

NEW YORK STOCK EXCHANGE

qase.io testing  
automation

# Table of contents

01

Task  
overview

02

Implementation

03

Framework's  
pros and cons

04

Troubleshooting

05

Summary

06

Working in  
team

# 01

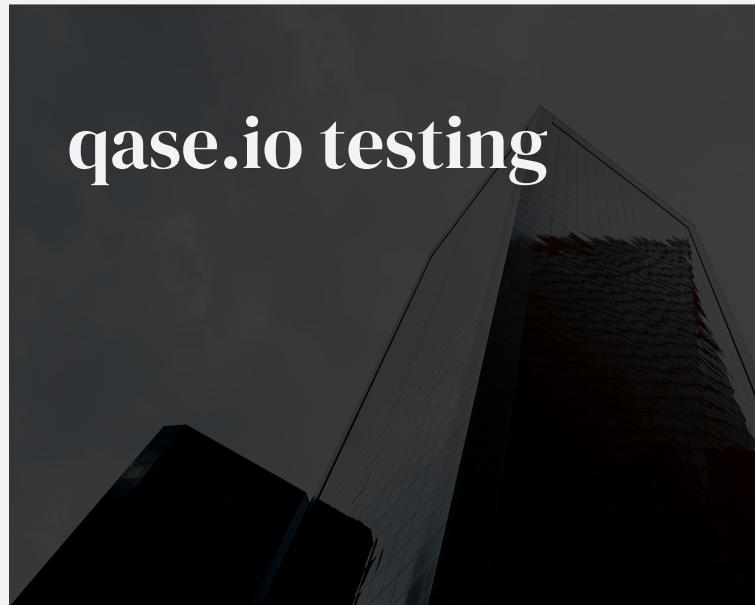
## Task overview

## Tasks:

- Create test cases in Qase management system
- Implement Automation framework for testing of Qase API and UI
- Setup Jenkins job to trigger automation test run at 11 AM each day
- At the end of each run allure report should be generated

## Tech stack:

- .Net Core • Nunit • Selenium • Fluent Assertions
- RestSharp • Git • Allure Reports • Jenkins





# Implementation

# 02

# Test cases in Qase management system

Diploma repository 8 suites | 20 tests ☰ Views ▾

+ Suite + Case Search for cases + Add filter

☰ Suites API + ⚙️ 🗑️

Suite	Count
API	15 ...
Projects	4 ...
Defects	5 ...
Test cases	5 ...
Authentication	1 ...
UI	5 ...
Projects	3 ...
Authorization	2 ...

API tests

+ Create quick test

Projects + ⚙️ 🗑️

Project api CRUD tests

- ⬆️⚙️ Diploma-29 Create a project via api with only required data
- ⬆️⚙️ Diploma-30 Read a project by code via api
- ⬆️⚙️ Diploma-31 Delete a project via api
- ⬆️⚙️ Diploma-32 Read all projects via api

+ Create quick test

Defects + ⚙️ 🗑️

Defect api CRUD tests

- ⬆️⚙️ Diploma-33 Create a defect via api with only required data
- ⬆️⚙️ Diploma-34 Update a defect via api with only required data

# Test cases in Qase management system

Diploma repository 8 suites | 20 tests Views ▾

+ Suite + Case Search for cases + Add filter

**Suites**

- API 15 ...
- Projects 4 ...
- Defects 5 ...
- Test cases 5 ...
- Authentication 1 ...
- UI 5 ...
- Projects 3 ...
- Authorization 2 ...

**UI** + ⚙️ 🗑️ 🗑️

UI tests  
+ Create quick test

**Projects** + ⚙️ 🗑️ 🗑️

CRUD ui tests for Projects functionality

- ↑ ⚙️ Diploma-11 Create a project with only required data filled
- ↑ ⚙️ Diploma-12 Update a project with only required data filled
- ↑ ⚙️ Diploma-13 Delete a project

