THIRD

ECLIPSE TRACTUS-X COMMUNITY DAYS

December 05–06, 2024
STUTTGART

Catena-X Automotive Network | Tractus X | ECLIPSE FOUNDATION | ARENA2036

SUPPORTED BY:

Cofinity-X | SOVITY

Funded by the European Union
NextGenerationEU

Supported by:
Federal Ministry for Economic Affairs and Climate Action
on the basis of a decision by the German Bundestag



Successful Test Automation using Open Source Methods

December 05th & 6th 2024

Harald Zimmer (doubleSlash)
Test Management for Catena-X e.V.

Fabian Kock (doubleSlash)

Agenda

Together we will walk through the basics of Cucumber in action within 2 hours



Introduction

- Tests pyramid and test automation
- Using Cucumber with easy to learn Gherkin syntax
- Creating a test cycle from test case to implementation to execution and report



Demonstration

- Simple generic examples of an implementation
- You don’t need to know about the functionality of a particular Tractus-X product



Hands-On Coding Exercise

- Try it out yourself and create your own first automated Cucumber test



Discussion & Wrap Up

- Lessons we learned and helpful resources
- A meeting format teams could use to plan test case creation into their agile development processes
- Alternative / complementary test methods that the community could use for test automation



PREREQUISITES: Engagement & fun. Bring your own notebook and favorite IDE.

Or just watch the demonstration and try out the examples for yourself later.



Introduction / Foundation

Cucumber – what is that and why should I consider using it?



Open-source tool / framework for Behavior-Driven Development (BDD)

- Arguably the #1, used by thousands of companies
- Bridges the gap between business requirements and technical tests



Facilitates Collaboration

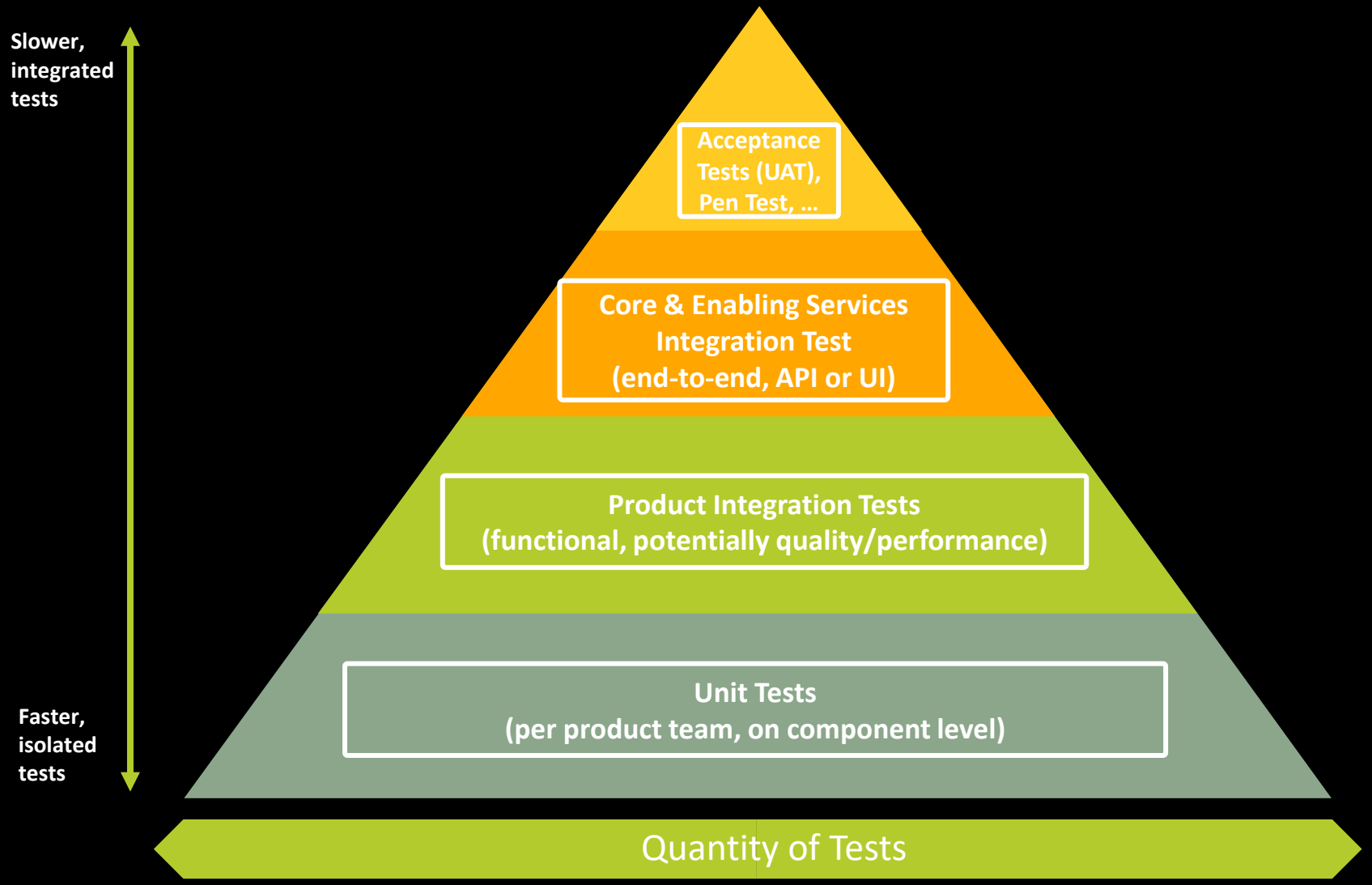
- Uses Gherkin, a plain-text, human readable syntax to define test cases
- Thus provides a shared language between business stakeholders, developers, and testers/QA



