

# Make Your Selenium Suite Faster and Reliable: Test Setup with REST APIs

---

Sargis Sargsyan

# Main Topics

---



How to design the framework to be able to use HTTP requests



How to create Test Data before test and clean up after test without UI actions



Practical example how to use HTTP lib from the Java code



Maintaining HTTP responses



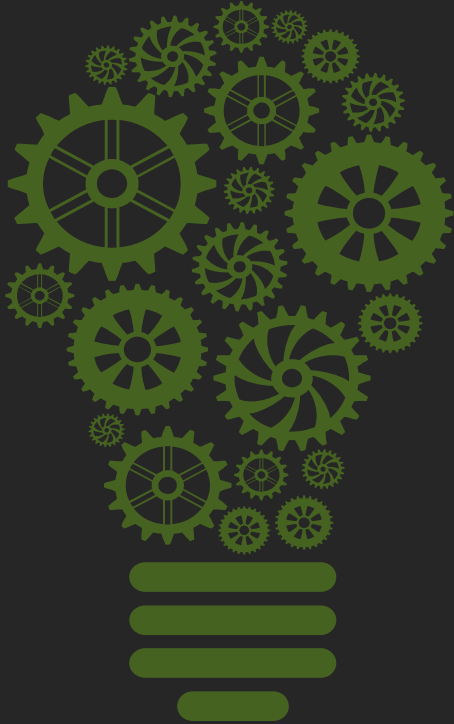
How to store login create/edit/delete objects without UI actions



Best practices of Selenium WebDriver

# What is Test Setup? Why is it Important?

---



A single Selenium test should test one and only thing. A bug in another part of the application that is not exactly related to the test should not cause the test to fail.



Every test should be independent. The test outcome should not be affected by another test in the suite.



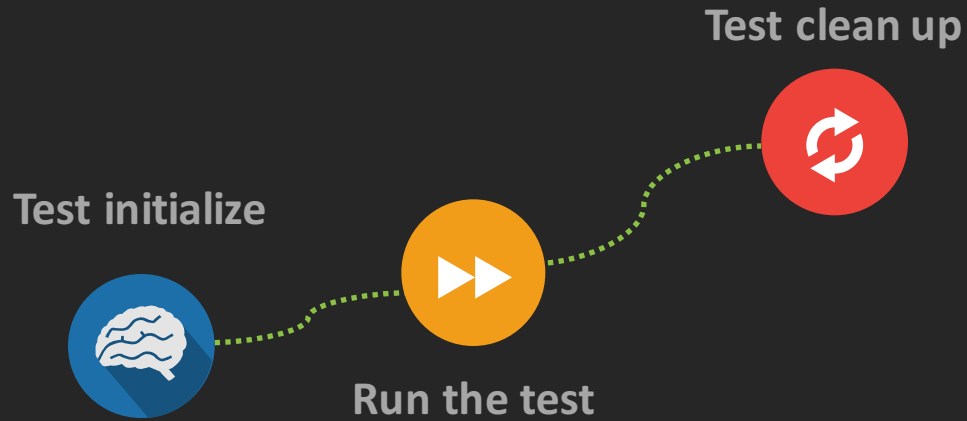
Make a faster tests. As quicker test suite as much useful it is.



Every test should create and clean the data before and the after the test run.

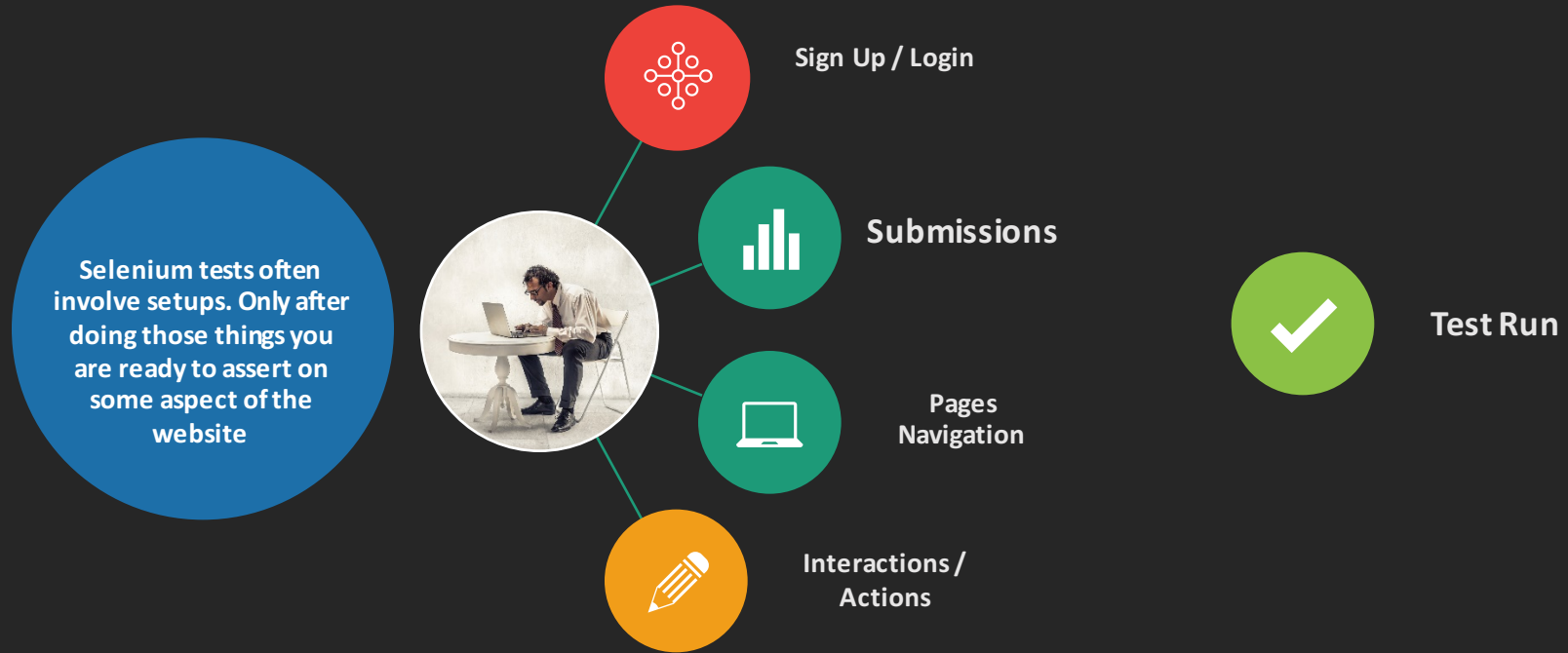
# Test Case Life Cycle

---



# Why should test data be created before test execution?

---





---

**TOOLS &  
LIBRARIES**

---

Java

---



Maven

---

***ma**v*en

TestNG

---



OkHttp

---

OkHttp3

Gson

---

{ G s o n }

Selenium

---





Help SQA Days

## DISCOVER PROJECTS

52882 public projects to discover

### ♥ MOST LIKED

Last week ▾



**Avans Portal**  
Avans portal project scrum board

♥ 5 👁 5 👤 7



**Bespın Test Project**  
Just for testing.

♥ 4 👁 3 👤 5



**SISTEMA EDUCATIVO**

♥ 4 👁 4 👤 5

### ↗ MOST ACTIVE

Last week ▾



**Valoo**

Regroupe tous les services de la plateforme : - Site web principal (www.valoo.com) - API principale - Serveur d'indexation - Serveur d'authentificatio...