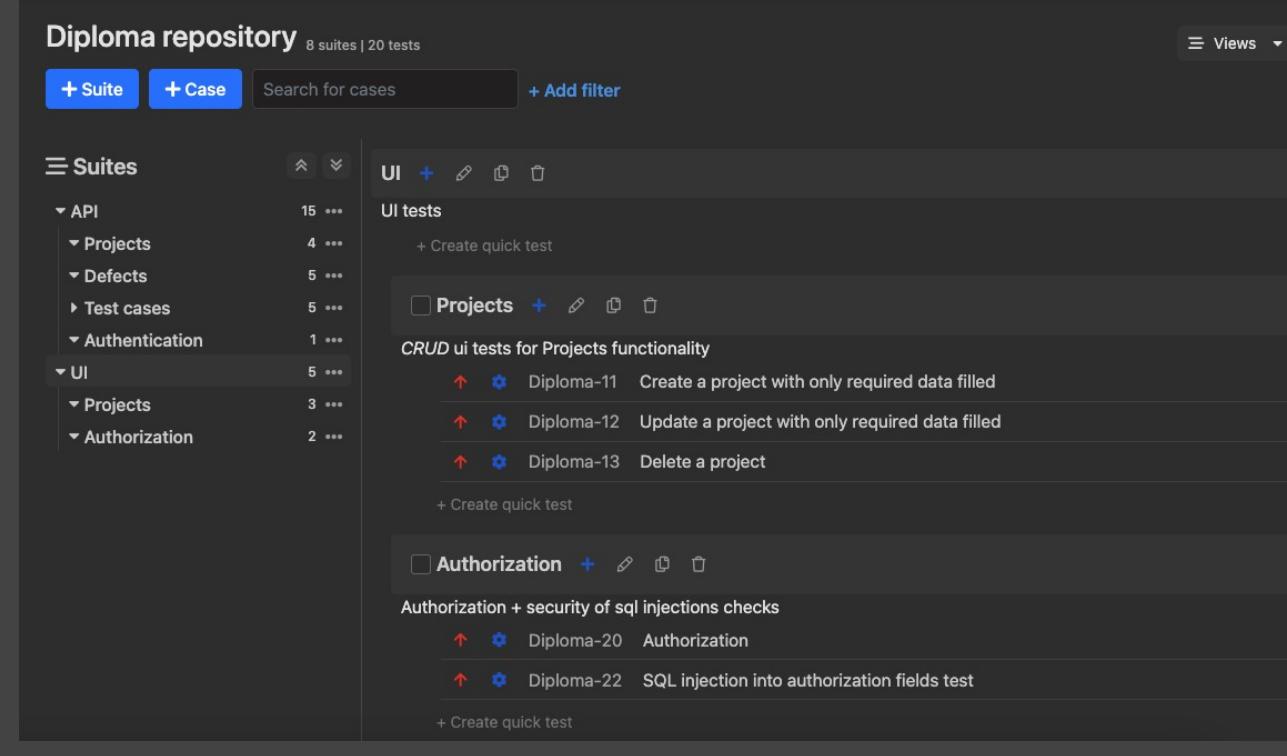
+ Create quick test

**Authorization** + ⚙️ 🗑️ 🗑️

Authorization + security of sql injections checks

- ↑ ⚙️ Diploma-20 Authorization
- ↑ ⚙️ Diploma-22 SQL injection into authorization fields test

+ Create quick test



# At least 5 API tests

- positive
- negative
- boundary

API		17
Authentication-API	#1 Authentication using invalid data	801ms
	#2 Authentication using no authentication set in client	794ms
	#3 Authentication using valid data	945ms
Defects-API		5
	#1 Create a new defect with required data filled	248ms
	#2 Update the defect with required data filled	266ms
	#3 Read the defect	362ms
	#4 Delete the defect	241ms
	#5 Read all the remaining defects	276ms
Projects-API		4
	#1 Create a new project with required data filled	957ms
	#2 Read the project	246ms
	#3 Delete the project	251ms
	#4 Read all the remaining projects	356ms
Test cases-API		5
	#1 Create a test case with required data filled	411ms
	#2 Update the test case with required data filled	368ms
	#3 Read the test case	1s 008ms
	#4 Delete the test case	323ms
	#5 Read the remaining test cases	260ms