Executes automated integration / acceptance tests

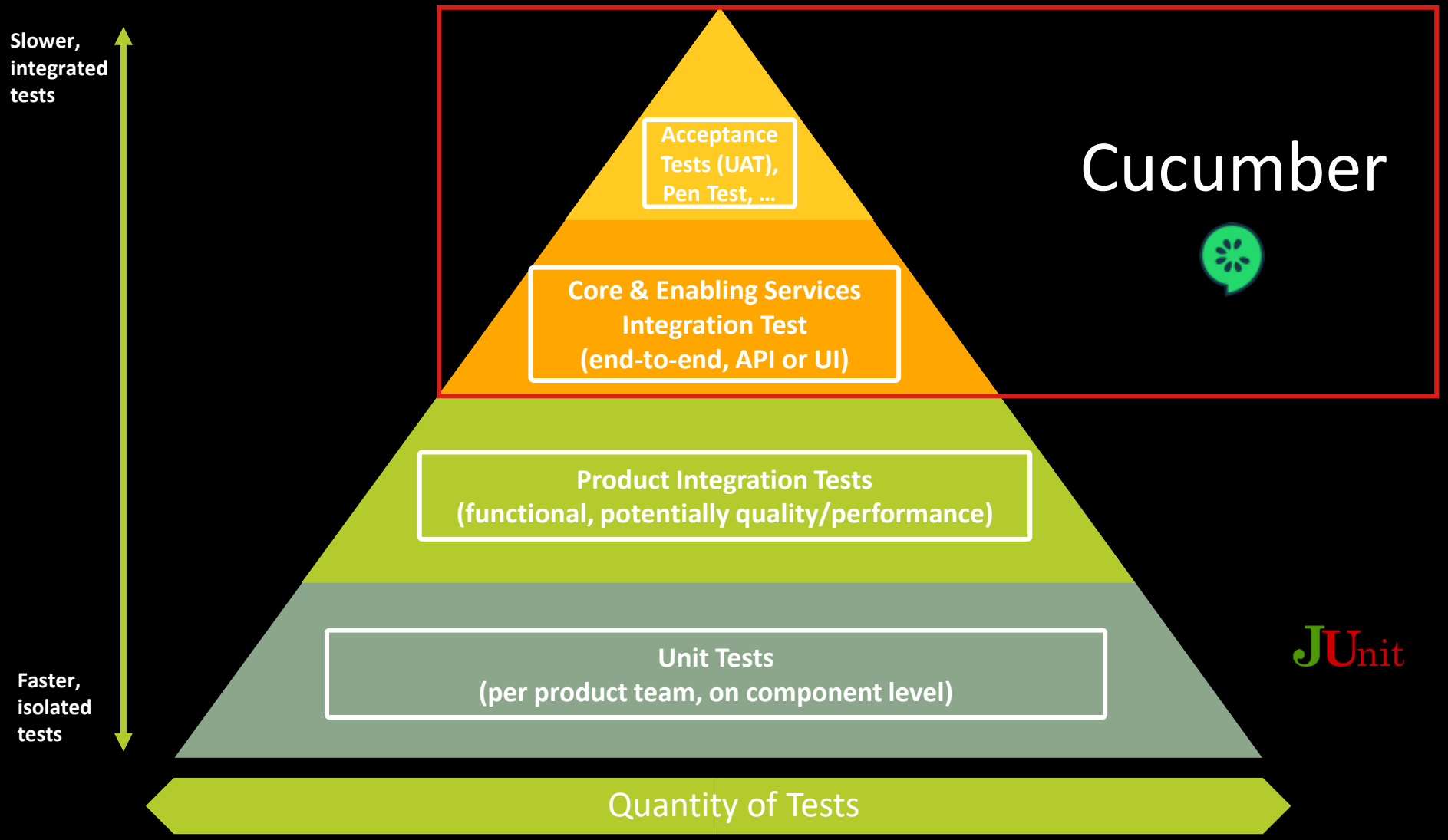
- Based on those scenarios written in a natural language
- Over time, a project can develop a “library” of reusable elements

Test stage pyramid



Test stage pyramid

Use Cucumber mainly for higher level Integration / Acceptance Tests



Why automate feature integration tests?



Build

Test
in Pipeline

Provide
Infrastructure

Deploy

Test
Integration



Test Case Descriptions in Gherkin

Write tests that people can understand with Cucumber

Feature: Is it Friday yet?
Everybody wants to know if it's Friday and they can soon go into the weekend ☺

Scenario: **Today is not Friday**
Given today is Thursday
When I ask whether it is Friday
Then the system should respond "No"

Scenario: **Today is Friday**
Given today is Friday
When I ask whether it is Friday
Then the system should respond "Yes"

Easy to learn Gherkin Syntax for Cucumber Test Cases

• Fast learning curve so test cases can be written and discussed by any role on the team (product owner or use case/business expert, developer, tester), potentially together in a joint meeting („3 Amigos“).