♥ 12 👁 73 👤 27



**kohadocs 18.05**

Home of Koha's documentation (kohadocs) for the 18.05 release cycle. New to the project? Start with the Wiki! (book icon on the left)

♥ 4 👁 10 👤 9



**PortailPro**

♥ 0 👁 22 👤 13



---

# GETTING STARTED WITH SELENIUM

---

# Let's Test Like and Watching Buttons

The screenshot displays a web application interface for a project. The top navigation bar includes a search icon, a 'Taiga Support Page' link, and a 'SQA Days' profile icon. The main content area features a project card for 'PROJECT TEST SQA DAYS' with the subtitle 'Test Description SQA Days Minsk 2018'. The project card includes a purple icon, a 'Like' button with a count of 0, and a 'Watching' button with a dropdown arrow and a count of 1. A green circle highlights these two buttons. Below the project card, there is a section for 'Team' with a 'SQA Days' profile icon and a 'Test Description SQA Days Minsk 2018' link. The bottom of the page shows the URL 'https://tree.taiga.io/support'.

Navigation icons: Home, Search, List, Check, Book, Profile, Settings

Header: Taiga Support Page, SQA Days

Project Card: PROJECT TEST SQA DAYS, Test Description SQA Days Minsk 2018, Like 0, Watching 1

Team: SQA Days, Test Description SQA Days Minsk 2018

Footer: https://tree.taiga.io/support

## Let's Test Like and Watching Buttons

---



# TAIGA

## LOVE YOUR PROJECT

[Forgot it?](#)

LOGIN

Or login with



SIGN IN WITH GITLAB



SIGN IN WITH GITHUB

Not registered yet? [create your free account here](#)

## Let's Test Like and Watching Buttons

---



TAIGA

LOVE YOUR PROJECT

[Forgot it?](#)

LOGIN

Or login with



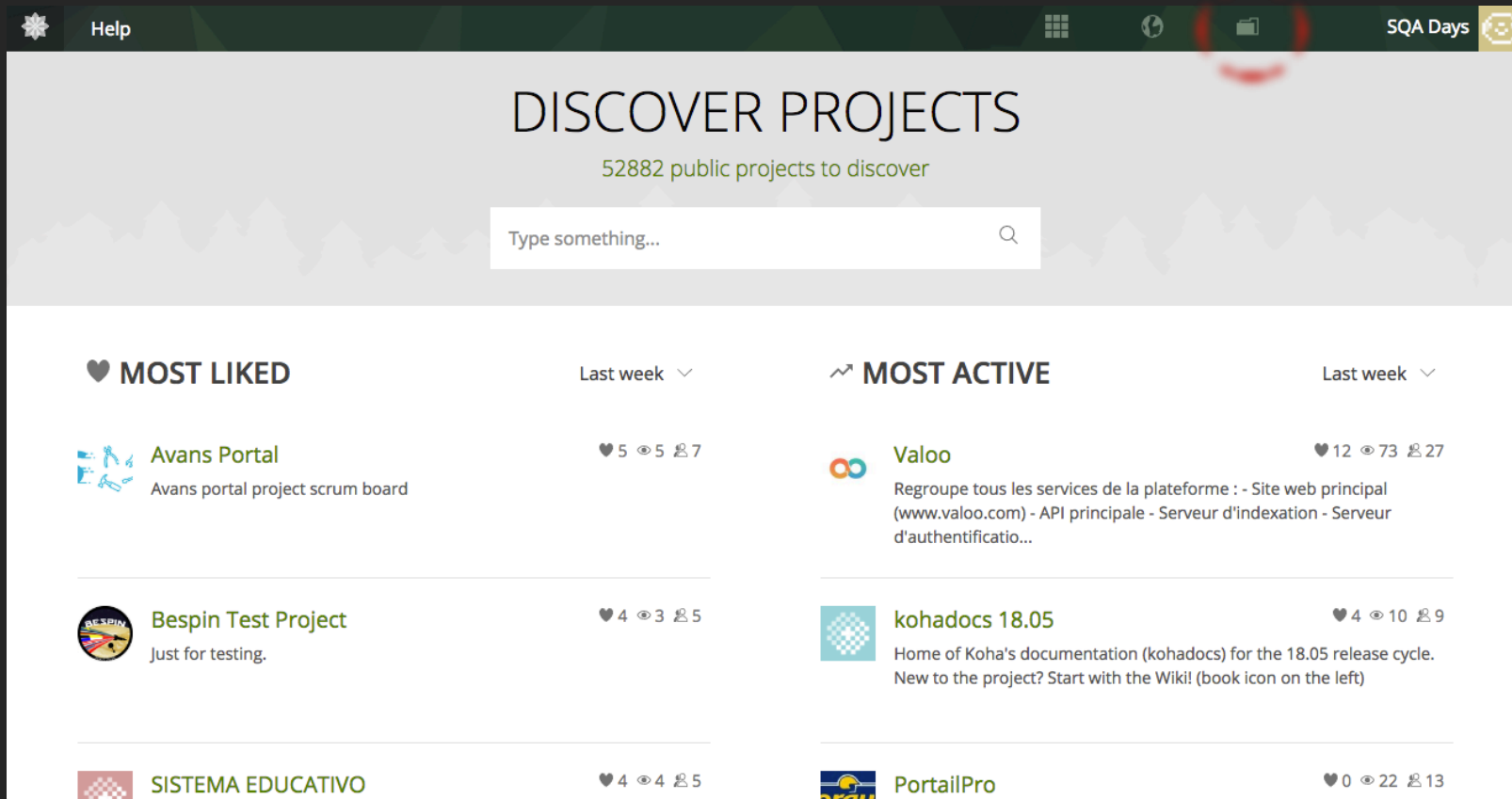
SIGN IN WITH GITLAB



SIGN IN WITH GITHUB

[Not registered yet? create your free account here](#)

# Let's Test Like and Watching Buttons



The screenshot shows the 'DISCOVER PROJECTS' interface. At the top, there is a dark green navigation bar with a 'Help' button on the left and a 'Like' button (represented by a heart icon) on the right, which is circled in red. The main header area is light gray and contains the text 'DISCOVER PROJECTS' and '52882 public projects to discover'. Below this is a search bar with the placeholder text 'Type something...'. The main content area is divided into two columns: 'MOST LIKED' and 'MOST ACTIVE'. Each column has a dropdown menu set to 'Last week'. The 'MOST LIKED' column lists three projects: 'Avans Portal', 'Bespin Test Project', and 'SISTEMA EDUCATIVO'. The 'MOST ACTIVE' column lists three projects: 'Valoo', 'kohadocs 18.05', and 'PortailPro'. Each project entry includes a small icon, the project name, a brief description, and a set of icons representing likes, views, and followers.

Help

SQA Days


## DISCOVER PROJECTS


52882 public projects to discover


Type something...

### ♥ MOST LIKED

Last week ▾


 **Avans Portal** ♥ 5 👁 5 👤 7  
Avans portal project scrum board


 **Bespin Test Project** ♥ 4 👁 3 👤 5  
Just for testing.


 **SISTEMA EDUCATIVO** ♥ 4 👁 4 👤 5

### ↗ MOST ACTIVE

Last week ▾

 **Valoo** ♥ 12 👁 73 👤 27  
Regroupe tous les services de la plateforme : - Site web principal (www.valoo.com) - API principale - Serveur d'indexation - Serveur d'authentificatio...

 **kohadocs 18.05** ♥ 4 👁 10 👤 9  
Home of Koha's documentation (kohadocs) for the 18.05 release cycle. New to the project? Start with the Wiki! (book icon on the left)

 **PortailPro** ♥ 0 👁 22 👤 13

# Let's Test Like and Watching Buttons

















The screenshot shows a web application interface with a dark green header. On the left, there is a 'Help' button with a star icon. On the right, there are icons for a grid, a globe, a folder, and a user profile labeled 'SQA Days'. Below the header is a light gray section titled 'MY PROJECTS' on the left and a green 'CREATE PROJECT' button on the right, which is circled in red. Below this is a list of eight project entries, each with a colored icon, a project name, and a description. To the right of the list, there is a text block explaining that the top 10 projects will appear in the top navigation bar project list.