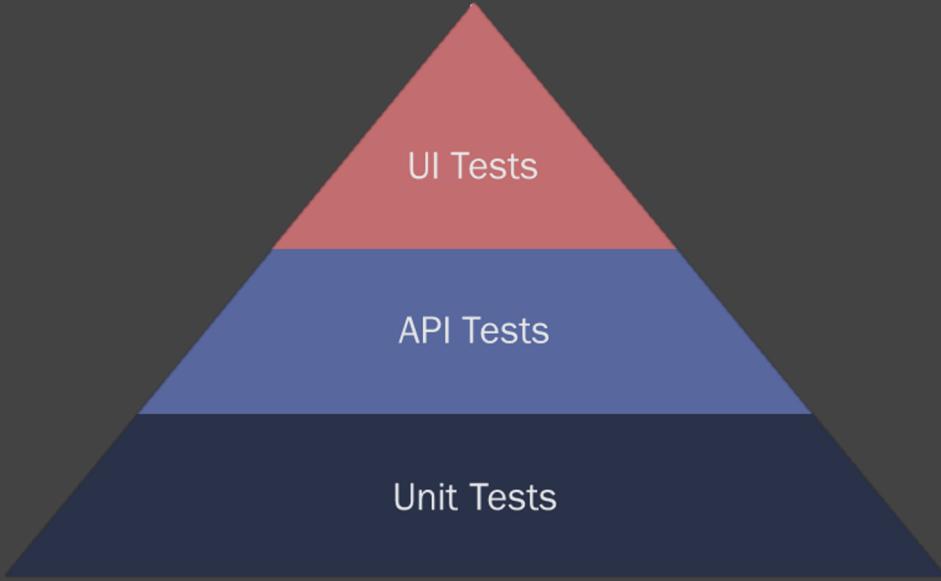
# At least 7 UI tests

- positive
- negative
- boundary
- security

UI		8
Authorization-UI		2
#1	Authorization using invalid data	11s 715ms
#2	Authorization using valid data	26s 253ms
Projects-UI		3
#1	Create a project with requited fields filled	24s 087ms
#2	Update the project with required fields filled	13s 031ms
#3	Delete the updated project	3s 999ms
SQL Injections		3
#1	Sql injections input into password field ' or ''	17s 096ms
#2	Sql injections input into password field UNION ALL SELECT USER()--	12s 812ms
#3	Sql injections input into password field admin' or 1=1	15s 869ms

## Why 8 to 17?

The test automation pyramid lays out the types of tests that should be included in an automated test suite. It also outlines the sequence and frequency of these tests



The agile test automation pyramid

## At least 2 data - driven tests

```
[Test]
[Category(name: "Security")]
[AllureSuite("SQL Injections")]
[AllureStory("SQL Injections")]
[TestCase("'" or ''), TestCase("UNION ALL SELECT USER()--"), TestCase("admin' or 1=1")]
[AllureName("Sql injections input into password field")]
[AllureTms(name: "tms", url: "suite=9&previewMode=modal&case=22")]
public void Authorization_InsertedSqlInjection_ErrorMessageDisplayed(string sqlInjections)
{
    _welcomingPage // WelcomingPage
        .ProceedToLoggingIn()
        .PopulateAuthorizationData(login: Configurator.Admin.Email, password: sqlInjections) // AuthorizationPage
        .SubmitAuthorizationForm();

    using (new AssertionScope())
    {
        _authorizationPage.ErrorMessageDisplayed().Should().BeTrue();
        _authorizationPage.ErrorMessageText().Should().Be("These credentials do not match our records.");
    }
}
```

# At least 2 data - driven tests

```
[Test]
[Category( name: "Positive")]
[AllureSuite("Authorization-UI")]
[AllureName("Authorization using valid data")]
[AllureTms( name: "tms", url: "suite=9&previewMode=modal&case=20")]
public void Authorization_PopulatedValidData_ProjectsPageOpened()
{
    _welcomingPage // WelcomingPage
        .ProceedToLoggingIn()
        .PopulateAuthorizationData( login: Configurator.Admin.Email, Configurator.Admin.Password ) // AuthorizationPage
        .SubmitAuthorizationForm();

    _projectsPage.PageOpened.Should().BeTrue();
}

[Test]
[Category( name: "Negative")]
[AllureSuite("Authorization-UI")]
[AllureName("Authorization using invalid data")]
[AllureTms( name: "tms", url: "suite=9&previewMode=modal&case=20")]
public void Authorization_PopulatedInvalidData_ErrorMessageDisplayed()
{
    const string invalidEmail = "1111@sma.b";
    const string invalidPassword = "11111";

    _welcomingPage // WelcomingPage
        .ProceedToLoggingIn()
        .PopulateAuthorizationData( login: invalidEmail, invalidPassword ) // AuthorizationPage
        .SubmitAuthorizationForm();

    using ( new AssertionScope() )
    {
        _authorizationPage.ErrorMessageDisplayed().Should().BeTrue();
        _authorizationPage.ErrorMessageText().Should().Be("These credentials do not match our records.");
    }
}
```