• Simple syntax to describe a test case via certain so-called “keywords”
  • Feature
  • Rule
  • Background (in newer versions of Cucumber)
  • Scenario (or “Example”)
  • ScenarioOutline (or “Scenario Template”)
  • Examples
  • Steps: given, when, then, and, but, *

• Localization: Those “keywords” have been translated in over 70 languages. Even Emoji ☺ 🥒 📕
Thus a test case in Gherkin doesn‘t necessarily have to be written e.g. in English or German

Implementing Cucumber Tests as Code



- Cucumber Step Definitions can be written independently from the programming language that the actual application you want to test is written in.

- There are several different “flavors” of Cucumber for different programming languages so you can pick your favorite based on skills or project preferences.
  - Java: Cucumber JVM – as shown in today’s demonstration
  - Javascript: Cucumber JS
  - Ruby: Cucumber Ruby
  - Go: Godog
  - …

The Gherkin parser and compiler is also implemented for many programming languages/frameworks:
Java, JavaScript, Ruby, Go, Python, .NET, C, C++, Perl, PHP, Dart

Key steps to implement a new Cucumber test
(Illustrative Example)

Sprint (best n-1 before feature implementation)



Description

As the backend system

I provide an API endpoint only accessible for authenticated users

So an admin test user can use the system's administrative functionalities.

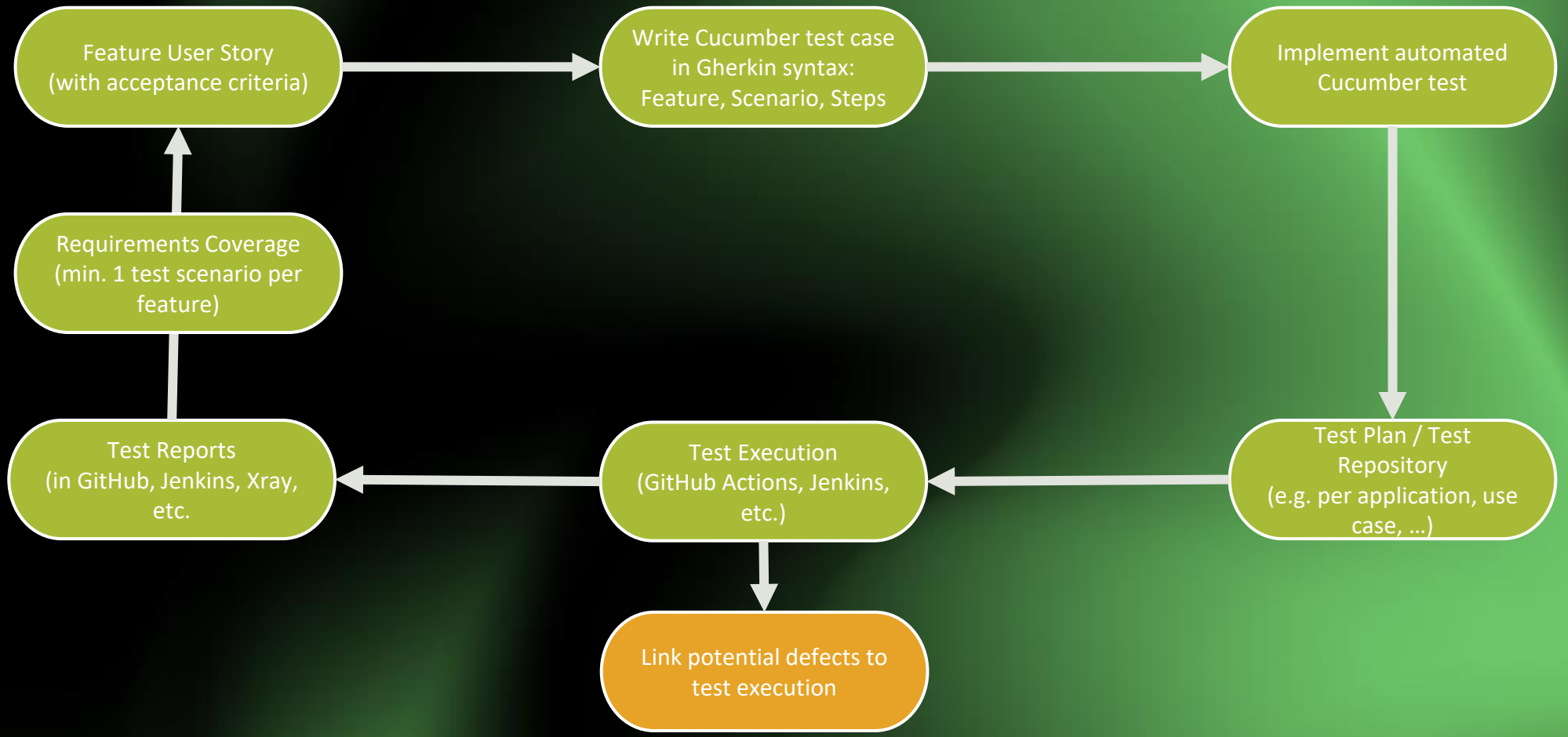
Scenario

Given I use the REST API
And I supply "AdminTestUser" credentials
When I call the "users" endpoint
Then I get response code 200
And I get "AdminTestUser" email address
And I get "AdminTestUser" department
And "AdminTestUser" role contains "Administrator"

Code Implementation of Steps

```
@Given ("^I use the REST API$")
public void iUseTheRESTAPI () {
// Code that calls the REST API
}
@And ("^I supply AdminTestUser credential")
public void iSupplyAdminCredentials () {
//Supply Admin Credentials
}
@When ("^I call users endpoint$")
...
```

Goal: Create a closed test cycle





Hands-On Code Demo



Testing a simple “Task Manager” demo app

Demonstration of a test implementation using Cucumber

- Writing a test scenario in Gherkin syntax (Given – When – Then)
- Implementing Cucumber step definitions in Java.
- Demonstrating API test creation as an example.
- Running tests and analyzing the results.
- Potentially: Integration with other tools (e.g. GitHub Actions) for automation.

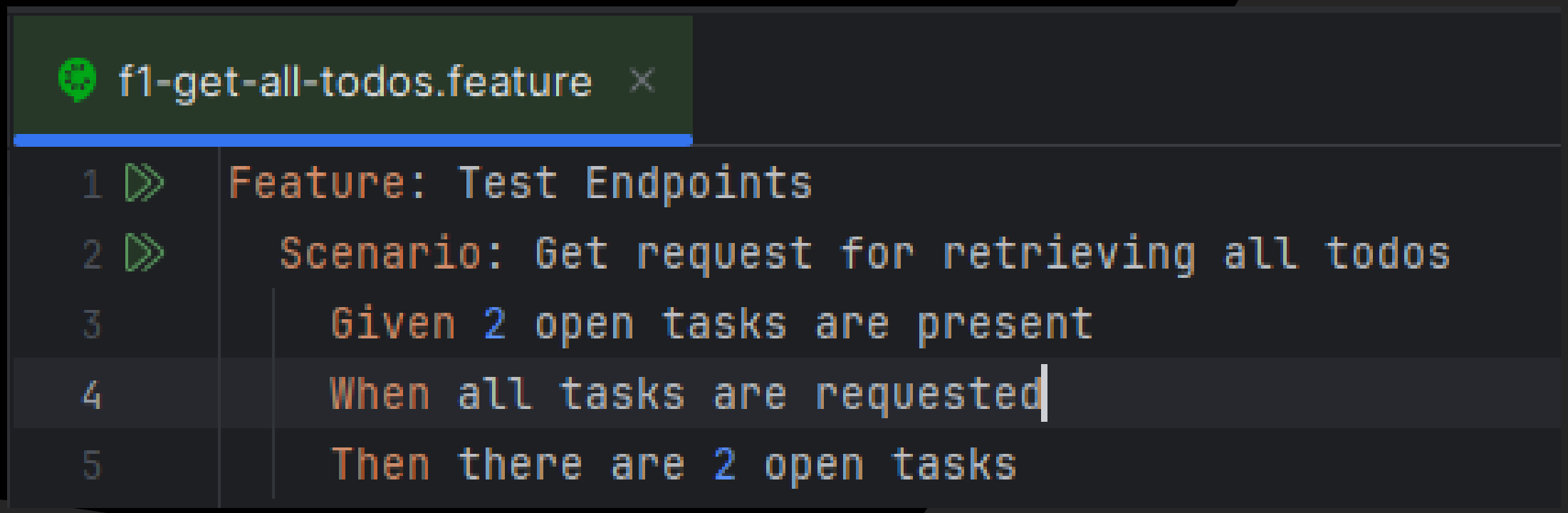
What is a Feature File?

- A **Feature File** is a core component in Cucumber, written in **Gherkin** syntax, that describes the functionality to be tested.

- A Feature file consists of **scenarios** (or **examples**) that represent specific test cases.

```
Feature:
  Scenario:
    Given [initial context]
    When [event occurs]
    Then [expected outcome]
```

Cucumber Feature Files (1/2)



Cucumber Feature Files (2/2)



```
Feature: Test Endpoints
  Scenario Outline: Get request for retrieving all todos
    Given <amount_of_tasks> open tasks are present
    When all tasks are requested
    Then there are <amount_of_tasks> open tasks
  Examples:
    | amount_of_tasks |
    | 10              |
    | 5               |
```



Developer

Coding Session 2 Mini-Challenges

Let´s try it out together

Hands-On



Repository

https://github.com/ds-fkock/AutomatedTestingWorkshop.git

Two mini-challenges to write your own first Cucumber test(s)

Building on the provided Cucumber tests for the previously demonstrated Task Manager application

1. Challenge:

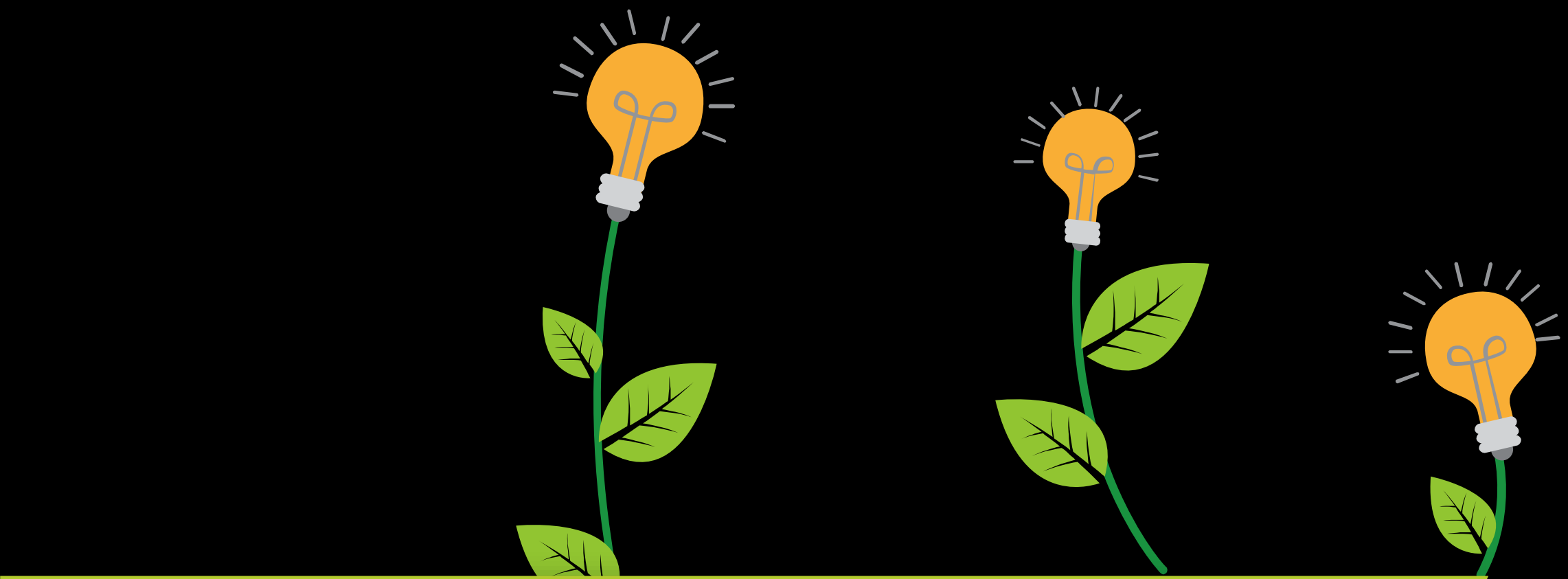
   Goal: Write a Cucumber test (feature file) that tests the **Patch endpoint** for a given task ID which is **non-existent**. So, the newly tested scenario would be that the task ID is not found (= negative scenario).

   Hint: Reuse as many steps from the provided “good case/happy case” scenario as possible.

2. Challenge:

   There is another previously untested **Filter endpoint** in the application for retrieving to-dos based on their title.

   Goal: Write a Cucumber test which validates the functionality of that endpoint.



Lessons Learned, Questions & Answers

Test Tool Stack Catena-X e.V. (in evaluation)

Test Management Tools

Recommended for:
- Test planning
- Test case definition
- Test reporting
- Defect handling (linked to GitHub issues)
- Planning in GitHub Boards + Timelines
- Evaluation: Alternative TM tools such as TestRail, Testmo, Testiny, etc.
  So far no better option for cost vs. features

Jira + XRAY

GitHub

Test Automation Tools / Frameworks

Recommended for:
- Cucumber for test automation
- SonarCloud for code coverage
- For Postman, Tavern, and similiar API test tools depending on product team + licensing
- Evaluation: Contract testing with Pact.io

Cucumber

sonarcloud

POSTMAN

PACT

- Potentially:
  GUI test automation with Cucumber + Selenium or Cypress
- Evaluation: Introduction of load test tools

Selenium

cypress

APACHE JMeter™

Gatling

CI/CD and Test Execution Tools

- GitHub repositories in Tractus-X respectively Catena-X e.V. organizations
- Test execution via GitHub Actions
- Argo CD for deployments on test environments
- HashiCorp Vault for Secrets

GitHub + argo

© 2024 Catena-X or a Catena-X affiliate company. All rights reserved.

Tractus-X Release Guidelines

1. For release 25.03 we are in progress of developing a Tractus-X Release Guideline (TRG) for Test Automation using Cucumber and potentially other open source methods

2. And another TRG for Code Coverage by tests using Sonarcloud



Tractus-X Real-Life Example: IRS

Policy Store API (update policy validUntil date)

Scenario

```
# Cleanup
Given no policies with prefix "aaaaaaaa-aaaa-aaaa-aaaa-" exist
# set up testdata
Given a policy with policyId "aaaaaaaa-aaaa-aaaa-aaaa-33333333333" is registered for BPN "BPNL1234567890AB" and validUntil "3333-11-11T11:11:11.111Z"
Then the create policy response should have HTTP status 201 and policyId "aaaaaaaa-aaaa-aaaa-aaaa-33333333333"
# act and assert
When I update policy "aaaaaaaa-aaaa-aaaa-aaaa-33333333333", BPN "BPNL1234567890AB", validUntil "3334-11-11T11:11:11.111Z"
Then the update policy response should have HTTP status 200
When I successfully fetch all policies
Then the BPN "BPNL1234567890AB" should have a policy with policyId "aaaaaaaa-aaaa-aaaa-aaaa-33333333333" and validUntil "3334-11-11T11:11:11.111Z"
```

Java implementation (small excerpt)

```
    @Then("the update policy response should have HTTP status {int}")
    public void theUpdatePolicyResponseShouldHaveStatus(final int httpStatus) {
        this.updatePoliciesResponse.statusCode(httpStatus);
    }
}
```

Tractus-X Real Examples

- Additionally, the colleagues have implemented integration workflows with Xray, GitHub runner, etc.

Overall Execution Status

19 PASSED 1 FAILED TOTAL TESTS: 20

Only My Test Runs | Filters | 100 | Columns

Rank	Key	Summary	Test Type	Dataset	#Defects	Status	A
1	CXTM-667	Policy Store API: Registering policy for invalid BPNL (ba...	Cucumber		0	PASSED	
2	CXTM-626	Policy Store API: Register policies	Cucumber		0	PASSED	
3	CXTM-625	Policy Store API: Update a policy validUntil date	Cucumber		0	PASSED	
4	CXTM-624	Policy Store API: Update a policy validUntil date for a p...	Cucumber		0	PASSED	
5	CXTM-623	Policy Store API: Add BPN to policy	Cucumber		0	PASSED	
6	CXTM-622	Policy Store API: Add policyId to given BPNs	Cucumber		0	PASSED	
7	CXTM-621	Policy Store API: Delete some policies	Cucumber		0	PASSED	
8	CXTM-620	Policy Store API: Register policy without validUntil (bad...	Cucumber		0	PASSED	
9	CXTM-619	Policy Store API: Fetching policies by BPNLs should fail ...	Cucumber		0	PASSED	



Tractus-X Real-Life Example: IRS Test Run Details

Scenario

```
1  # Cleanup
2  Given no policies with prefix "aaaaaaaa-aaaa-aaaa-aaaa-" exist
3  # set up testdata
4  Given a policy with policyId "aaaaaaaa-aaaa-aaaa-aaaa-33333333333" is registered for BPN "BPNL1234567890AB" and validUntil "3333-11-11T11:11:11.111Z"
5  Then the create policy response should have HTTP status 201 and policyId "aaaaaaaa-aaaa-aaaa-aaaa-33333333333"
6  # act and assert
7  When I update policy "aaaaaaaa-aaaa-aaaa-aaaa-33333333333", BPN "BPNL1234567890AB", validUntil "3334-11-11T11:11:11.111Z"
8  Then the update policy response should have HTTP status 200
9  When I successfully fetch all policies
10 Then the BPN "BPNL1234567890AB" should have a policy with policyId "aaaaaaaa-aaaa-aaaa-aaaa-33333333333" and validUntil "3334-11-11T11:11:11.111Z"
```

Results 1

Context	Duration	Status
-	4s 566ms	PASSED
Backgrounds		
Given the IRS URL "https://irs.int.catena-x.net/" -- policystore	1 millisec	PASSED
And the admin user api key -- policystore	0 millisec	PASSED
Steps		
Given no policies with prefix "aaaaaaaa-aaaa-aaaa-aaaa-" exist	3 secs	PASSED
Given a policy with policyId "aaaaaaaa-aaaa-aaaa-aaaa-33333333333" is registered for BPN "BPNL1234567890AB" and validUntil "3333-11-11T11:11:11.111Z"	516 millisec	PASSED
Then the create policy response should have HTTP status 201 and policyId "aaaaaaaa-aaaa-aaaa-aaaa-33333333333"	8 millisec	PASSED
When I update policy "aaaaaaaa-aaaa-aaaa-aaaa-33333333333", BPN "BPNL1234567890AB", validUntil "3334-11-11T11:11:11.111Z"	505 millisec	PASSED
Then the update policy response should have HTTP status 200	0 millisec	PASSED
When I successfully fetch all policies	489 millisec	PASSED
Then the BPN "BPNL1234567890AB" should have a policy with policyId "aaaaaaaa-aaaa-aaaa-aaaa-33333333333" and validUntil "3334-11-11T11:11:11.111Z"	9 millisec	PASSED

Other test automation frameworks/tools already in use by Tractus-X product teams

#	Product or Product Team	Used Test Framework	Links	Remark
01	Trace-X	Cucumber	GitHub	For Xray integration see GitHub Action
02	IRS	Cucumber Tavern	GitHub	Tavern used for API tests (Python Pytest) Plus Insomnia Collections to trigger tests.
03	PURIS	Postman	GitHub	Plan: Using Bruno as an open-source alternative to Postman
04	Knowledge Agents	Postman	Postman.com	
05	SLDT Team	Postman Tavern	Jira Xray (CXTM-60, -61)	Also as Insomnia Collection (hosted internally) Tavern with GitHub Action for API tests
06	BPDM	Postman	GitHub	
	…	…	…	

For consideration:

- Postman not accepted in some organizations due to unclear licensing situation.
  Where in use, Postman tests could be integrated for reporting in Xray using Newman

Goal: Increasing test automation across the integrated tool chain, while harmonizing recommended testing frameworks and tools

Reducing / Phasing Out

- Reliance on manual tests, especially for regression test
- Potentially the number of used testing tools / frameworks between products
- Reliance on external systems as part of the test object
- Test execution time and effort
- Reporting effort

Expanding

- Automated tests (of APIs)
- Test coverage and quantification
- Test-/Behavior-Driven Development, in particular Cucumber Tests with Gherkin syntax
- Consolidate and add mocks/stubs for product tests and future test beds? (Wiremock, other frameworks)

Adding

- End-to-end smoke tests for main use cases, in order to quickly evaluate test environment following a reset or release candidate deployments
- Automated GUI tests with Selenium or Cypress (where useful instead of manual tests)
- Potentially pen testing, load testing and other test types on Stable Environment
- To be evaluated: Pact.io for interface contract definitions between applications

Suggestion to embed test case creation into your agile development process: “Three Amigos” meetings



Details: Suggestion to embed test case creation into your agile development process: “Three Amigos” meetings

The **”Three Amigos”** is a meeting format of team members from Business (Design), Development, and Testing/QA, to mature new testing-related product backlog items before they are added to a sprint.
It facilitates shared understanding and shared vocabulary about features across a Scrum team.

- Occurs one to two sprints before a feature is expected to go into development
- Brings together three perspectives: Business, Development, and Testing/QA
- Does NOT focus on features respectively tests planned to be implemented in the current sprint
- Matures user stories so that they meet the team’s Definition of Ready
- Builds a shared understanding about the intent of a work item/increment



Typical “Three Amigos” meeting outline and goals

- 1 or max. 2 half-hour meetings during each sprint to discuss product backlog items for future sprints
- Business representative (product owner, use case expert, etc.) presents the requirements and acceptance criteria for the new feature
- Jointly discussing the new requirements and how to develop tests for them
- Stories and tasks as needed are added to product backlog
- Team members identify missing requirements and edge cases
- Team estimates the effort
- Requirements are updated until they are ready for development
- Team members identify who the assignee should be for each story and task



Lessons Learned (1): Shift-left/shift-down your tests in the test pyramid

- As discussed earlier for the test pyramid, we would recommend to use Cucumber initially for a fairly small number of tests at the higher test levels of end-to-end integration or “acceptance” level tests.
- Test development, maintenance effort and runtimes can arguably be too “expensive” to use it for lower-level product (or sub-system) and unit tests, for which other established methods and tools exist.
- If a higher-level test spots an error and there was no lower-level test failing, you should also add a lower-level test covering that issue to catch it earlier.
- As for other test framworks, the quality of the used test data is still very important for working Cucumber tests



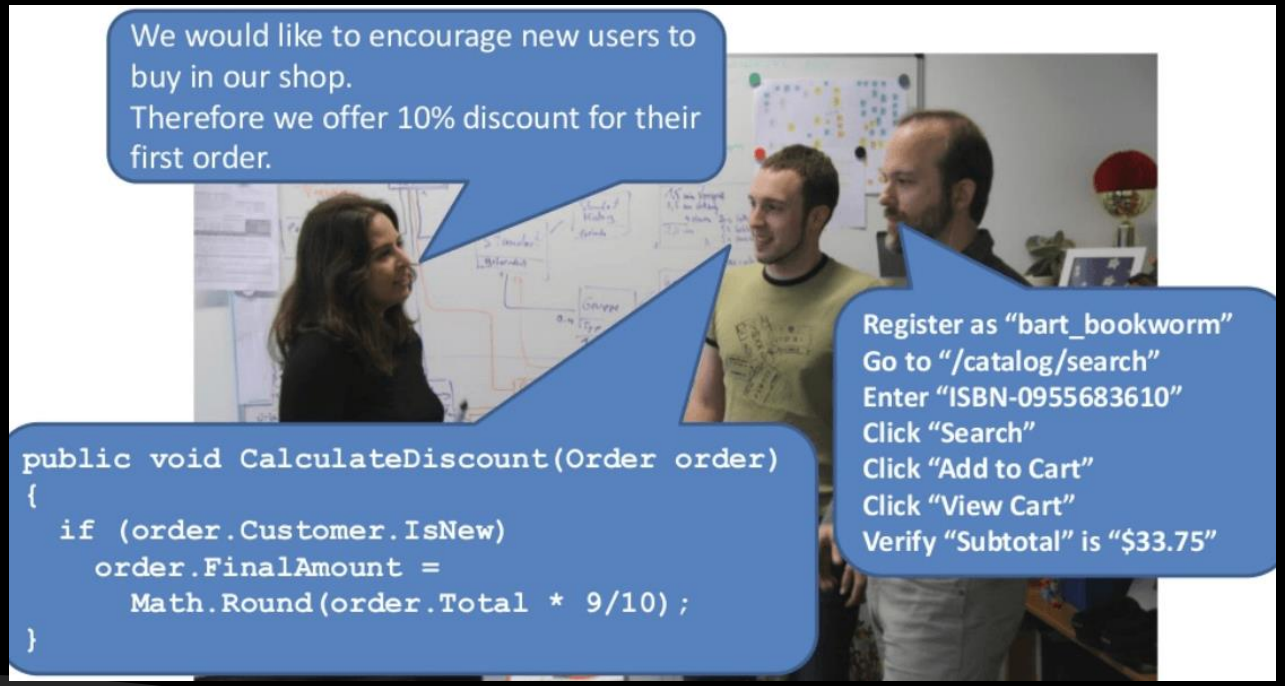
Lessons Learned (2):
Keep tests focused on use cases, not technical terms

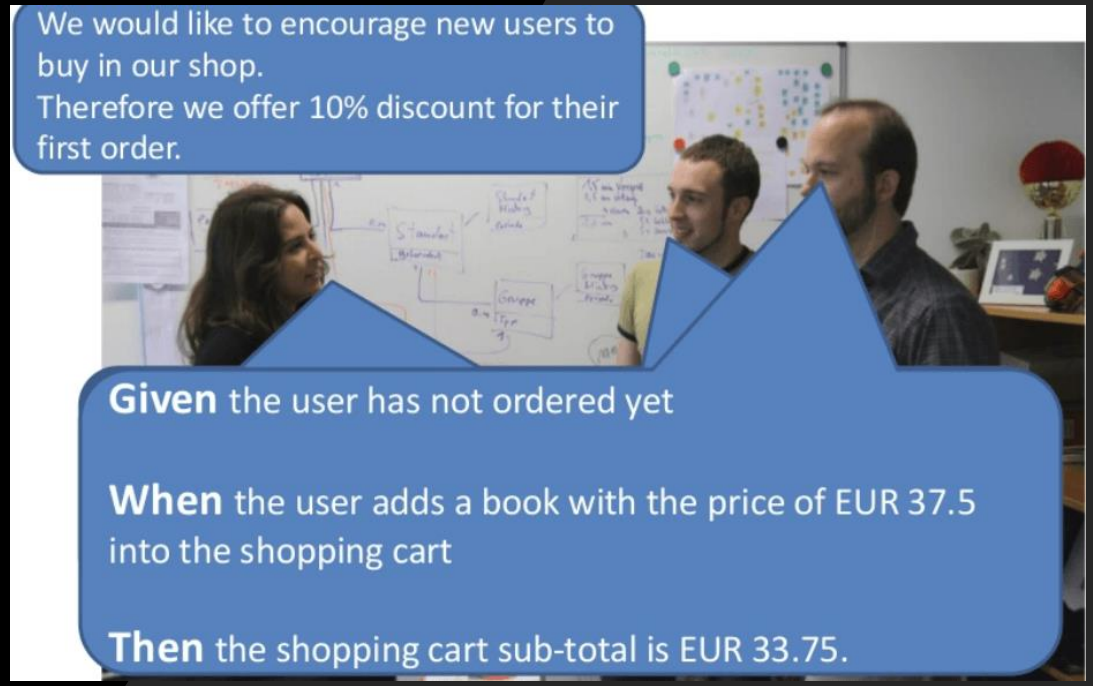
Regarding Cucumber and Gherkin syntax:

- Write scenarios in **plain language** that non-technical stakeholders (e.g. business users) can easily understand.
- Keep the **Gherkin feature file** rather free from technical details. Focus on use cases and behavior, not the technological implementation of system internals or API calls. That only becomes relevant in the code implementation of the test.
- Try to avoid feature-coupled step definitions, as those can‘t be resused across features or scenarios
- Try to avoid „conjunction steps“ that combine too many things in one test step (split up with And, But)

Lessons Learned (2):

Keep tests focused on use cases, not technical terms





Lessons Learned (3): Best Practices

- Each scenario should test **one specific behavior** or feature.
- Break down complex scenarios into smaller, more focused ones.

  Bad:

  Scenario: User logs in, updates profile, and logs out

  Good:

  Scenario: User logs in

  Scenario: User updates profile

  Scenario: User logs out

References

Demo Application used in this workshop:

- Sample code and challenges https://github.com/ds-fkock/AutomatedTestingWorkshop.git
- Wiki https://cxwiki.bonnconsulting.io/en/tractus-x-community-days/2024/test-automation-with-open-source-metho

Documentation on Cucumber and Gherkin

https://github.com/cucumber/gherkin
https://cucumber.io/docs/cucumber/

Tractus-X Examples

Trace-X https://github.com/eclipse-tractusx/traceability-foss/tree/main/tx-cucumber-tests
IRS https://github.com/eclipse-tractusx/item-relationship-service/tree/main/irs-cucumber-tests
IRS Xray integration code: https://github.com/eclipse-tractusx/item-relationship-service/blob/main/.github/workflows/cucumber-integration-test-xray.yaml



Thank you!

Contact me regarding our
Test Management Team for Catena-X e.V.
harald.zimmer@doubleslash.de

fabian.kock@doubleslash.de