Help

SQA Days

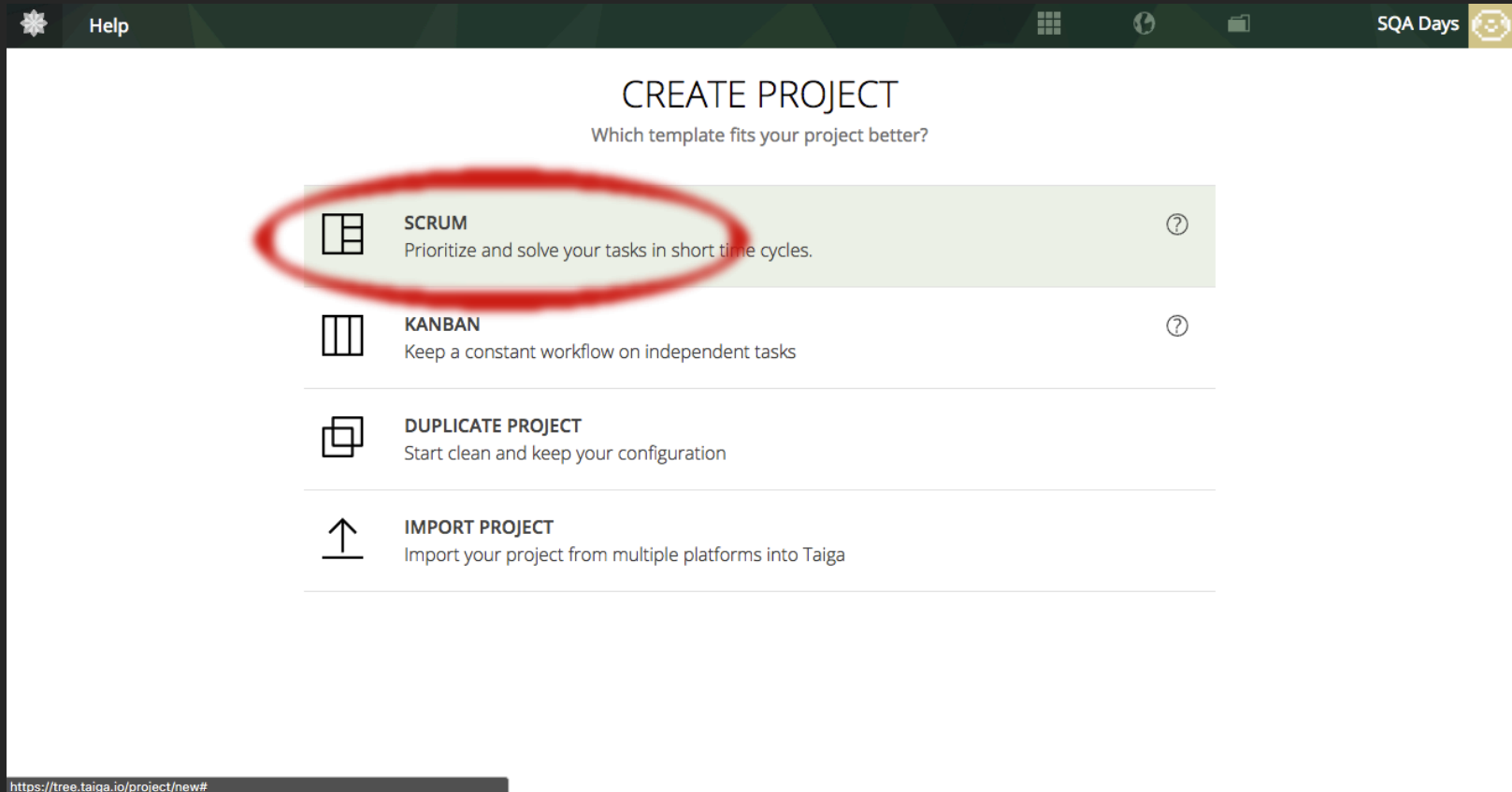
## MY PROJECTS

[CREATE PROJECT](#)

-  **zyrah**   
ekaaeewefkuthfzwebzv
-  **Test Project Name vdqxs**   
Test Project Descriptionxtefppjrne
-  **Test Project Name udhly**   
Test Project Descriptionpwwggoggd
-  **Test Project Name yrkln**   
Test Project Descriptionunmtcvysjz
-  **Test Project Name rowih**   
Test Project Descriptionxsxvonxmi
-  **Test Project Name fokum**   
Test Project Descriptioncfrxmqjxnb
-  **Test Project Name jivpe**   
Test Project Descriptionbnomuormqj
-  **Test Project Name vmnby**   
Test Project Descriptionwoyqzkeaw

Reorder your projects to set in the top the most used ones.  
The top 10 projects will appear in the top navigation bar project list

# Let's Test Like and Watching Buttons



The screenshot shows the 'CREATE PROJECT' interface in Taiga. The header includes a 'Help' button, a grid icon, a refresh icon, a folder icon, and 'SQA Days' with a smiley face icon. The main heading is 'CREATE PROJECT' with the subtext 'Which template fits your project better?'. There are four options listed:

- SCRUM**: Prioritize and solve your tasks in short time cycles. (This option is highlighted with a red circle and has a question mark icon to its right.)
- KANBAN**: Keep a constant workflow on independent tasks. (This option has a question mark icon to its right.)
- DUPLICATE PROJECT**: Start clean and keep your configuration.
- IMPORT PROJECT**: Import your project from multiple platforms into Taiga.

The URL at the bottom left is <https://tree.taiga.io/project/new#>.

# Let's Test Like and Watching Buttons

---

Help SQA Days

## SCRUM

Prioritize and solve your tasks in short time cycles.

New project details

Project Test SQA Days

Test Description SQA Days Minsk 2018

PUBLIC PROJECT  PRIVATE PROJECT

[BACK](#) [CREATE PROJECT](#)

# Let's Test Like and Watching Buttons

The screenshot shows a project management interface with a dark green header and a vertical sidebar on the left. The sidebar contains several icons, with the top three (a search icon, a hamburger menu icon, and a calendar icon) circled in red. The main content area is divided into two panels. The left panel is titled "PROJECT TEST SQA DAYS BACKLOG" and features a progress bar at 0%, statistics for 0 defined points, 0 closed points, and 0 points per sprint. Below this is a section titled "CUSTOMIZE YOUR BACKLOG GRAPH" with a bar chart icon and text explaining that users need to set up points and sprints through the Admin. At the bottom of this panel are buttons for "SHOW FILTERS", "+ ADD A NEW USER STORY", and a menu icon. The right panel is titled "SPRINTS" and contains a decorative graphic of a blue swirl and an orange leaf, along with the text "THERE ARE NO SPRINTS YET" and "You may want to create a new sprint in your project". At the bottom of the main content area is a decorative banner with a yellow background and green cacti, containing the text "THE BACKLOG IS EMPTY!" and "You may want to create a new user story".

Help

PROJECT TEST SQA DAYS BACKLOG

0% 0 defined points 0 closed points 0 points / sprint

CUSTOMIZE YOUR BACKLOG GRAPH

To have a nice graph that helps you follow the evolution of the project you have to set up the points and sprints through the **Admin**

SHOW FILTERS + ADD A NEW USER STORY

SPRINTS

THERE ARE NO SPRINTS YET  
You may want to create a new sprint in your project

THE BACKLOG IS EMPTY!  
You may want to create a new user story

# Let's Test Like and Watching Buttons

The screenshot shows a web interface for a project named "PROJECT TEST SQA DAYS". The page header includes "Taiga Support Page" on the left and "SQA Days" on the right. The project title is "PROJECT TEST SQA DAYS" with a subtitle "Test Description SQA Days Minsk 2018". A green oval highlights the interaction buttons: "Like" (0) and "Watching" (1). A notification from "SQA Days" states "created the project Project Test SQA Days" with a timestamp of "a few seconds ago". The project description "Test Description SQA Days Minsk 2018" is visible below. A "Team" section on the right shows a single member. The URL "https://tree.taiga.io/support" is at the bottom left.

Taiga Support Page

Help

SQA Days

PROJECT TEST SQA DAYS

Test Description SQA Days Minsk 2018

Like 0

Watching 1

SQA Days created the project Project Test SQA Days a few seconds ago

Team

Test Description SQA Days Minsk 2018

https://tree.taiga.io/support

## Let's Test Like and Watching Buttons

---

```
@Test
public void buttonsVisibility() {
    //login
    LoginPage loginPage = new LoginPage().open();
    loginPage.setUsername("sqa.days@yandex.ru");
    loginPage.setPassword("Armenia2018");
    HomePage homePage = loginPage.clickSubmit();

    //navigte
    homePage = homePage.init();
    ProjectsPage projectsPage = homePage.clickFolderIcon();

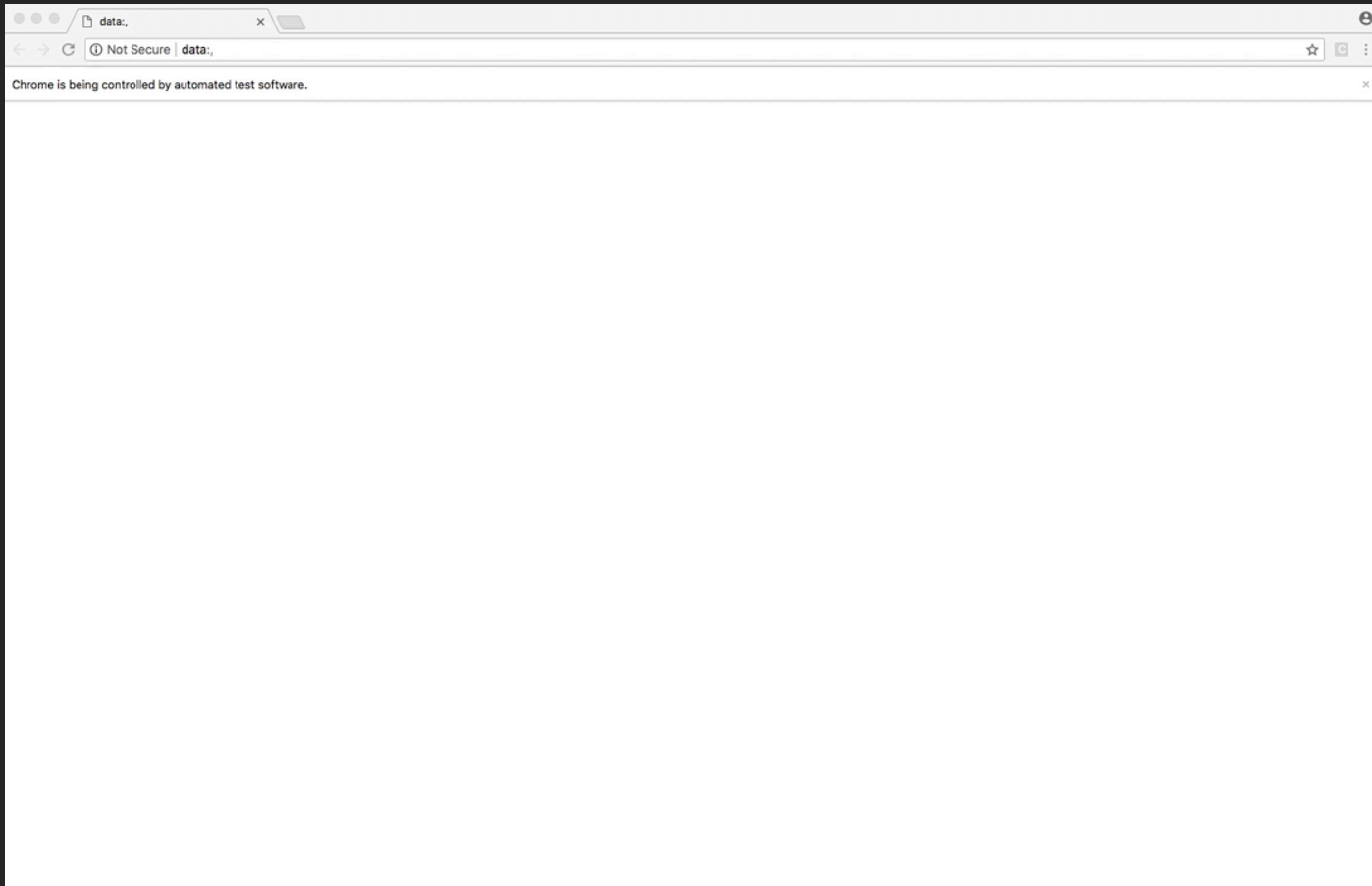
    //create project
    projectsPage = projectsPage.init();
    projectsPage.clickCreateProject();
    projectsPage.selectProjectType();
    projectsPage.setProjectName("Testing Project Name");
    projectsPage.setProjectDescription("Testing Project Description");
    projectsPage.submit();

    //navigate to Project page
    ProjectPage projectPage = new ProjectPage().init();
    projectPage.clickTimeline();

    //make an assertions
    assertTrue(projectPage.isLikeButtonVisible(), s:"Like button should be visible!");
    assertTrue(projectPage.istrackButtonVisible(), s:"Track button should be visible!");
}
```

# Let's Test Like and Watching Buttons

---





---

**LOGIN**

---

# What's Happening When You Click on Login?

---



# What's Happening When You Click on Login?

The screenshot shows a web browser window displaying the 'DISCOVER PROJECTS' page. The page title is 'DISCOVER PROJECTS' and the main content area shows '52911 public projects to discover'. The browser's developer tools are open, showing the Network tab. A red circle highlights the 'auth' request in the network log. The 'Headers' tab is selected, showing the following details:

- General**
  - Request URL: `https://api.taiga.io/api/v1/auth`
  - Request Method: POST
  - Status Code: 200 OK
  - Remote Address: 5.57.231.21:443
  - Referrer Policy: no-referrer-when-downgrade
- Response Headers**
  - Access-Control-Allow-Credentials: true
  - Access-Control-Allow-Headers: content-type, x-requested-with, authorization, accept-encoding, x-disable-pagination, x-lazy-pagination, x-host, x-session-id, set-headers
  - Access-Control-Allow-Methods: POST, GET, OPTIONS, PUT, DELETE, PATCH, HEAD
  - Access-Control-Allow-Origin: \*
  - Access-Control-Expose-Headers: x-pagination-count, x-paginated, x-paginated-by, x-pagination-current, x-pagination-next, x-pagination-prev, x-site-host, x-site-register, taiga-info-total-opened-milestones, taiga-info-total-closed-milestones, taiga-info-project-memberships, taiga-info-project-is-private, taiga-info-order-updated
  - Access-Control-Max-Age: 3600
  - Allow: POST, OPTIONS
  - Content-Encoding: gzip
  - Content-Language: en
  - Content-Type: application/json
  - Date: Mon, 21 May 2018 08:04:31 GMT
  - Server: gunicorn/19.7.1
  - Strict-Transport-Security: max-age=31536000;

## What's Happening When You Click on Login?

---

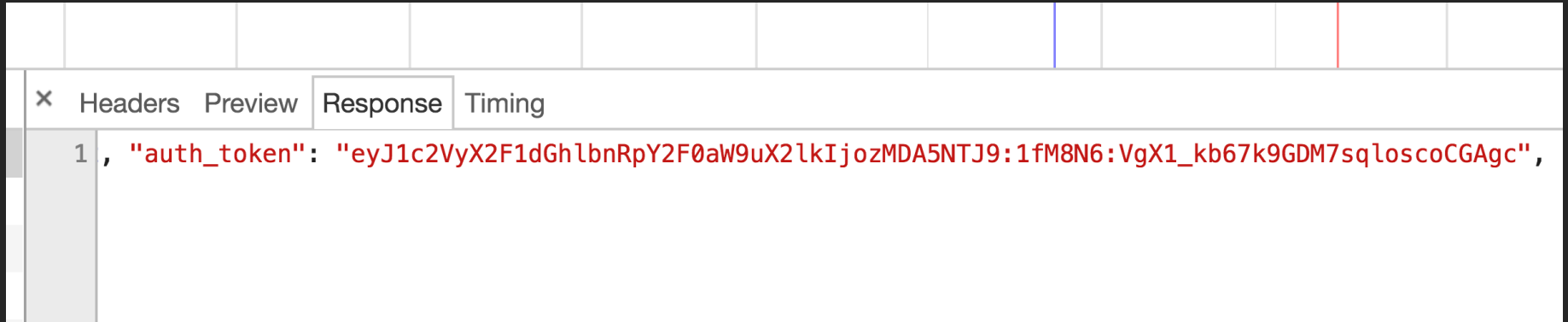
X-Session-Id: b70d0caadb1b3d1f5589c0c70c07d1f55802910

### ▼ Request Payload [view source](#)

```
▼ {username: "sqa.days@yandex.ru", password: "Armenia2018", type: "normal"}  
  password: "Armenia2018"  
  type: "normal"  
  username: "sqa.days@yandex.ru"
```

# What's Happening When You Click on Login?

---



# What's Happening When You Click on Login?

The screenshot shows a web application interface with a dark green header. The main content area is titled "DISCOVER PROJECTS" and displays "52911 public projects to discover". Below this is a search bar with the placeholder text "Type something...".

Below the search bar, there are two sections: "MOST LIKED" and "MOST ACTIVE". The "MOST LIKED" section shows a project named "Avans Portal" with 5 likes, 5 views, and 7 users. The "MOST ACTIVE" section shows a project named "Valoo" with 12 likes, 73 views, and 27 users.

The bottom part of the image shows a browser's developer console with the "Application" tab selected. The console displays a table of storage items:

Key	Value
intercom-state	{ "app": { "color": "#5b8200", "secondaryColor": "#5b8200", "name": "Taiga", "features": { "inboundMes..."
token	*eyJ1c2VyX2F1dGh1bnRpY2F0aW9uX2lkIjozMDA5NTJ9:1fKfKT:_hpWFZuoSZZ-NtYUJAhEaJClrOk*
userInfo	{ "email": "sqa.days@yandex.ru", "id": "300952", "date_joined": "2018-04-22T09:15:21.191Z", "max_m..."

The "token" value is highlighted in blue. Below the table, a single line of code is visible: "eyJ1c2VyX2F1dGh1bnRpY2F0aW9uX2lkIjozMDA5NTJ9:1fKfKT:\_hpWFZuoSZZ-NtYUJAhEaJClrOk".

# What's Happening When You Click on Login?

Help SQA Days

## DISCOVER PROJECTS

52911 public projects to discover

Type something...

### MOST LIKED

Last week

**Avans Portal** ❤️ 5 👁️ 5 👤 7  
Avans portal project scrum board

### MOST ACTIVE

Last week

**Valoo** ❤️ 12 👁️ 73 👤 27  
Regroupe tous les services de la plateforme : - Site web principal (www.valoo.com) - API principale - Serveur d'indexation - Serveur

Elements Console Sources Network Performance Memory Application Security Audits

service workers

Clear storage

Storage

- Local Storage
  - https://tree.taiga.io
    - userinfo
- Session Storage
- IndexedDB
- Web SQL
- Cookies

Cache

Key	Value
intercom-state	{ "app": { "color": "#5b8200", "secondaryColor": "#5b8200", "name": "Taiga", "features": { "inboundMes..." } }
token	"eyJ1c2VyX2F1dGhibnRpY2F0aW9uX2lkIjozMdA5NTJ9:1fKfkt:_hpWFZuoSZZ-NtYUJAhEaJClrOk"
userinfo	{ "email": "sqa.days@yandex.ru", "id": "300952", "date_joined": "2018-04-22T09:15:21.191Z", "max_m..." }
	<pre>▼ {email: "sqa.days@yandex.ru", id: 300952, date_joined: "2018-04-22T09:15:21.191Z", ...}   auth_token: "eyJ1c2VyX2F1dGhibnRpY2F0aW9uX2lkIjozMdA5NTJ9:1fKfkt:_hpWFZuoSZZ-NtYUJAhEaJClrOk"   big_photo: null   bio: ""   color: "#3c2d0c"   date_joined: "2018-04-22T09:15:21.191Z"</pre>

# What's Happening When You Click on Login?

The screenshot shows the Postman interface with a REST client request for a login endpoint. The request is a POST to `https://api.taiga.io/api/v1/auth` with the following body:

Key	Value	Description
<input checked="" type="checkbox"/> username	sqa.days@yandex.ru	
<input checked="" type="checkbox"/> password	Armenia2018	
<input checked="" type="checkbox"/> type	normal	
New key	Value	Description

The response is a 200 OK status with a JSON body:

```
1- {
2  "big_photo": null,
3  "id": 300952,
4  "roles": [
5    "Product Owner"
6  ],
7  "max_private_projects": 1,
8  "timezone": "",
9  "theme": "",
10 "full_name": "SQA Days",
11 "gravator_id": "c0a11ea4579ef0cb9447e711c33eda5",
12 "max_memberships_public_projects": null,
13 "bio": "",
14 "total_public_projects": 30,
15 "full_name_display": "SQA Days",
16 "max_memberships_private_projects": 3,
17 "is_active": true,
18 "date_joined": "2018-04-22T09:15:21.191Z",
19 "max_public_projects": null,
20 "lang": "",
21 "username": "sqa.days.test",
22 "uid": "1de5028f72944bc69133079e0bee4456",
23 "total_private_projects": 0,
24 "photo": null,
25 "email": "sqa.days@yandex.ru",
26 "color": "#3c2d0c",
27 "auth_token": "eyJ1c2VyX2F1dGh1bnR5ZF0aW9uX21k1jozM0ASNTJ39:1fKfS6:GqNpmN3Yk0k-gv8HrGd9ggk6mFA"
28 }
```

## What's Happening When You Click on Login?

---

```
public static String login(String username, String password) {
    Response response;
    String responseJson;
    OkHttpClient client = new OkHttpClient();

    MediaType mediaType = MediaType.parse("application/x-www-form-urlencoded");
    RequestBody body = RequestBody.create(mediaType, content: "username=" + username +
        "&password=" + password + "&type=normal");
    Request request = new Request.Builder()
        .url(BASE_URL + "/auth")
        .post(body)
        .addHeader(name: "Content-Type", value: "application/x-www-form-urlencoded")
        .build();

    try {
        response = client.newCall(request).execute();
        responseJson = response.body().string();
    } catch (IOException e) {
        throw new ApiException(e.getMessage());
    }
    return responseJson;
}
```

## What's Happening When You Click on Login?

---

```
protected void login(String username, String password) {
    String loginResponse = HttpClient.login(username, password);
    new LoginPage().open();

    ((JavascriptExecutor) getDriver()).executeScript(
        s: "window.localStorage.setItem('token', '"
        + new JsonParser().parse(loginResponse).getAsJsonObject().get("auth_token") + "');"");

    ((JavascriptExecutor) getDriver()).executeScript(
        s: "window.localStorage.setItem('userInfo', '"
        + loginResponse + "');"");
}
```

```
login( username: "sqa.days@yandex.ru", password: "Armenia2018");
```



---

**CREATE PROJECT**

---

# How to Create Project

The screenshot shows a web browser window with the URL `https://tree.taiga.io/project/sqadaystest-project-test-sqa-days-1/backlog`. The page content includes a search bar, a navigation menu, and a main area with a bar chart and text: "CUSTOMIZE YOUR BACKLOG GRAPH" and "THERE ARE NO SPRINTS YET". Below the page content, a network inspector tool is open, displaying a list of requests. The selected request is a POST to `https://api.taiga.io/api/v1/projects`. The network inspector shows a timeline of requests with response times ranging from 200 ms to 3800 ms. The selected request has a response time of 200 ms. The network inspector also shows the request headers and response headers for the selected request.

Backlog - Project Test SQA Days 1

Secure `https://tree.taiga.io/project/sqadaystest-project-test-sqa-days-1/backlog`

Apps Personal Workfront Travel Erasmus Plus C-Labs Tech & Life Berlin Selenium Course TESTING

SEARCH

CUSTOMIZE YOUR BACKLOG GRAPH

To have a nice graph that helps you follow the evolution of the project you have to set up the points and sprints through the **Admin**

THERE ARE NO SPRINTS YET

You may want to create a new sprint in

Elements Console Sources Network Performance Memory Application Security Audits

View: Group by frame Preserve log Disable cache Offline Online

Filter Hide data URLs XHR JS CSS Img Media Font Doc WS Manifest Other

200 ms 400 ms 600 ms 800 ms 1000 ms 1200 ms 1400 ms 1600 ms 1800 ms 2000 ms 2200 ms 2400 ms 2600 ms 2800 ms 3000 ms 3200 ms 3400 ms 3600 ms 3800 ms

Name Headers Preview Response Timing

- projects
- projects
- 5-sB2DOph\_s-UdmUfxyPV6Hk9Wkk...
- by\_slug?slug=sqadaystest-project-te...
- collect?v=1&\_v=j67&a=530967766&t...
- events
- by\_slug?slug=sqadaystest-project-te...
- stats
- milestones?closed=false&project=27...
- userstories?milestone=null&page=1&...
- menu.png
- collect?v=1&\_v=j67&a=530967766&t...
- empty\_mex.png
- empty\_sprint.png
- stats
- milestones?closed=false&project=27...
- userstories?milestone=null&page=1&...

22 requests | 45.9 KB transferred

General

- Request URL: `https://api.taiga.io/api/v1/projects`
- Request Method: POST
- Status Code: 201 Created
- Remote Address: 5.57.231.21:443
- Referrer Policy: no-referrer-when-downgrade

Response Headers

- Access-Control-Allow-Credentials: true
- Access-Control-Allow-Headers: content-type,x-requested-with,authorization,accept-encoding,x-disable-pagination,x-lazy-pagination,x-host,x-session-id,set-headers
- Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE,PATCH,HEAD
- Access-Control-Allow-Origin: \*
- Access-Control-Expose-Headers: x-pagination-count,x-paginated,x-paginated-by,x-pagination-current,x-pagination-next,x-pagination-prev,x-site-host,x-site-register,taiga-info-total-opened-milestones,taiga-info-total-closed-milestones,taiga-info-project-memberships,taiga-info-project-is-private,taiga-info-order-updated
- Access-Control-Max-Age: 3600
- Allow: GET,POST,HEAD,OPTIONS
- Content-Language: en
- Content-Length: 12315
- Content-Type: application/json
- Date: Mon, 21 May 2018 12:29:17 GMT
- Server: gunicorn/19.7.1
- Strict-Transport-Security: max-age=31536000;

## How to Create Project

---

### ▼ Request Payload

[view source](#)

```
▼ {is_private: false, creation_template: 1, name: "SQA Days 2018 Project",...}  
  creation_template: 1  
  description: "SQA Days 2018 Project Description"  
  is_private: false  
  name: "SQA Days 2018 Project"
```

# How to Create Project

The screenshot displays the Postman interface for a REST client. The top bar shows the workspace name "My Workspace" and an "Upgrade" button. The main area is divided into a left sidebar and a main workspace.

**Left Sidebar:**

- Search: Filter
- History: Clear all
- Collections
- Today:
  - POST https://api.taiga.io/api/v1/projects
  - POST https://api.taiga.io/api/v1/projects
  - POST https://api.taiga.io/api/v1/projects
  - POST https://api.taiga.io/api/v1/projects

**Main Workspace:**

- URL: https://api.taiga.io/api/v1/projects
- Method: POST
- Body: raw (JSON application/json)
- Request Body (Raw):

```
1- {  
2   "is_private": false,  
3   "creation_template": 1,  
4   "name": "Project Test SQA Days",  
5   "description": "Project Test Description SQA Days"  
6 }
```
- Status: 201 Created, Time: 1017 ms, Size: 12.92 KB
- Response Body (Pretty):

```
1- {  
2   "issue_custom_attributes": null,  
3   "created_date": "2018-05-21T12:35:01.812Z",  
4   "is_kanban_activated": false,  
5   "is_wiki_activated": true,  
6   "public_permissions": [  
7     "view_milestones",  
8     "view_wiki_links",  
9     "view_issues",  
10    "view_tasks",  
11    "view_epics",  
12    "view_us",  
13    "view_project",  
14    "view_wiki_pages"  
15  ],  
16   "default_issue_status": 1897217,  
17   "transfer_token": null,  
18   "priorities": [  
19     {  
20       "order": 1,  
21       "id": 813866,  
22       "color": "#666666",  
23       "project_id": 270625,  
24       "name": "Low"  
25     },  
26     {  
27       "order": 3,  
28       "id": 813867,  
29       "color": "#666666",  
30       "project_id": 270625,  
31       "name": "Medium"  
32     }  
33   ]  
34 }
```

## How to Create Project

---

```
public static Response post(String url, String jsonInString) {
    if (ACCESS_TOKEN == null) {
        String jsonString = login(USERNAME, PASSWORD);
        ACCESS_TOKEN = parser.parse(jsonString).getAsJsonObject().get("auth_token").getAsString();
    }
    OkHttpClient httpClient = new OkHttpClient();
    MediaType mediaType = MediaType.parse("application/json");
    RequestBody body = RequestBody.create(mediaType, jsonInString);
    String authorization = "Bearer " + ACCESS_TOKEN;
    Request request = new Request.Builder()
        .url(BASE_URL + url)
        .post(body)
        .addHeader("authorization", authorization)
        .addHeader("content-type", "application/json")
        .build();
    Response response;
    try {
        response = httpClient.newCall(request).execute();
    } catch (IOException e) {
        throw new ApiException(e);
    }

    if (!response.isSuccessful())
        throw new ApiException("Failed : HTTP error code : "
            + String.valueOf(response.code()));

    return response;
}
```

## How to Create Project

---

```
public static JsonObject createProject() {
    HashMap projectMap = new HashMap();
    projectMap.put("is_private", false);
    projectMap.put("creation_template", 1);
    projectMap.put("name", "Test Project Name " + randomString(len:5));
    projectMap.put("description", "Test Project Description" + randomString(len:10));
    String jsonString = gson.toJson(projectMap);
    Response response = HttpClient.post(url:"/projects", jsonString);
    return getJsonObject(response);
}
```

## Integrate with Page Object Patten

---

 Secure | <https://tree.taiga.io/project/sqadaystest-sqa-days-2018-project/>

```
public ProjectPage(JsonObject project) {
    super(getDriver());
    this.name = project.get("name").getAsString();
    this.id = project.get("id").getAsString();
    this.project = project;
}
```

```
@Override
public String getPageUrl() {
    return "/project/" + USERNAME + "-" + name.replace(target: " ", replacement: "-").toLowerCase();
}
```

## The Same Test with Rest API setup

---

**@BeforeMethod**

**public void setUp() {**

```
login( username: "sqa.days@yandex.ru", password: "Armenia2018");
```

```
project = ApiHelper.createProject();
```

**}**

**@AfterMethod**

**public void tearDown() {**

```
ApiHelper.deleteProject(project);
```

**}**

**@Test**

**public void trackAndLikeButtonsVisibility() {**

```
ProjectPage projectPage = new ProjectPage(project);
```

```
projectPage = projectPage.open();
```

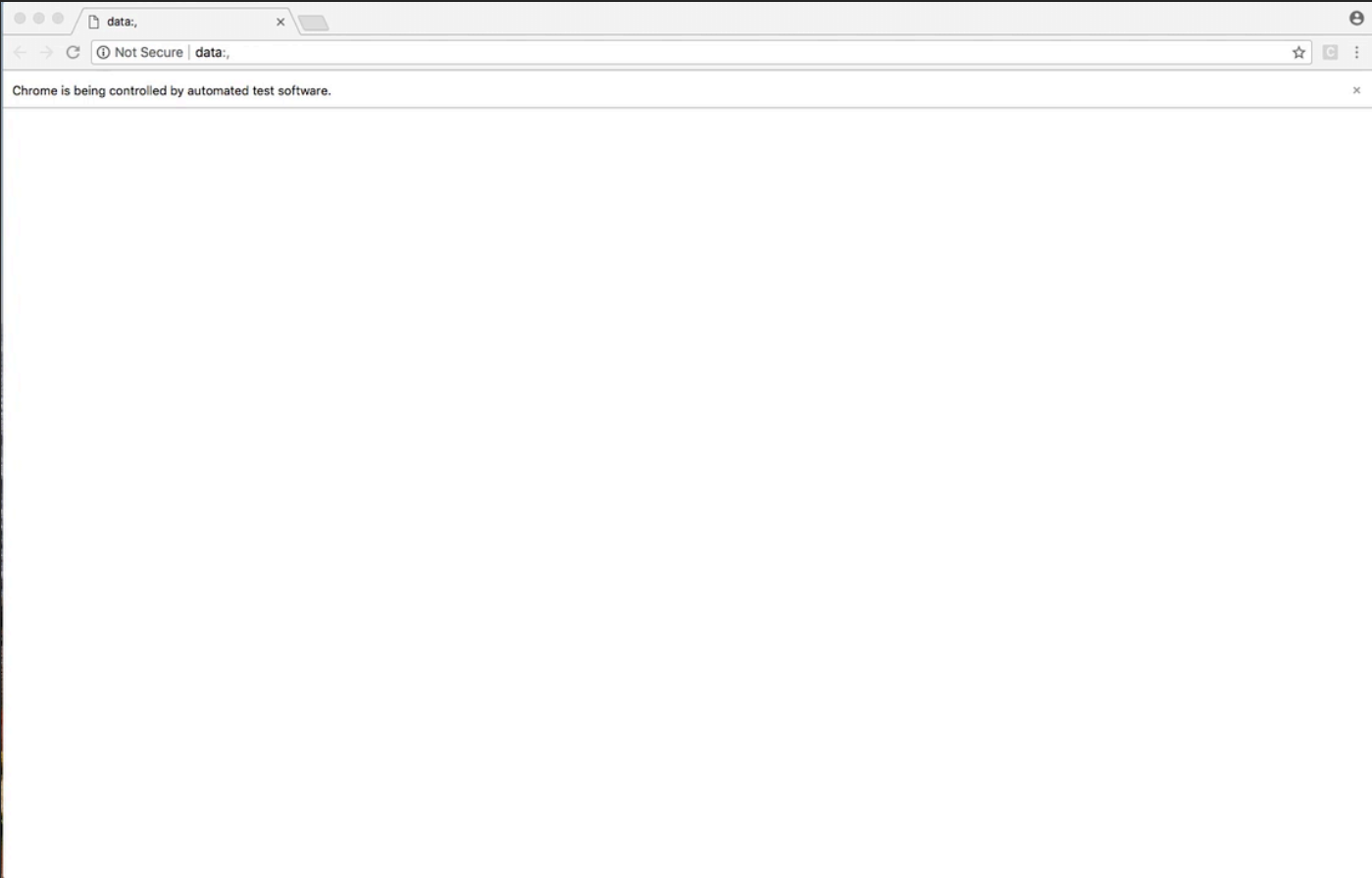
```
assertTrue(projectPage.isLikeButtonVisible(), s: "Like button should be visible!");
```

```
assertTrue(projectPage.istrackButtonVisible(), s: "Track button should be visible!");
```

**}**

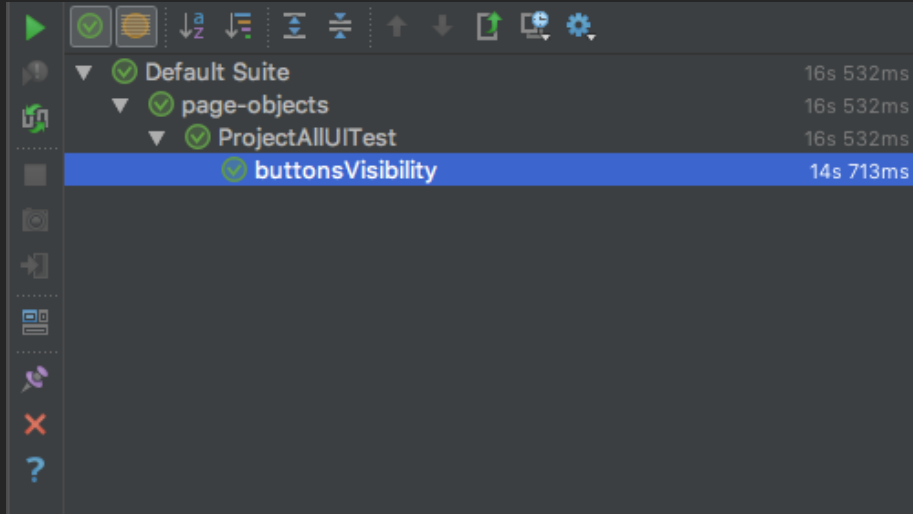
# The Test Run with Rest API setup

---



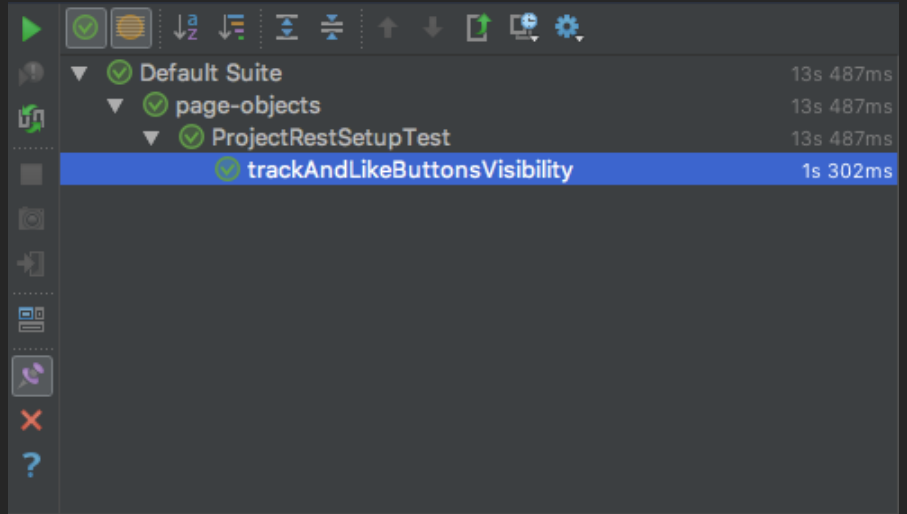
# Test Run Time

---



This screenshot shows a test run summary for 'ProjectAllUITest'. The test suite is 'Default Suite' with a total duration of 16s 532ms. The test is located under 'page-objects' and has a duration of 14s 713ms. The interface includes a toolbar with icons for play, checkmarks, sort, expand/collapse, and other test management functions.

Test Suite	Duration
Default Suite	16s 532ms
page-objects	16s 532ms
ProjectAllUITest	16s 532ms
buttonsVisibility	14s 713ms



This screenshot shows a test run summary for 'ProjectRestSetupTest'. The test suite is 'Default Suite' with a total duration of 13s 487ms. The test is located under 'page-objects' and has a duration of 1s 302ms. The interface includes a toolbar with icons for play, checkmarks, sort, expand/collapse, and other test management functions.

Test Suite	Duration
Default Suite	13s 487ms
page-objects	13s 487ms
ProjectRestSetupTest	13s 487ms
trackAndLikeButtonsVisibility	1s 302ms

## Create Issue in Project with API

---

```
public static JsonObject createIssue(JsonObject project) {
    HashMap projectMap = new HashMap();
    projectMap.put("project", project.get("id").getAsInt());
    projectMap.put("subject", "Test Issue Subject " + randomString(len: 5));
    projectMap.put("description", "Test Issue Description" + randomString(len: 10));
    String jsonString = gson.toJson(projectMap);
    Response response = HttpClient.post(url: "/issues", jsonString);
    return getJsonObject(response);
}
```



---

**CLEAN UP AFTER**

---

# Delete Created Project

The screenshot shows a web browser window with the URL `https://tree.taiga.io`. The page title is "PROJECTS DASHBOARD". There are two main tabs: "WORKING ON" and "WATCHING". A user profile for "zyrah" is visible in the top right corner.

The browser's developer tools are open to the Network tab. A list of requests is shown on the left, with the request `270498` selected. The details for this request are as follows:

- Name:** 270498
- Request URL:** `https://api.taiga.io/api/v1/projects/270498`
- Request Method:** DELETE
- Status Code:** 204 No Content
- Remote Address:** 5.57.231.21:443
- Referrer Policy:** no-referrer-when-downgrade
- Response Headers:**
  - Access-Control-Allow-Credentials: true
  - Access-Control-Allow-Headers: content-type, x-requested-with, authorization, accept-encoding, x-disable-pagination, x-lazy-pagination, x-host, x-session-id, set-orders
  - Access-Control-Allow-Methods: POST, GET, OPTIONS, PUT, DELETE, PATCH, HEAD
  - Access-Control-Allow-Origin: \*
  - Access-Control-Expose-Headers: x-pagination-count, x-paginated, x-paginated-by, x-pagination-current, x-pagination-next, x-pagination-prev, x-site-host, x-site-registered, taiga-info-total-opened-milestones, taiga-info-total-closed-milestones, taiga-info-project-memberships, taiga-info-project-is-private, taiga-info-order-updated
  - Access-Control-Max-Age: 3600
  - Allow: GET, PUT, PATCH, DELETE, HEAD, OPTIONS
  - Content-Language: en
  - Content-Length: 0
  - Date: Mon, 21 May 2018 13:29:22 GMT
  - Server: gunicorn/19.7.1
  - Strict-Transport-Security: max-age=31536000;
  - Vary: Accept-Language

At the bottom of the network inspector, it shows "26 requests | 22.9 KB transferred".

## Delete Created Project

---

```
public static void deleteProject(JsonObject project) {  
    HttpClient.delete(url: "/projects/" + project.get("id").getAsString());  
}
```



---

# TIPS AND TRICKS

---

## Login Info in Cookies

---

```
public void login(User user) {  
    new LoginPage().open();  
    getWebDriver().manage().addCookie(new Cookie( name: "c-labs.token", getApiHelper().login(user)));  
}
```

## Manage Cookies

---

```
public void addCookie(String cookieName, String cookieValue) {  
    Cookie cookie = new Cookie(cookieName, cookieValue);  
    getDriver().manage().addCookie(cookie);  
}
```

```
public Cookie getCookie(String cookieName) {  
    return getDriver().manage().getCookieNamed(cookieName);  
}
```

```
public Set<Cookie> getAllCookie() {  
    return getDriver().manage().getCookies();;  
}
```

```
public void deleteCookie(String cookieName) {  
    getDriver().manage().deleteCookieNamed(cookieName);  
}
```

```
public void deleteAllCookie() {  
    getDriver().manage().deleteAllCookies();  
}
```

## Download Selenium Drivers with Maven plugin

---

```
<plugin>
  <groupId>com.github.webdriverextensions</groupId>
  <artifactId>webdriverextensions-maven-plugin</artifactId>
  <version>3.1.2</version>
  <executions>
    <execution>
      <goals>
        <goal>install-drivers</goal>
      </goals>
    </execution>
  </executions>
  <configuration>
    <installationDirectory>./src/main/resources/drivers</installationDirectory>
    <drivers>
      <driver>
        <name>chromedriver</name>
        <platform>mac</platform>
        <bit>64</bit>
        <version>2.37</version>
      </driver>
      <driver>
        <name>chromedriver</name>
        <platform>linux</platform>
        <bit>64</bit>
        <version>2.37</version>
      </driver>
    </drivers>
  </configuration>
</plugin>
```

## Execute JavaScript to Click on Element

---

```
public void clickJS(WebElement element) {  
    JavascriptExecutor javascriptExecutor = (JavascriptExecutor) getDriver();  
    javascriptExecutor.executeScript(s: "arguments[0].click();", element);  
}
```

## Wait for Page Load

---

```
public void waitForPageLoaded() {
    ExpectedCondition<Boolean> expectation = driver ->
        ((JavascriptExecutor) driver).executeScript(s:"return document.readyState").toString().equals("complete");
    try {
        Thread.sleep(millis: 1000);
        WebDriverWait wait = new WebDriverWait(getDriver(), timeOutInSeconds: 30);
        wait.until(expectation);
    } catch (Throwable error) {
        Assert.fail(s:"Timeout waiting for Page Load Request to complete.");
    }
}
```

## Take a Screenshot on Failure

---

```
public static void takeScreenshot(WebDriver webdriver, String filePath) throws Exception {  
    TakesScreenshot scrShot = ((TakesScreenshot) webdriver);  
    File SrcFile = scrShot.getScreenshotAs(OutputType.FILE);  
    File DestFile = new File(filePath);  
    FileUtils.copyFile(SrcFile, DestFile);  
}
```

```
public String getInnerHTML() {  
    WebElement element = getDriver().findElement(By.id("form-submit"));  
    return element.getAttribute(s: "outerHTML");  
}
```

```
public void maximizeWondow() {  
    getDriver().manage().window().maximize();  
}
```

```
public WebDriver maximizeWondowChrome() {  
    ChromeOptions options = new ChromeOptions();  
    options.addArguments("--start-maximized");  
    return new ChromeDriver( options );  
}
```



• **Thank You!**

Have a Nice Day...



[sargis.sargsyan@live.com](mailto:sargis.sargsyan@live.com)



[/sargissargsyan](#)



[@sargisqa](#)