# API calls for precondition steps

As Test Case can be created only inside of a Project, the test first calls a CreateNewProject and uses the instance of the created Project. After the test execution the created Project is deleted

```
[OneTimeSetUp]
public void PreconditionCreateProject()
{
    var creationProjectResponse = ProjectService.CreateNewProject(_projectToAdd).Result;
    _onSiteProjectCodeAfterCreation = creationProjectResponse.Result.Code;
}
```

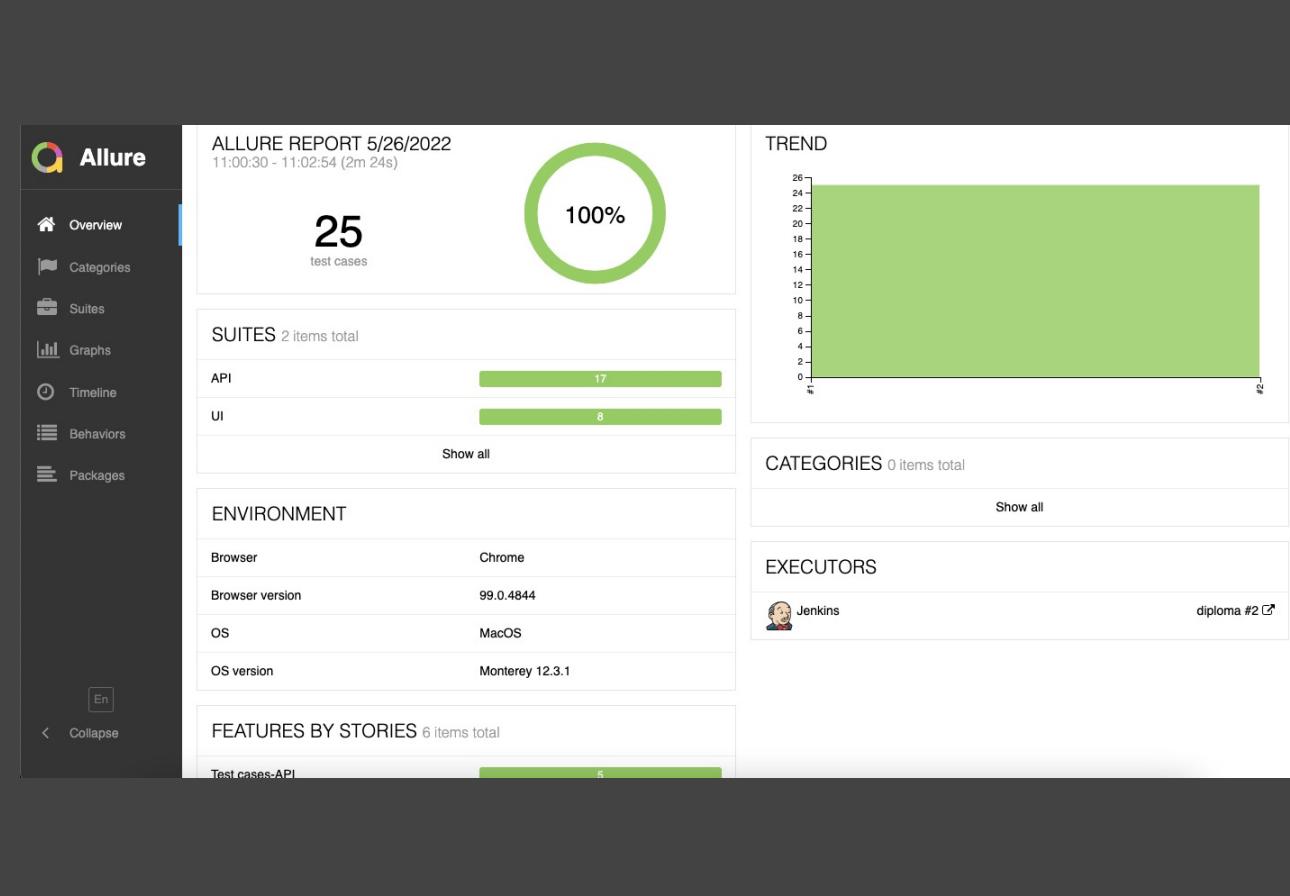
## Clearing of test data after test run

```
[OneTimeTearDown]
public void PostconditionDeleteProject()
{
    ProjectService.DeleteProjectByCode(_onSiteProjectCodeAfterCreation).Wait();
}
```

The same test class. When the execution finished the OneTimeTearDown deletes the created-on-site project making it free of test data

# Readable and understandable allure report

The generated after each test run allure report contains human-like test methods' naming, test environment data, tests' severity, steps executed, parameters taken, tms links



# Readable and understandable allure report

Look of a test

Tags: **Negative** **Authorization-UI**

Severity: blocker

Duration: 0 11s 715ms

**Links**

  tms

**Execution**

**Set up**

- ✓ [OpenBrowserAtWelcomingPage](#) 0s
- ✓ [InstantiateRequiredPages](#) 0s

**Test body**

- ✓ [Authorization\\_PopulatedInvalidData\\_ErrorMessageDisplayed](#) 11s 715ms
  - 3 sub-steps, 1 attachment
  - ✓ Click "Login" button 3s 061ms
  - > Populate authorization data with: login 1111@sma.b password 11111 184ms
    - 2 parameters
  - ✓ Submit authorization form 728ms
    - > [Console Output](#) 196 B

[Authorization\\_PopulatedInvalidData\\_ErrorMessageDisplayed](#)

**Tear down**

- ✓ [QuitBrowser](#) 0s

# Setup Jenkins job to trigger automation test run at 11 AM each day

**Build Triggers**

Trigger builds remotely (e.g., from scripts) ?

Build after other projects are built ?

Build periodically ?

Schedule ?

```
TZ=Europe/Minsk  
0 11 * * *
```

**⚠ Spread load evenly by using 'H 11 \* \* \*' rather than '0 11 \* \* \*'**

Would last have run at Saturday, May 28, 2022 at 11:00:32 AM Moscow Standard Time; would next run at Sunday, May 29, 2022 at 11:00:32 AM Moscow Standard Time.

GitHub hook trigger for GITScm polling ?

Poll SCM ?

At the end of each run allure report should be generated

The screenshot shows a "Build History" interface with a dark header and body. The header includes a blue icon, the text "Build History", and a "trend" button with an upward arrow. Below the header is a search bar with the placeholder "Filter builds...". The main area displays a list of five builds, each with a green checkmark icon and a yellow square icon:

Build ID	Date	Status
#5	May 28, 2022, 11:00 AM	Yellow Square
#4	May 27, 2022, 11:00 AM	Yellow Square
#3	May 26, 2022, 3:43 PM	Yellow Square
#2	May 26, 2022, 11:00 AM	Yellow Square
#1	May 25, 2022, 10:20 PM	Yellow Square

# Patterns: Factory

```
public class DriverFactory
{
    private static IWebDriver _driver = null;

    [B 8 usages]
    public static IWebDriver Driver
    {
        get
        {
            if(_driver == null)
                throw new NullReferenceException(message:"The WebDriver browser instance was not initialized. You sh
            return _driver;
        }

        private set => _driver = value;
    }

    [B 2 usages]
    public static void InitBrowser()
    {
        Driver = Configurator.AppSettings.BrowserType switch
        {
            "chrome" => DriverSetUp.GetChromeDriver(),
            "firefox" => DriverSetUp.GetFirefoxDriver(),
            _ => throw new ArgumentException(message:"Check that your BrowserType property in appsettings.json is se
        };

        Driver.Manage().Window.Maximize();
        Driver.Manage().Cookies.DeleteAllCookies();
        Driver.Manage().Timeouts().ImplicitWait = TimeSpan.FromSeconds(0);
    }
}
```

## DriverSetUp class

```
public static IWebDriver GetChromeDriver()
{
    var chromeOptions = new ChromeOptions();

    chromeOptions.AddArguments("--disable-gpu");
    chromeOptions.AddArguments("--disable-extensions");

    chromeOptions.SetLoggingPreference(LogType.Browser, LogLevel.All);
    chromeOptions.SetLoggingPreference(LogType.Driver, LogLevel.All);

    new DriverManager().SetUpDriver(new ChromeConfig());
    return new ChromeDriver(chromeOptions);
}

[B 1 usage]
public static IWebDriver GetFirefoxDriver()
{
    var mimeTypesString =
        "image/png,image/gif,image/jpeg,image/pjpeg,application/pdf,text/csv,application/vnd.ms-excel," +
        "application/vnd.openxmlformats-officedocument.spreadsheetml.sheet" +
        "application/vnd.openxmlformats-officedocument.wordprocessingml.document";

    var ffOptions = new FirefoxOptions();
    var ffProfile = new FirefoxProfile();

    ffProfile.SetPreference(name: "browser.download.folderList", value: 2);
    ffProfile.SetPreference(name: "browser.helperApps.neverAsk.saveToDisk", value: mimeTypes);
    ffProfile.SetPreference(name: "browser.helperApps.neverAsk.openFile", value: mimeTypes);
    ffOptions.Profile = ffProfile;

    ffOptions.SetLoggingPreference(LogType.Browser, LogLevel.All);
    ffOptions.SetLoggingPreference(LogType.Driver, LogLevel.All);

    new DriverManager().SetUpDriver(new FirefoxConfig());
    return new FirefoxDriver(ffOptions);
}
```

# Patterns: Factory

## Usage:

```
[SetUp]
public void OpenBrowserAtWelcomingPage()
{
    DriverFactory.InitBrowser();
    Driver = DriverFactory.Driver;

    DriverFactory.Driver.Navigate().GoToUrl(Configurator.AppSettings.BaseUiUrl);
}
```

# Patterns: Chain of invocations

```
[AllureStep(name: "Click \"Login\" button")]
[ 4 usages]
public AuthorizationPage ProceedToLoggingIn()
{
    LoginButton.Click();

    return new AuthorizationPage(Driver);
}
```

```
[AllureStep(name: "Populate authorization data with: login {0} password {1}")]
[ 4 usages]
public AuthorizationPage PopulateAuthorizationData(string login, string password)
{
    EmailField.SendKeys(login);
    PasswordField.SendKeys(password);

    return this;
}

[AllureStep(name: "Submit authorization form")]
[ 4 usages]
public void SubmitAuthorizationForm()
{
    LoginButton.Click();
}
```

## Usage:

```
[Test]
[Category(name: "Positive")]
[AllureSuite("Authorization-UI")]
[AllureName("Authorization using valid data")]
[AllureTms(name: "tms", url: "suite=9&previewMode=modal&case=20")]
public void Authorization_PopulatedValidData_ProjectsPageOpened()
{
    _welcomingPage // WelcomingPage
        .ProceedToLoggingIn()
        .PopulateAuthorizationData(login: Configurator.Admin.Email, Configurator.Admin.Password)
        .SubmitAuthorizationForm();

    _projectsPage.PageOpened.Should().BeTrue();
}
```

# Patterns: Decorator

```
public class Table
{
    private readonly BaseElementWrapper _baseElementWrapper;

    ☐ 1 usage
    private ReadOnlyCollection<IWebElement> Headers => _baseElementWrapper.FindElements(By.CssSelector("thead th"));

    ☐ 1 usage
    private ReadOnlyCollection<IWebElement> Rows => _baseElementWrapper.FindElements(By.CssSelector("tbody tr"));

    ☐ 1 usage
    private ReadOnlyCollection<IWebElement> Cells(IWebElement row) => row.FindElements(By.TagName("td"));

    ☐ 2 usages
    public Table(IWebDriver driver, By @by)
    {
        _baseElementWrapper = new BaseElementWrapper(driver, by);
    }

    ☐ 1 usage
    public bool ProjectExists(string pageTitle)
    {
        const string projectNameColumnHeader = "Project name";

        var pageTitleColumnIndex = Headers.TakeWhile(header:IWebElement => !header.Text.Normalize().Equals(projectNameColumnHeader)).Count();

        foreach (var row:IWebElement in Rows)
        {
            var cells:ReadOnlyCollection<IWebElement> = Cells(row);
            var projectNameElement = cells[pageTitleColumnIndex].FindElement(By.CssSelector("a.defect-title"));

            if (projectNameElement.Text == pageTitle)
            {
                return true;
            }
        }
    }

    return false;
}
```

```
public class BaseElementWrapper : IWebElement
{
    private readonly WaitService _waitService;
    private readonly IWebElement _webElement;

    public string TagName => _webElement.TagName;

    public string Text => _webElement.Text;

    public bool Enabled => _webElement.Enabled;

    public bool Selected => _webElement.Selected;

    public Point Location => _webElement.Location;

    public Size Size => _webElement.Size;

    public bool Displayed => _webElement.Displayed;

    ☐ 1 usage
    public BaseElementWrapper(IWebDriver driver, By locator)
    {
        _waitService = new WaitService(driver);
        _webElement = _waitService.WaitUntilElementExists(locator);
    }
}
```

# Patterns: Decorator

## Usage:

```
[Test]
[Order(3)]
[AllureName("Delete the updated project")]
[AllureTms(name:"tms", url:"suite=6&previewMode=modal&case=13")]
public void DeleteProject_ConfirmDeletion_ProjectIsDeleted()
{
    _projectSettingsPage// ProjectSettingsPage
        .DeleteProject()// DeleteProjectPage
        .ConfirmDeletion();

    _projectsPage.ProjectExistsInTable(_projectToUpdateWith.Title).Should().BeFalse();
}
```

# Patterns: Mediator

```
public class LoginStep
{
    private readonly IWebDriver _webDriver;
    private readonly WelcomingPage _welcomingPage;
    private readonly AuthorizationPage _authorizationPage;

    [1 usage]
    public LoginStep(IWebDriver webDriver)
    {
        _webDriver = webDriver;
        _welcomingPage = new WelcomingPage(_webDriver);
        _authorizationPage = new AuthorizationPage(_webDriver);
    }

    [AllureStep(name: "Log in with {0} and {1}")]
    [1 usage]
    public ProjectsPage LogIn(string email, string password)
    {
        _welcomingPage
            .ProceedToLoggingIn();

        _authorizationPage
            .PopulateAuthorizationData(login: email, password) // AuthorizationPage
            .SubmitAuthorizationForm();

        return new ProjectsPage(_webDriver);
    }
}
```

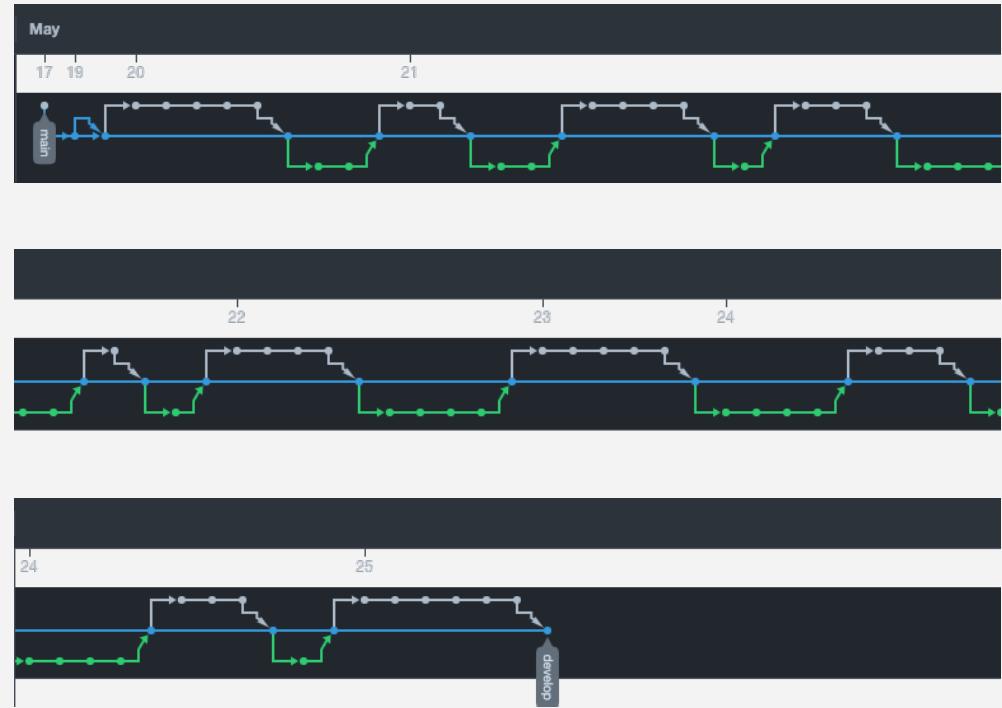
## Usage:

```
[Test]
[Order(1)]
[AllureName("Create a project with requited fields filled")]
[AllureTms(name: "tms", url: "suite=6&previewMode=modal&case=11")]
public void CreateProject_PopulateProjectForm_ProjectIsCreated()
{
    _loginStep // LoginStep
        .LogIn(Configurator.Admin.Email, Configurator.Admin.Password) // ProjectsPage
        .ClickCreateNewProjectButton()
        .PopulateProjectData(_projectToAdd) // CreateProjectPage
        .SubmitProjectForm();

    using (new AssertionScope())
    {
        _projectPage.PageOpened.Should().BeTrue();
        _projectPage.ProjectTitleText().Should().Be(_projectToAdd.Title);
    }
}
```

# Git strategy

We were supposed to have the following branches : main and develop coming off main. Out of the develop we had to create feature branches. When a feature development would be done we had to merge the feature branch into develop and delete the feature branch.



# Git strategy

Naming conventions:

branches: feature/case—id-new-awesome-feature  
commits: CASE-ID: Title starting with a capital letter

Merge pull request #22 from daverbk/feature/Diploma-22-add-correct-er...	...	Verified		351718e	
daverbk committed 4 days ago	Diploma-22: Added authorization error message correctness check		2ca5bfe		
Merge pull request #21 from daverbk/feature/no—case-remove-unused-code	...	Verified		7fbfa37	
daverbk committed 4 days ago	NO-CASE: Removed unused "User" type in UserTypes enum		4960270		
NO-CASE: Removed unused search for all appsettings.*.json files in Co...	...		83d1c4f		
daverbk committed 4 days ago	Merge pull request #20 from daverbk/feature/no—case-separate-api-crud...	Verified		43e65bc	
daverbk committed 4 days ago	NO-CASE: Separated the tms links for api crud tests		2345aee		



03

Framework's  
pros and cons

## Pros

## Cons

### Simplicity

Easy to understand  
structure

### Stability

Didn't crush ones during  
the runs

### Overused chain of invocations

Page method refers to  
better be seen

### Not paralleled

Runs are going one after  
another, which makes them  
a bit slow



04

# Troubleshooting

# Problem

API response represents an object that has two properties: status and result. In its turn the result property can be of any type: test case, defect, etc. The same refers to group selections which are returned from get-all methods.

**Response** 200

A Project.

**RESPONSE BODY**

```
object
status boolean
result object
  title string
  code string
  counts object
    COUNTS OBJECT
```

+

Close

**Response** 200

**RESPONSE BODY**

```
object
status boolean
result object
  total integer
  filtered integer
  count integer
  entities array of objects
    object
      title string
      code string
      counts object
```

Close

# Solution

Using generics to avoid duplicates

3 usages

```
public ProjectService(RestClientExtended restClient){...}
```

3 usages

```
public Task<Response<Project>> CreateNewProject(Project project){...}
```

1 usage

```
public Task<Response<Project>> GetProjectByCode(string projectCode){...}
```

4 usages

```
public Task<Response<GroupSelection<Project>>> GetAllProjects(){...}
```

3 usages

```
public Task<Response<Project>> DeleteProjectByCode(string projectCode){...}
```

# Problem

Projects table only exists when there is at least one project created in workspace. It makes the testing of project deletion from table a bit complicated because we can't know for sure if it would be the last project.

# Solution

In ProjectExistsInTable method check if the table itself exists. If the table doesn't exist then the project too.

```
public bool ProjectExistsInTable(string projectName)
{
    if (ProjectsTableExists() == false)
    {
        return false;
    }

    return ProjectsTable.ProjectExists(projectName);
}

[ 1 usage ]
private bool ProjectsTableExists()
{
    var tableFoundIndicator :ReadOnlyCollection<IWebElement>? = Driver.FindElements(By.TagName("table"));

    return tableFoundIndicator.Count != 0;
}
```



# 05

# Summary

# Main stages

Reporting bugs

Executing tests with  
the help of Jenkins

Building the test  
framework

Test cases creation

Upcoming testing  
analysis

# Reporting bugs

## Defects

Create new defect Search for defects Status: Open + Add filter Export

Defect	Author	Assignee	Severity	Milestone	External
<p>! [Defects] - Sending an api request "Delete defect" via Patch method causes Server Error (500) D-7 · created 1 week ago</p>	 Dave	Unassigned	Normal	Release 1.0	...
<p>! [Test Cases] - Sending an api request "Update a test case" without body causes Server Error (500) D-6 · created 1 week ago</p>	 Dave	Unassigned	Normal	Release 1.0	...
<p>! [Projects] - "Requirements" section is unresponsive D-3 · created 1 week ago</p>	 Dave	Unassigned	Normal	Release 1.0	...
<p>! [Groups] - Grammar mistake in no-groups-found-like message D-1 · created 1 week ago</p>	 Dave	Unassigned	Trivial	Release 1.0	...

06

Working  
in team

**Had different projects, but discussed  
occurred problems together**



Daniil  
Borisevich  
[@PHYSI6](https://twitter.com/@PHYSI6)



Dmitry Laevsky  
[@DimaLayeuski](https://twitter.com/@DimaLayeuski)



Thanks

NEW YORK