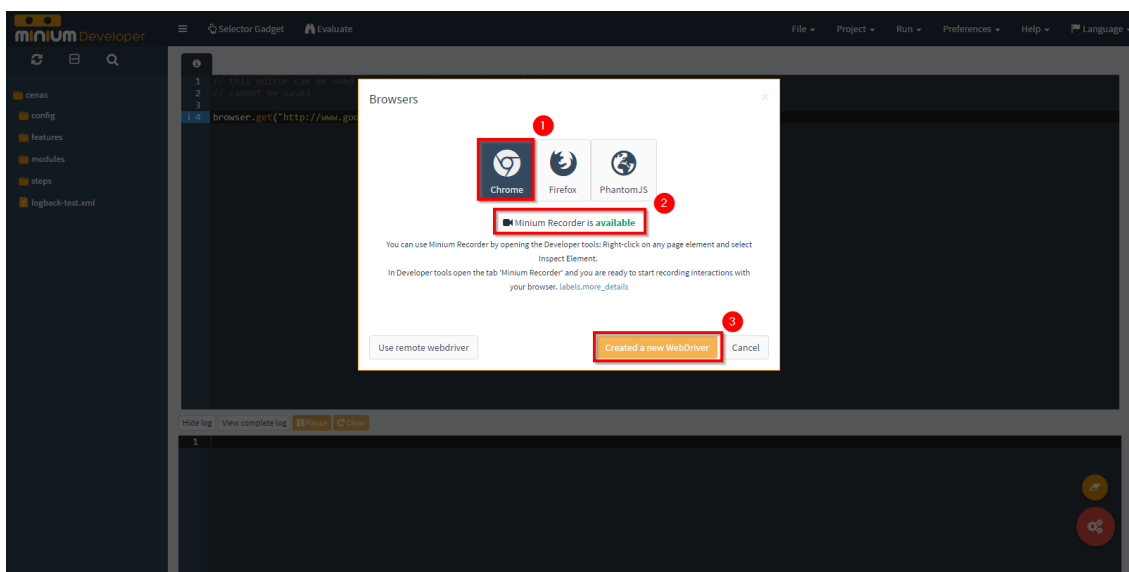
```
browser.get("http://www.google.com/ncr")
```

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:

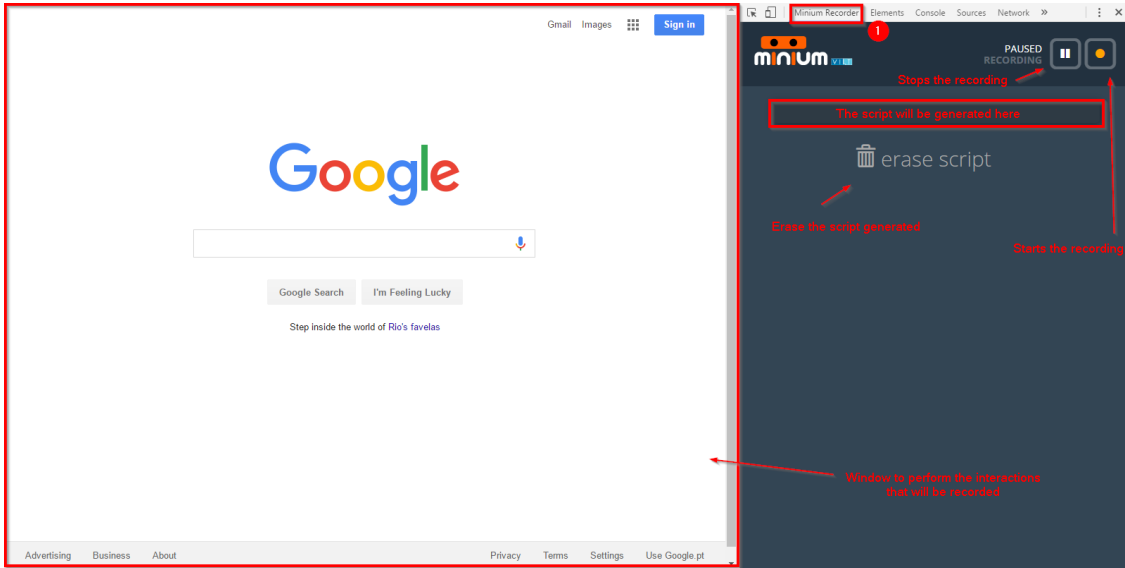


Select the **Chrome browser (1)**, check if **Minium Recorder is available (2)** and click **Created a new WebDriver (3)**:



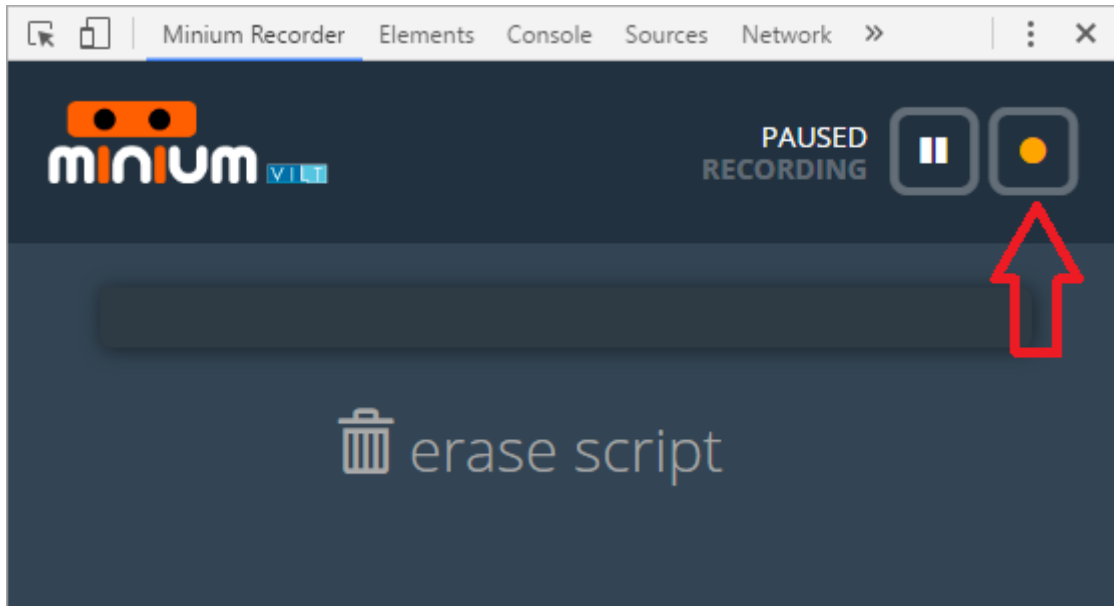
A new instance of Chrome will be launched with the Minium Recorder extension installed. In the instance of Chrome, open the `Developer tools` by using the (F12) shortcut (use another [shortcut](#)) or select `More Tools > Developer tools`.

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



10. Start recording

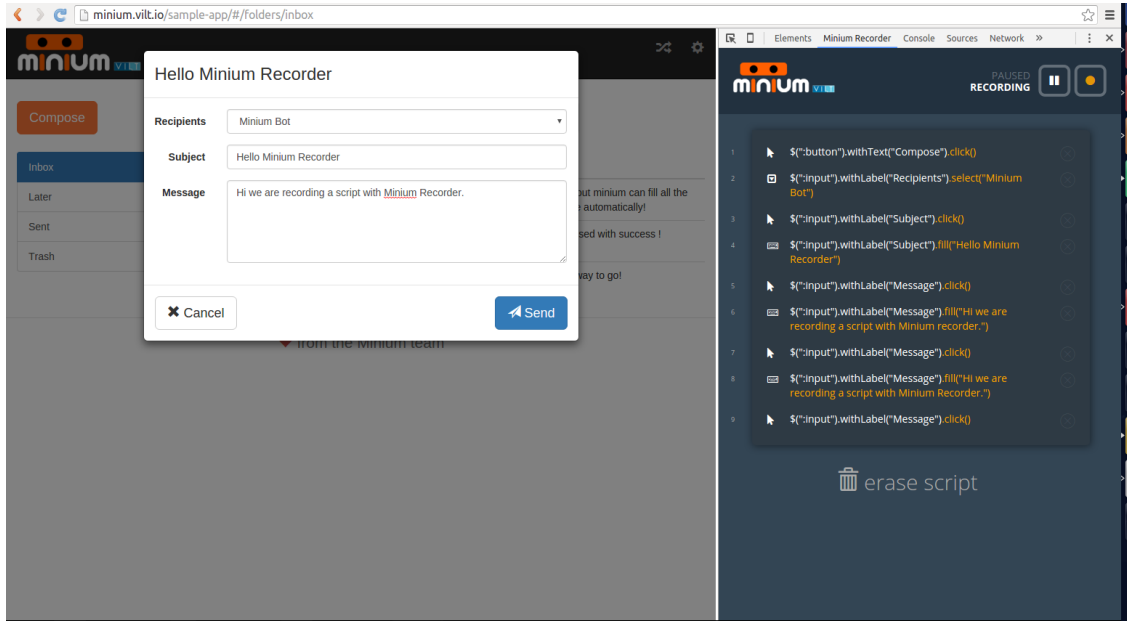
To start recording, click on the following button:



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.



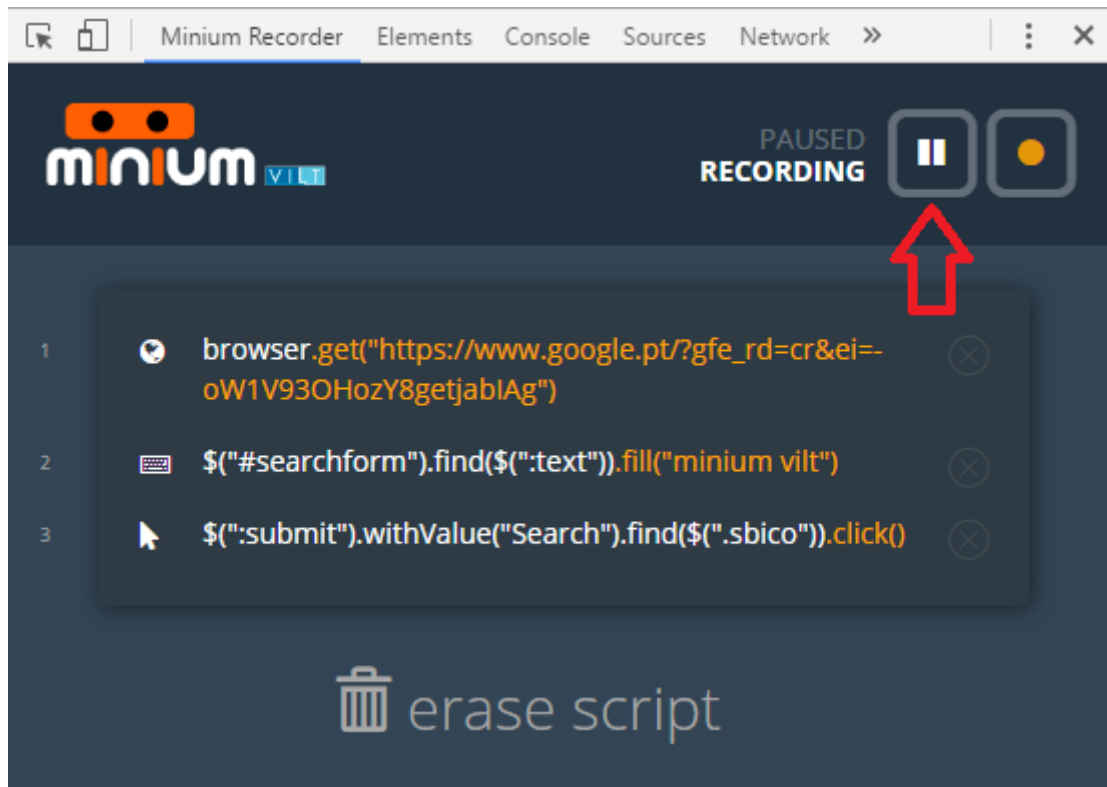
To try other expressions, click on the expression and a list of alternative expressions will be displayed:



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`:

