PORTFOLIO

Sigrid Laas-Hurt, Junior Software Tester

2024

Table of Contents

Project#1 Restaurant Food Ordering application

Project overview

The customer has decided to build a food ordering application that will allow contactless ordering in the restaurant (replacing the physical menu). The app must provide a customer with an overview of all the food and drink available in the restaurant, accept orders, and securely store customer data to comply with GDPR and data privacy laws.

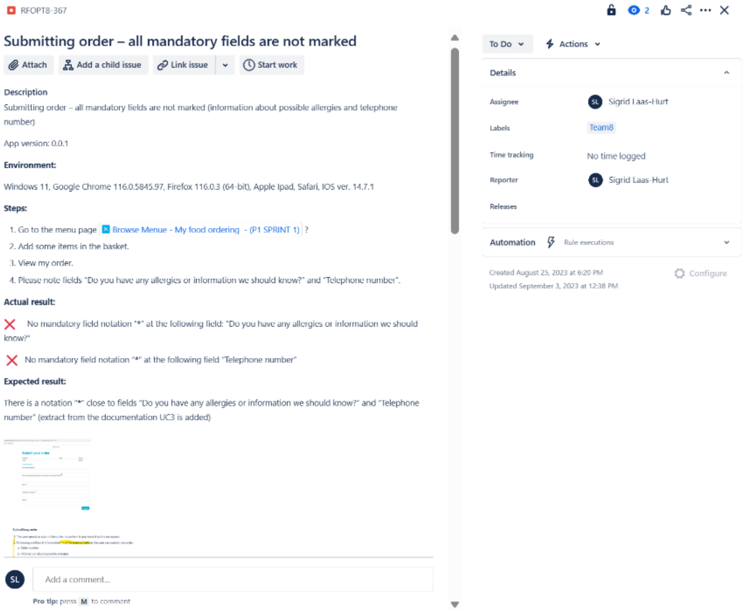
The main functionalities of this app are:

- Contactless ordering for customers to keep staff safe
- No download is required, just scan a QR code or enter a web address
- Browse the menu, and order food and drinks
- Simple product management
- Contact Tracing securely records customer data and meets privacy laws.

My responsibilities include

1. Worked in an agile team and followed Scrum routines.
2. Acted as the coordinator for Spint#1.
3. Utilized various testing techniques such as functional testing, use case testing, checklist-based testing, and cross-browser testing.
4. Wrote test documentation in **Confluence**, including checklists for use cases and final report.
5. Reported bugs in **Jira**. Our team has found 35 issues.

Example of the reported bug:



RFOPT8-367

Submitting order – all mandatory fields are not marked

Description

Submitting order – all mandatory fields are not marked (information about possible allergies and telephone number)

App version: 0.0.1

Environment:

Windows 11, Google Chrome 116.0.5845.97, Firefox 116.0.3 (64-bit), Apple Ipad, Safari, IOS ver. 14.7.1

Steps:

1. Go to the menu page Browse Menue - My food ordering - (P1 SPRINT 1) ?
2. Add some items in the basket.
3. View my order.
4. Please note fields "Do you have any allergies or information we should know?" and "Telephone number".

Actual result:

☒ No mandatory field notation "*" at the following field: "Do you have any allergies or information we should know?"

☒ No mandatory field notation "*" at the following field "Telephone number"

Expected result:

There is a notation "*" close to fields "Do you have any allergies or information we should know?" and "Telephone number" (extract from the documentation UC3 is added)

Details	
Assignee	SL Sigrid Laas-Hurt
Labels	Team8
Time tracking	No time logged
Reporter	SL Sigrid Laas-Hurt
Releases	

Automation Rule executions

Created August 25, 2023 at 6:20 PM
Updated September 3, 2023 at 12:38 PM

Project#2 Company Collaboration Platform

Project overview

The large energy company decided to build an application, that will increase employee productivity and internal communication. The company has more than 3000 employees, that are distributed between tens of departments and several offices. The pandemic highlighted the need for an up to date, cross-device solution for reaching the entire organization. In the increasingly hybrid world, employees need to be empowered to make decisions and communicate openly with peers.

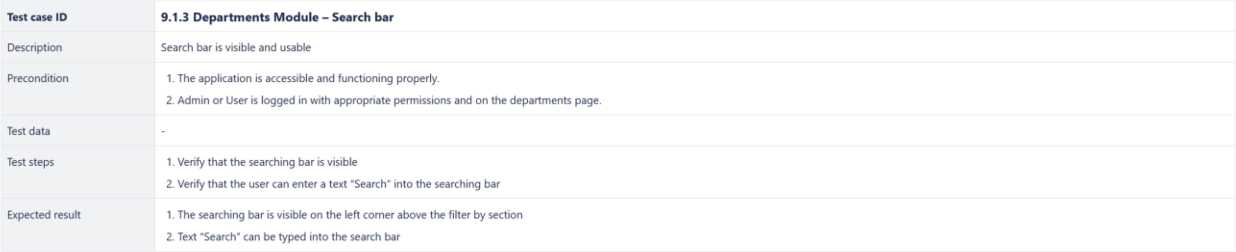
The company decided to build an application that will allow employees to:

1. Have visibility over the company structure with an up to date organization chart
2. Build stronger connections with colleagues by having easy access to their interests, experience, skills and more
3. Be more productive by enabling them to quickly browse through facility and meeting room availability and office locations
4. Collaborate efficiently by creating custom contact lists for projects, events, social activities
5. Create and update their user status, such as sick leave, vacation, maternity leave

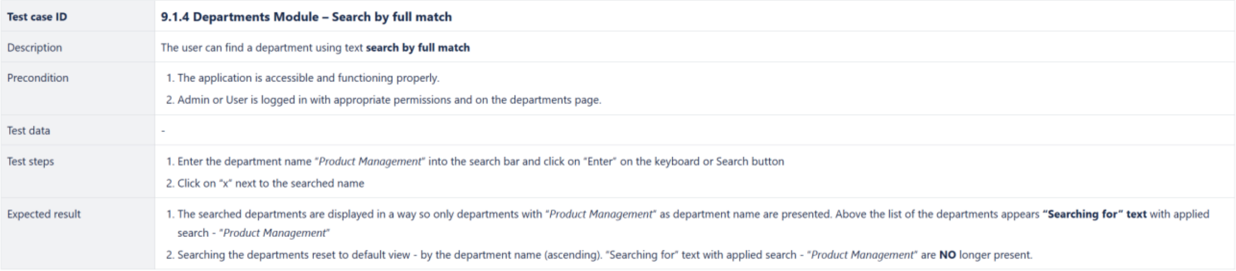
My responsibilities include

1. Discovered existing functionalities and improved testing documentation
2. Developed new functionalities to the system that extended system capabilities
3. Worked in an agile team and followed Scrum routines.
4. Wrote test documentation in **Confluence**, including test cases and final report.
5. Utilized various testing techniques such as exploratory testing, functional testing, checklist-based testing, cross-browser testing, and regression testing.
6. Reported bugs in **Jira**. Our team has found 26 issues.
7. Tracked time in **Jira**.

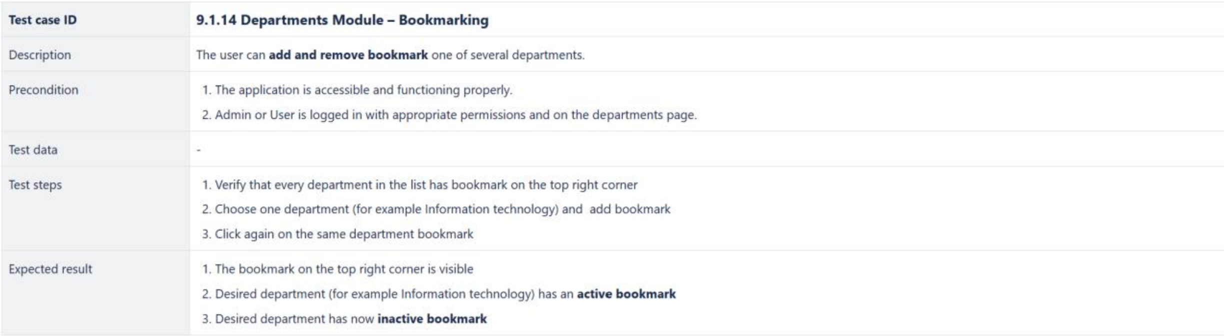
Example of the Test cases:



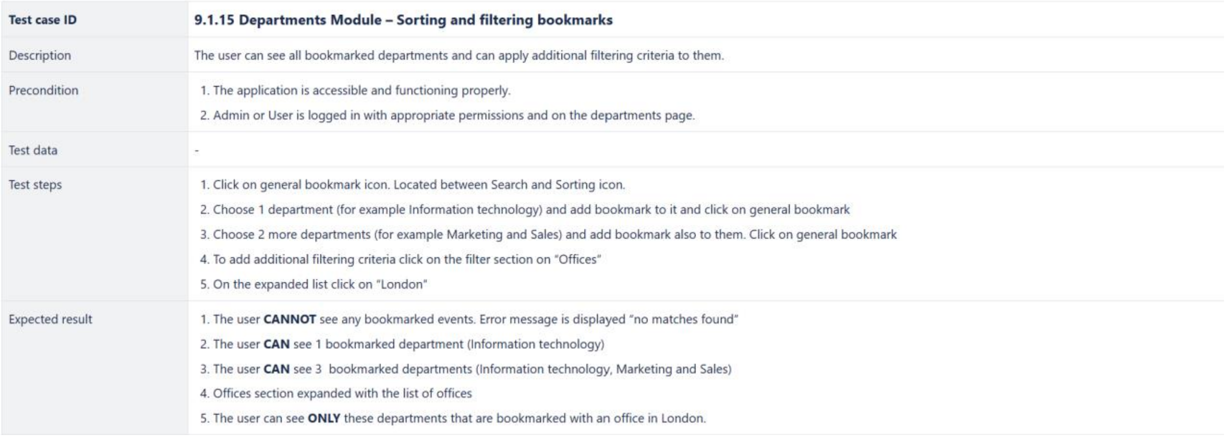
Test case ID	**9.1.3 Departments Module – Search bar**
Description	Search bar is visible and usable
Precondition	1. The application is accessible and functioning properly. 2. Admin or User is logged in with appropriate permissions and on the departments page.
Test data	-
Test steps	1. Verify that the searching bar is visible 2. Verify that the user can enter a text "Search" into the searching bar
Expected result	1. The searching bar is visible on the left corner above the filter by section 2. Text "Search" can be typed into the search bar



Test case ID	**9.1.4 Departments Module – Search by full match**
Description	The user can find a department using text **search by full match**
Precondition	1. The application is accessible and functioning properly. 2. Admin or User is logged in with appropriate permissions and on the departments page.
Test data	-
Test steps	1. Enter the department name "*Product Management*" into the search bar and click on "Enter" on the keyboard or Search button 2. Click on "x" next to the searched name
Expected result	1. The searched departments are displayed in a way so only departments with "*Product Management*" as department name are presented. Above the list of the departments appears **"Searching for" text** with applied search - "*Product Management*" 2. Searching the departments reset to default view - by the department name (ascending). "Searching for" text with applied search - "*Product Management*" are **NO** longer present.

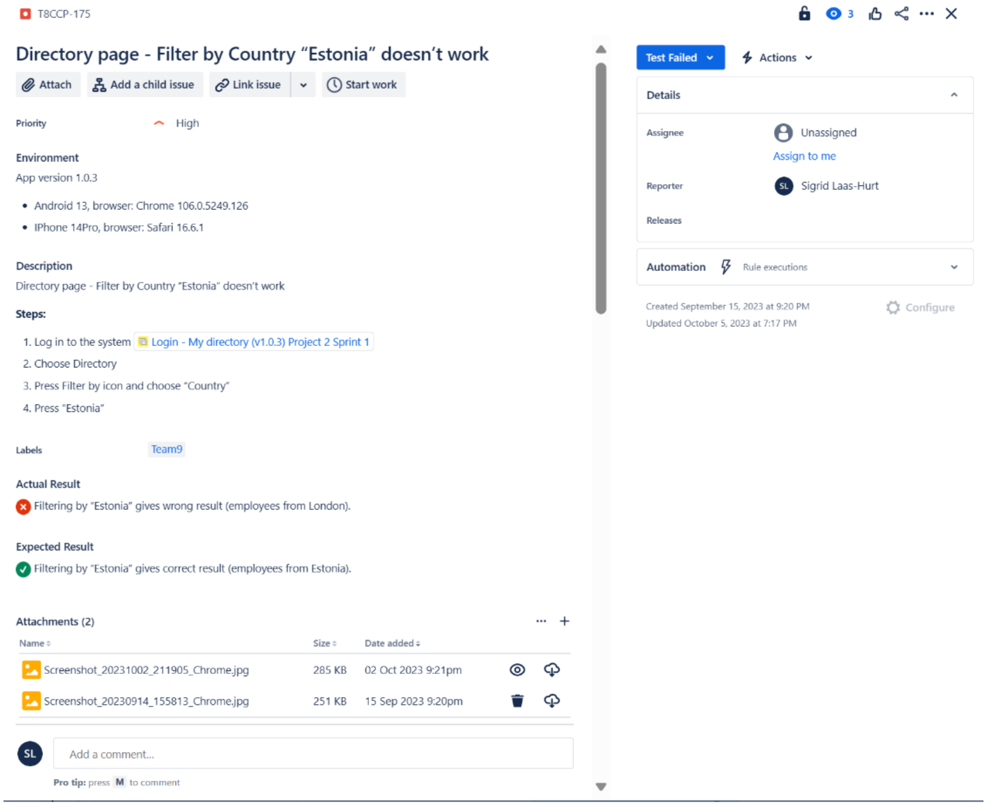


Test case ID	**9.1.14 Departments Module – Bookmarking**
Description	The user can **add and remove bookmark** one of several departments.
Precondition	1. The application is accessible and functioning properly. 2. Admin or User is logged in with appropriate permissions and on the departments page.
Test data	-
Test steps	1. Verify that every department in the list has bookmark on the top right corner 2. Choose one department (for example Information technology) and add bookmark 3. Click again on the same department bookmark
Expected result	1. The bookmark on the top right corner is visible 2. Desired department (for example Information technology) has an **active bookmark** 3. Desired department has now **inactive bookmark**



Test case ID	**9.1.15 Departments Module – Sorting and filtering bookmarks**
Description	The user can see all bookmarked departments and can apply additional filtering criteria to them.
Precondition	1. The application is accessible and functioning properly. 2. Admin or User is logged in with appropriate permissions and on the departments page.
Test data	-
Test steps	1. Click on general bookmark icon. Located between Search and Sorting icon. 2. Choose 1 department (for example Information technology) and add bookmark to it and click on general bookmark 3. Choose 2 more departments (for example Marketing and Sales) and add bookmark also to them. Click on general bookmark 4. To add additional filtering criteria click on the filter section on "Offices" 5. On the expanded list click on "London"
Expected result	1. The user **CANNOT** see any bookmarked events. Error message is displayed "no matches found" 2. The user **CAN** see 1 bookmarked department (Information technology) 3. The user **CAN** see 3 bookmarked departments (Information technology, Marketing and Sales) 4. Offices section expanded with the list of offices 5. The user can see **ONLY** these departments that are bookmarked with an office in London.

Example of the reported bug:



T8CCP-175

Directory page - Filter by Country "Estonia" doesn't work

Priority: High

Environment

App version 1.0.3

- Android 13, browser: Chrome 106.0.5249.126
- IPhone 14Pro, browser: Safari 16.6.1

Description

Directory page - Filter by Country "Estonia" doesn't work

Steps:

1. Log in to the system Login - My directory (v1.0.3) Project 2 Sprint 1
2. Choose Directory
3. Press Filter by icon and choose "Country"
4. Press "Estonia"

Labels: Team9

Actual Result

✗ Filtering by "Estonia" gives wrong result (employees from London).

Expected Result

✓ Filtering by "Estonia" gives correct result (employees from Estonia).

Attachments (2)

Name	Size	Date added
Screenshot_20231002_211905_Chrome.jpg	285 KB	02 Oct 2023 9:21pm
Screenshot_20230914_155813_Chrome.jpg	251 KB	15 Sep 2023 9:20pm

Add a comment...

Pro tip: press M to comment

Details

Assignee: Unassigned
Assign to me

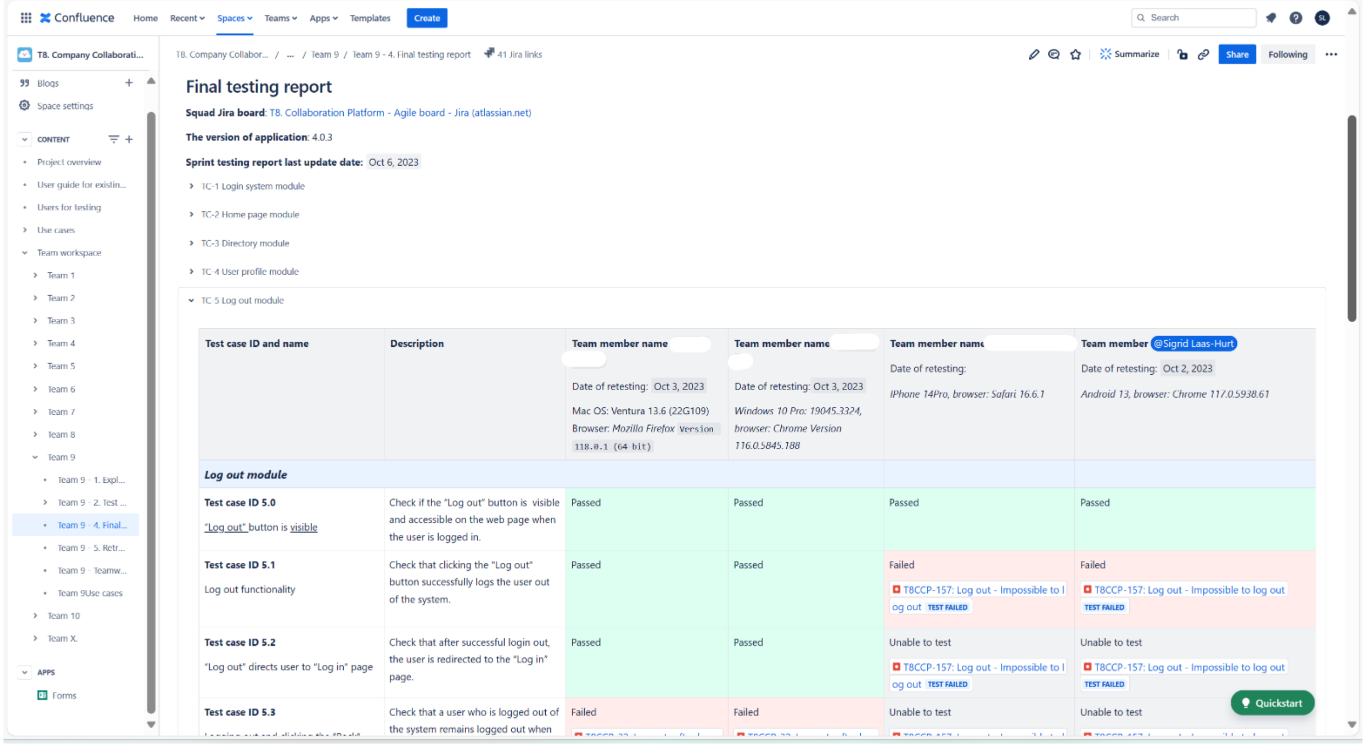
Reporter: Sigrid Laas-Hurt

Releases

Automation — Rule executions

Created September 15, 2023 at 9:20 PM
Updated October 5, 2023 at 7:17 PM

Example of the Final report:



Final testing report

Squad Jira board: T8. Collaboration Platform - Agile board - Jira (atlassian.net)

The version of application: 4.0.3

Sprint testing report last update date: Oct 6, 2023

- TC-1 Login system module
- TC-2 Home page module
- TC-3 Directory module
- TC-4 User profile module
- TC-5 Log out module

Test case ID and name	**Description**	**Team member name** [redacted] [redacted] Date of retesting: Oct 3, 2023 Mac OS: Ventura 13.6 (22G109) Browser: *Mozilla Firefox* `Version 118.0.1 (64-bit)`	**Team member name** [redacted] [redacted] Date of retesting: Oct 3, 2023 *Windows 10 Pro: 19045.3324, browser: Chrome Version 116.0.5845.188*	**Team member name** [redacted] Date of retesting: *IPhone 14Pro, browser: Safari 16.6.1*	**Team member** @Sigrid Laas-Hurt Date of retesting: Oct 2, 2023 *Android 13, browser: Chrome 117.0.5938.61*
Log out module					
Test case ID 5.0 "Log out" button is visible	Check if the "Log out" button is visible and accessible on the web page when the user is logged in.	Passed	Passed	Passed	Passed
Test case ID 5.1 Log out functionality	Check that clicking the "Log out" button successfully logs the user out of the system.	Passed	Passed	Failed T8CCP-157: Log out - Impossible to log out TEST FAILED	Failed T8CCP-157: Log out - Impossible to log out TEST FAILED
Test case ID 5.2 "Log out" directs user to "Log in" page	Check that after successful login out, the user is redirected to the "Log in" page.	Passed	Passed	Unable to test T8CCP-157: Log out - Impossible to log out TEST FAILED	Unable to test T8CCP-157: Log out - Impossible to log out TEST FAILED
Test case ID 5.3	Check that a user who is logged out of the system remains logged out when	Failed	Failed	Unable to test	Unable to test

Project#3 CRM Platform

Project overview

Customer is a digital agency offering innovative marketing and web design solutions. They use custom strategies, eye-catching visuals, and data-driven optimization to make online experiences that are fun and increase the visibility of a company's brand.

They organize their sales and project management with a simple CRM platform. There are 3 main business processes that are supported by the CRM:

- Collection of potential customers from different marketing channels and organising the marketing activities.
- Preparing offers, negotiating terms with potential customers and signing contracts.
- Managing projects with the signed contracts and tracking the progress of projects.

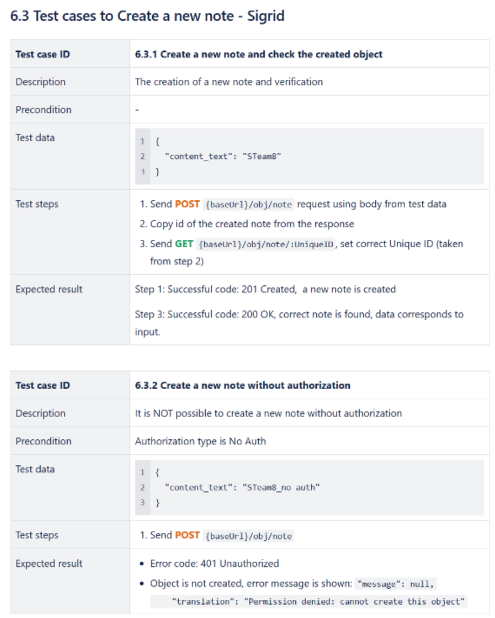
The marketing and sales teams currently use the CRM system on a daily basis. Because the number of manual tasks is continuously going up, our customer wants to start automating some parts of the processes and integrating them with other systems used in the company.

The main goal of this project is to make an API that covers all of the main functions of the CRM platform. The CRM platform lets our customers manage their work and different sales processes, like getting new customers, handling customer requests, making proposals, and closing them. The CRM platform is also used as a database of all customers and their data. The first step for this initiative is to build a comprehensive API for the CRM, that will allow for integrations and implementation automation procedures. All of the system's main tasks must be able to be done through the web services, which means that the API must allow this.

My responsibilities include

1. Worked in an agile team and followed Scrum routines.
2. Acted as the coordinator for Spint#2.
3. Wrote test documentation in **Confluence**, including API testing test cases and final report.
4. Built a stable and well-tested API for the following functionalities of the CRM system:
    a. *Leads management (creation and management of the customer's database);*
    b. *Invitation of new users;*
    c. Funnels management (creation and management of CRM funnels);
    d. *Deals management (creation and management of deals in different funnels*.
5. Utilized various testing techniques such as exploratory testing, API testing, browser testing, and end-to-end testing (example is in separate file Team 8 End-to-end test cases_Sigrid_LH).
6. Created **Postman** collections for testing of API.
7. Reported bugs in **Jira**. Our team has found 65 issues.
8. Tracked time in **Jira**

Example of the Test cases:



6.3 Test cases to Create a new note - Sigrid

Test case ID	**6.3.1 Create a new note and check the created object**
Description	The creation of a new note and verification
Precondition	-

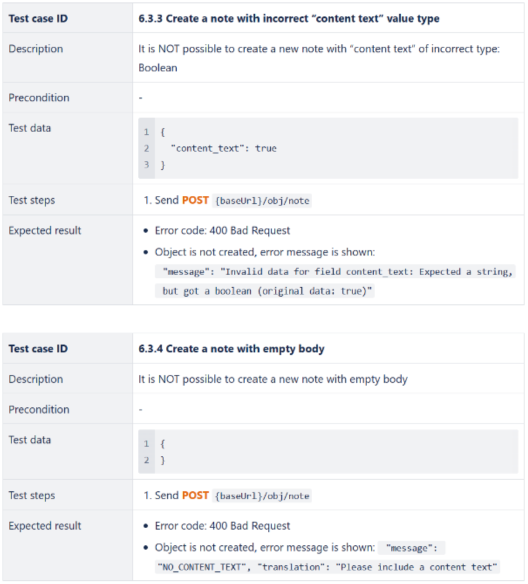
Test data	`{`
	`  "content_text": "STeam8"`
	`}`

Test steps	1. Send **POST** `{baseUrl}/obj/note` request using body from test data 2. Copy id of the created note from the response 3. Send **GET** `{baseUrl}/obj/note/:UniqueID`, set correct Unique ID (taken from step 2)
Expected result	Step 1: Successful code: 201 Created, a new note is created Step 3: Successful code: 200 OK, correct note is found, data corresponds to input.

Test case ID	**6.3.2 Create a new note without authorization**
Description	It is NOT possible to create a new note without authorization
Precondition	Authorization type is No Auth

Test data	`{`
	`  "content_text": "STeam8_no auth"`
	`}`

Test steps	1. Send **POST** `{baseUrl}/obj/note`
Expected result	• Error code: 401 Unauthorized • Object is not created, error message is shown: `"message": null, "translation": "Permission denied: cannot create this object"`



Test case ID	**6.3.3 Create a note with incorrect "content text" value type**
Description	It is NOT possible to create a new note with "content text" of incorrect type: Boolean
Precondition	-

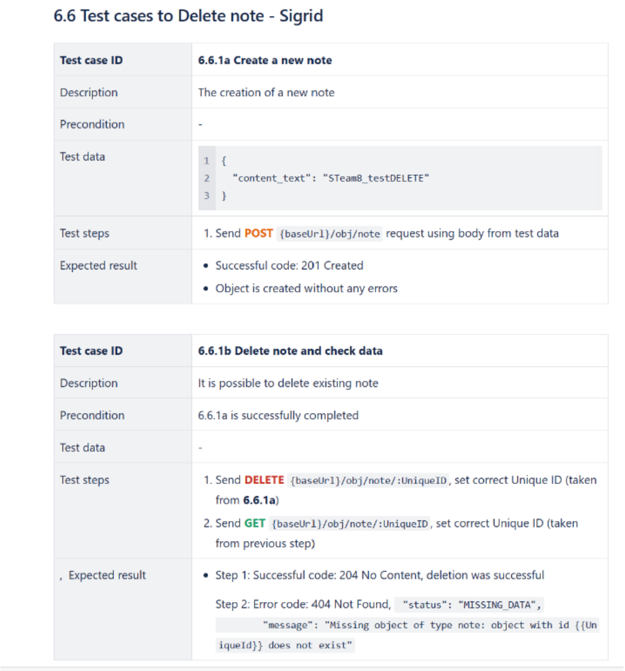
Test data	`{`
	`  "content_text": true`
	`}`

Test steps	1. Send **POST** `{baseUrl}/obj/note`
Expected result	• Error code: 400 Bad Request • Object is not created, error message is shown: `"message": "Invalid data for field content_text: Expected a string, but got a boolean (original data: true)"`

Test case ID	**6.3.4 Create a note with empty body**
Description	It is NOT possible to create a new note with empty body
Precondition	-

Test data	`{`
	`}`

Test steps	1. Send **POST** `{baseUrl}/obj/note`
Expected result	• Error code: 400 Bad Request • Object is not created, error message is shown: `"message": "NO_CONTENT_TEXT", "translation": "Please include a content text"`



6.6 Test cases to Delete note - Sigrid

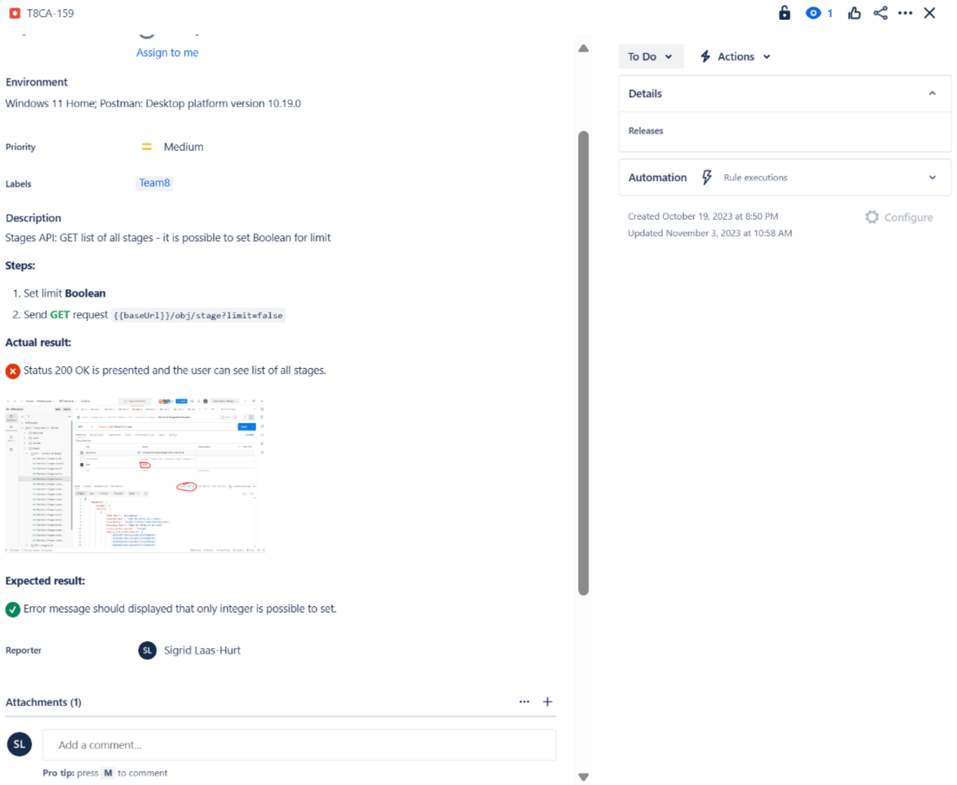
Test case ID	**6.6.1a Create a new note**
Description	The creation of a new note
Precondition	-

Test data	`{`
	`  "content_text": "STeam8_testDELETE"`
	`}`

Test steps	1. Send **POST** `{baseUrl}/obj/note` request using body from test data
Expected result	• Successful code: 201 Created • Object is created without any errors

Test case ID	**6.6.1b Delete note and check data**
Description	It is possible to delete existing note
Precondition	6.6.1a is successfully completed
Test data	-
Test steps	1. Send **DELETE** `{baseUrl}/obj/note/:UniqueID`, set correct Unique ID (taken from **6.6.1a**) 2. Send **GET** `{baseUrl}/obj/note/:UniqueID`, set correct Unique ID (taken from previous step)
, Expected result	• Step 1: Successful code: 204 No Content, deletion was successful Step 2: Error code: 404 Not Found, `"status": "MISSING_DATA", "message": "Missing object of type note: object with id {{UniqueId}} does not exist"`

Example of the reported bug:



T8CA-159

Assign to me

To Do

Actions

Details

Releases

Automation

Rule executions

Created October 19, 2023 at 8:50 PM
Updated November 3, 2023 at 10:58 AM

Configure

Environment

Windows 11 Home; Postman: Desktop platform version 10.19.0

Priority Medium

Labels Team8

Description

Stages API: GET list of all stages - it is possible to set Boolean for limit

Steps:

1. Set limit **Boolean**
2. Send **GET** request `{{baseUrl}}/obj/stage?limit=false`

Actual result:

✗ Status 200 OK is presented and the user can see list of all stages.

Expected result:

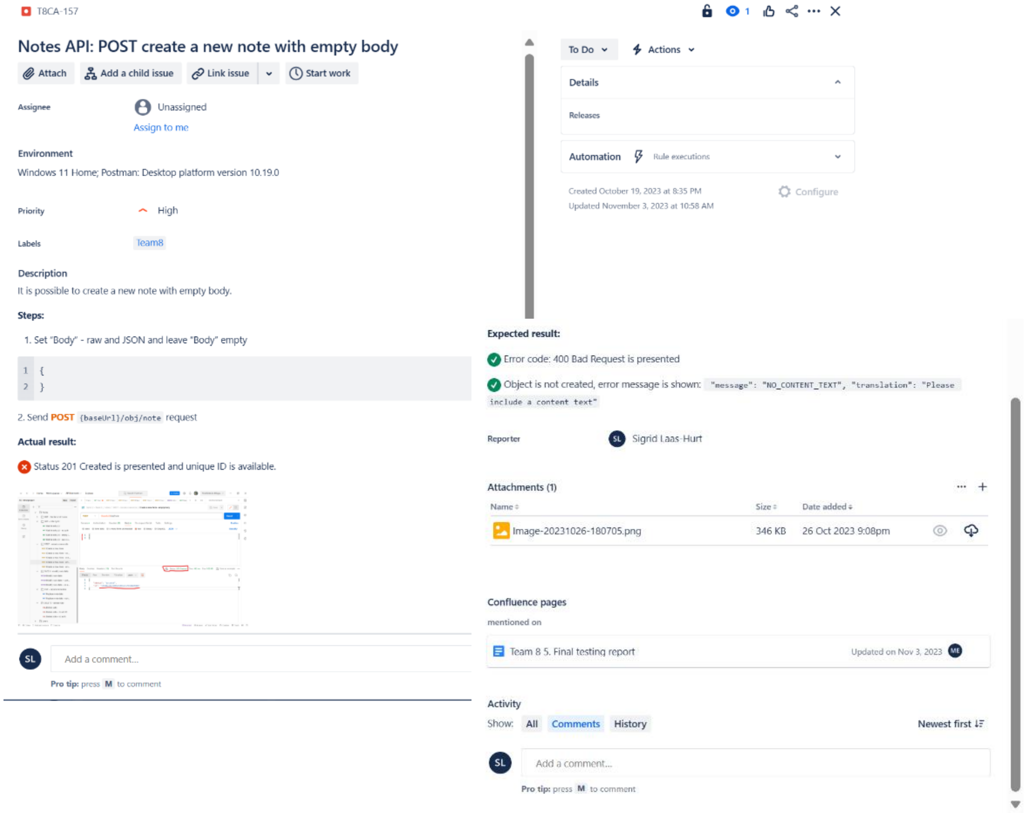
✓ Error message should displayed that only integer is possible to set.

Reporter SL Sigrid Laas-Hurt

Attachments (1)

SL Add a comment...

Pro tip: press M to comment



T8CA-157

Notes API: POST create a new note with empty body

Attach | Add a child issue | Link issue | Start work

To Do

Actions

Details

Releases

Automation

Rule executions

Created October 19, 2023 at 8:35 PM
Updated November 3, 2023 at 10:58 AM

Configure

Assignee Unassigned

Assign to me

Environment

Windows 11 Home; Postman: Desktop platform version 10.19.0

Priority High

Labels Team8

Description

It is possible to create a new note with empty body.

Steps:

1. Set "Body" - raw and JSON and leave "Body" empty

```
{
}
```

2. Send **POST** `{baseUrl}/obj/note` request

Actual result:

✗ Status 201 Created is presented and unique ID is available.

SL Add a comment...

Pro tip: press M to comment

Expected result:

✓ Error code: 400 Bad Request is presented

✓ Object is not created, error message is shown: `"message": "NO_CONTENT_TEXT", "translation": "Please include a content text"`

Reporter SL Sigrid Laas-Hurt

Attachments (1)

Name	Size	Date added
Image-20231026-180705.png	346 KB	26 Oct 2023 9:08pm

Confluence pages

mentioned on

Team 8 5. Final testing report Updated on Nov 3, 2023 ME

Activity

Show: All Comments History Newest first

SL Add a comment...

Pro tip: press M to comment

Project#4 Registration Forms

Project overview

We have a project with different registration forms (HTML, CSS, and JavaScript stack is used). Using those registration forms we will not only cover the basic functionality of Cypress but also make an introduction to the main parts of front-end architecture. Registration forms allow people to enter their data for registering them as users on the Web page.

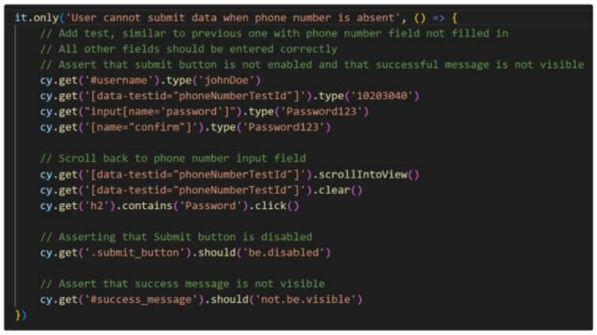
The registration forms are partially covered by the automation tests, which ensure that forms are working and all workflows and elements on the forms are working, though some parts are not tested at all. Also, some parts of the automated tests for the second form intentionally contain some bugs, so automated tests are not working properly and need to be fixed.

My responsibilities include

1. Worked individually.
2. Created UI automation tests using **Cypress**.
3. Modified the code using the **Visual Studio Code**
4. **Git** workflow. Code is committed to the **Github** repository
5. Tracked time in **Jira**.

Example of the test:

Add a new test when the phone number is not filled in: Add content to the empty test 'User cannot submit data when phone number is absent'

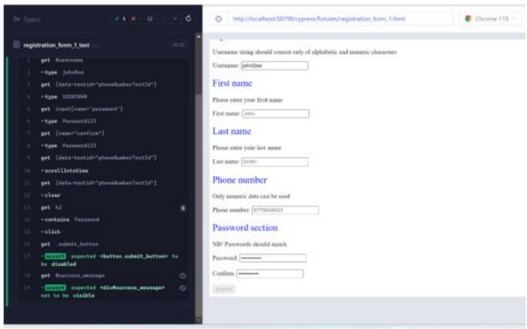


```
it.only('User cannot submit data when phone number is absent', () => {
    // Add test, similar to previous one with phone number field not filled in
    // All other fields should be entered correctly
    // Assert that submit button is not enabled and that successful message is not visible
    cy.get('#username').type('johnDoe')
    cy.get('[data-testid="phoneNumberTestId"]').type('10203040')
    cy.get("input[name='password']").type('Password123')
    cy.get('[name="confirm"]').type('Password123')

    // Scroll back to phone number input field
    cy.get('[data-testid="phoneNumberTestId"]').scrollIntoView()
    cy.get('[data-testid="phoneNumberTestId"]').clear()
    cy.get('h2').contains('Password').click()

    // Asserting that Submit button is disabled
    cy.get('.submit_button').should('be.disabled')

    // Assert that success message is not visible
    cy.get('#success_message').should('not.be.visible')
})
```



Specs
✓ 1 ✗
http://localhost:58798/cypress/fixtures/registration_form_1.html
Chrome 119

registration_form_1_test

```
get #username
-type johnDoe
get [data-testid="phoneNumberTestId"]
-type 10203040
get input[name='password']
-type Password123
get [name="confirm"]
-type Password123
get [data-testid="phoneNumberTestId"]
-scrollIntoView
get [data-testid="phoneNumberTestId"]
-clear
get h2
-contains Password
-click
get .submit_button
-assert expected <button.submit_button> to be disabled
get #success_message
-assert expected <div#success_message> not to be visible
```

Username string should consist only of alphabetic and numeric characters

Username: johnDoe

First name

Please enter your first name

First name: John

Last name

Please enter your last name

Last name: Smith

Phone number

Only numeric data can be used

Phone number:

Password section

NB! Passwords should match

Password: ••••••••••

Confirm: ••••••••••

Submit

Summary of results is presented in file Project4_outcomes-Sigrid Laas-Hurt

Github repository: [Automation-Project-1/e2e at master · SigridLH/Automation-Project-1 (github.com)]

Project#5 Task Management Platform

Project overview

The goal of the project is to automate UI testing of the Jira Clone application. The Jira Clone is a lightweight ReactJS application that copies functionalities from the popular Jira task management system. The Jira Clone allows users to manage tasks for the project using a simple Kanban board. The goal of this project will be to cover as many functionalities as possible in the UI tests using the Cypress.

My responsibilities include

1. Worked individually.
2. Created UI automation tests using **Cypress** for the following functionalities
   a. Creation, editing and deleting of tasks;
   b. Creation, editing and deleting of task comments;
   c. Time tracking functionality;
   d. Search functionality
3. Used the tool **Visual Studio Code**
4. Page Object Model approach
5. **Git** workflow. Code is committed to the **Github** repository
6. Tracked time in **Jira**.

Summary of results is presented in file Project5_outcomes-Sigrid Laas-Hurt

Github repository: Automation-Project-2/cypress/e2e at master · SigridLH/Automation-Project-2 (github.com)

Project#6 Work Management Tool

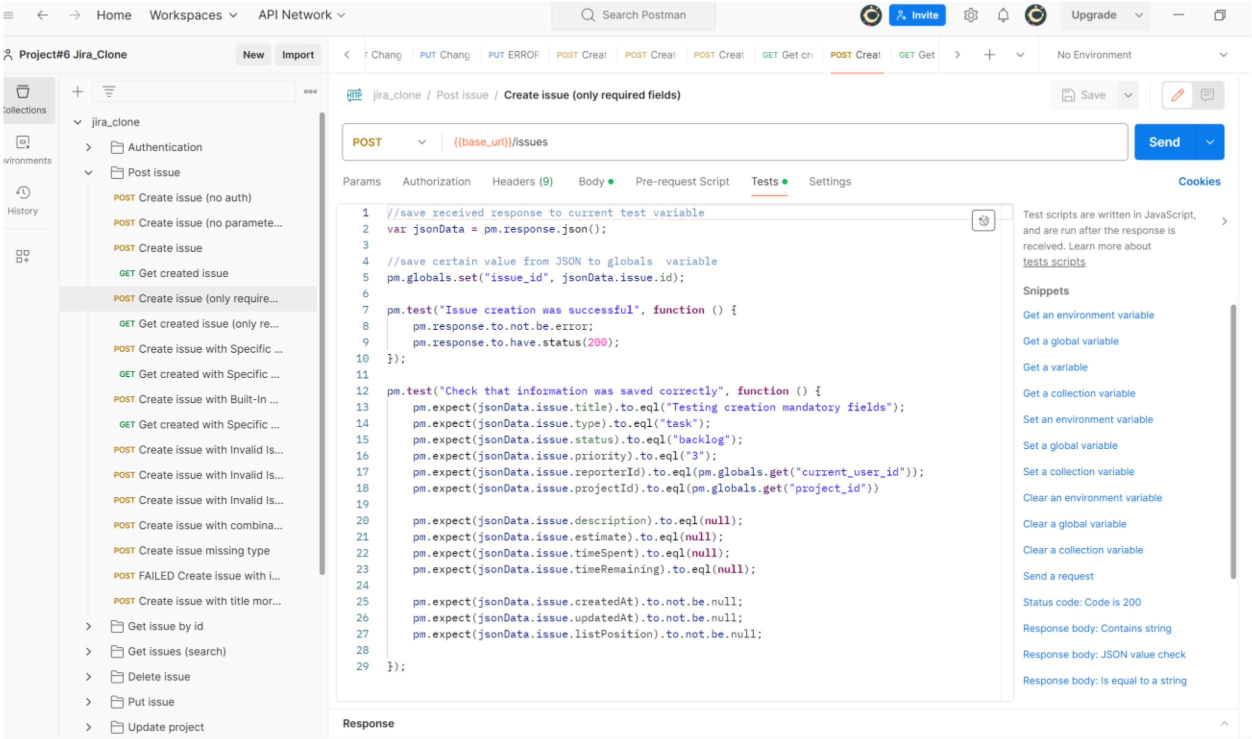
Project overview

The goal of the project is to automate API testing of the Jira Clone application.

My responsibilities include

1. Worked individually.
2. Created Postman collections and wrote backend automation tests with **Postman** for the following functionalities
   a. Creating and updating tasks
   b. Deleting tasks
   c. Time tracking functionality – fir logging time and adding time estimation
   d. Creating, editing and deleting task comments
3. Tracked time in **Jira**.

Example:


Home
Workspaces
API Network
Search Postman
Invite
Upgrade
Project#6 Jira_Clone
New
Import
Chang
PUT Chang
PUT ERROR
POST Creat
POST Creat
POST Creat
GET Get cr
POST Creat
GET Get
No Environment
Collections
Environments
History
jira_clone
Authentication
Post issue
POST Create issue (no auth)
POST Create issue (no paramete...
POST Create issue
GET Get created issue
POST Create issue (only require...
GET Get created issue (only re...
POST Create issue with Specific ...
GET Get created with Specific ...
POST Create issue with Built-In ...
GET Get created with Specific ...
POST Create issue with Invalid Is...
POST Create issue with Invalid Is...
POST Create issue with Invalid Is...
POST Create issue with combina...
POST Create issue missing type
POST FAILED Create issue with i...
POST Create issue with title mor...
Get issue by id
Get issues (search)
Delete issue
Put issue
Update project
jira_clone / Post issue / Create issue (only required fields)
Save
Send
Params
Authorization
Headers (9)
Body
Pre-request Script
Tests
Settings
Cookies
Test scripts are written in JavaScript, and are run after the response is received. Learn more about tests scripts
Snippets
Get an environment variable
Get a global variable
Get a variable
Get a collection variable
Set an environment variable
Set a global variable
Set a collection variable
Clear an environment variable
Clear a global variable
Clear a collection variable
Send a request
Status code: Code is 200
Response body: Contains string
Response body: JSON value check
Response body: Is equal to a string
Response

POST {{base_url}}/issues

```javascript
//save received response to current test variable
var jsonData = pm.response.json();

//save certain value from JSON to globals  variable
pm.globals.set("issue_id", jsonData.issue.id);

pm.test("Issue creation was successful", function () {
    pm.response.to.not.be.error;
    pm.response.to.have.status(200);
});

pm.test("Check that information was saved correctly", function () {
    pm.expect(jsonData.issue.title).to.eql("Testing creation mandatory fields");
    pm.expect(jsonData.issue.type).to.eql("task");
    pm.expect(jsonData.issue.status).to.eql("backlog");
    pm.expect(jsonData.issue.priority).to.eql("3");
    pm.expect(jsonData.issue.reporterId).to.eql(pm.globals.get("current_user_id"));
    pm.expect(jsonData.issue.projectId).to.eql(pm.globals.get("project_id"))

    pm.expect(jsonData.issue.description).to.eql(null);
    pm.expect(jsonData.issue.estimate).to.eql(null);
    pm.expect(jsonData.issue.timeSpent).to.eql(null);
    pm.expect(jsonData.issue.timeRemaining).to.eql(null);

    pm.expect(jsonData.issue.createdAt).to.not.be.null;
    pm.expect(jsonData.issue.updatedAt).to.not.be.null;
    pm.expect(jsonData.issue.listPosition).to.not.be.null;

});
```