



NTT DATA

BuDDy

BDD Automation Platform

**AGILE/
DEVOPS
GLOBAL
CONFERENCE**

Cesare Salerno

Solution Architect - - NTT
DATA

Cesare Salerno



**Solution Architect
NTT DATA**

Cesare Salerno works in NTT DATA Italy as Solution Architect and Project Team Leader since 2016.

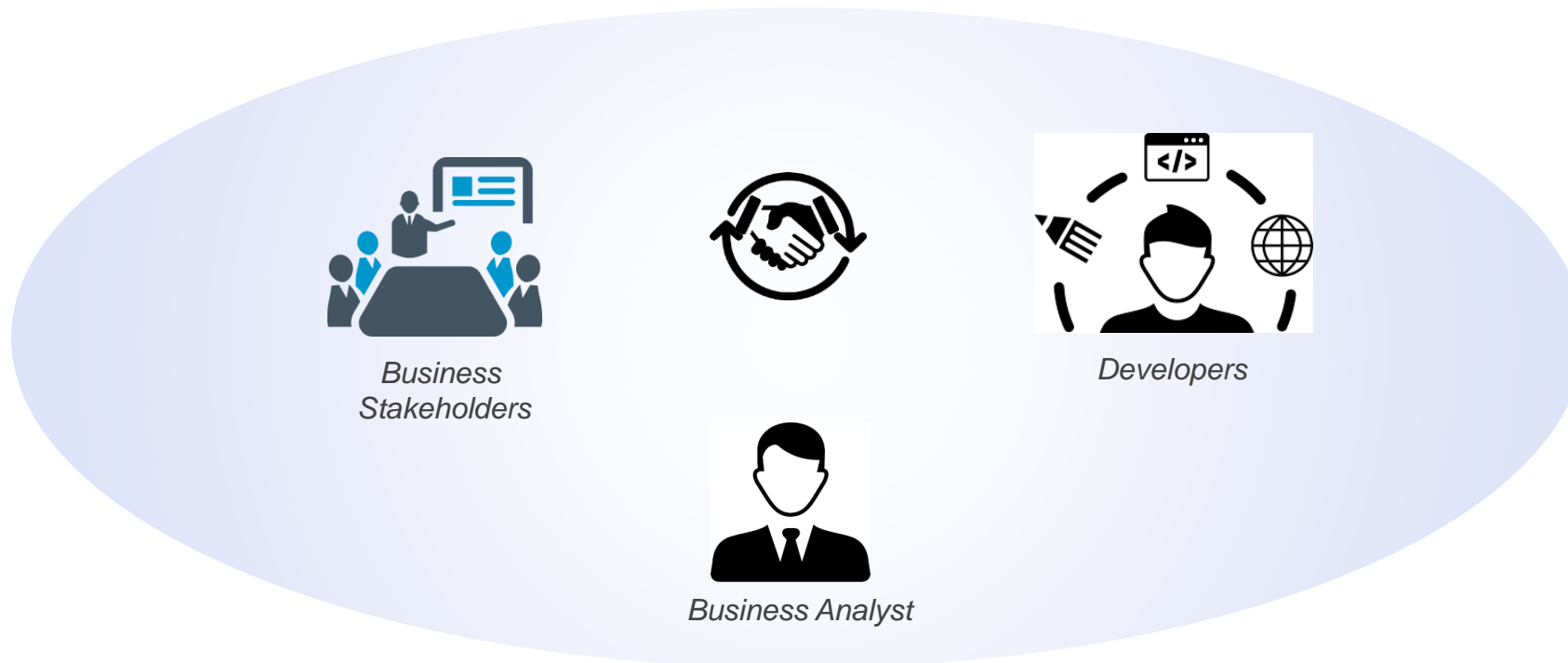
He is passionate about computer science, software quality expert and agile evangelist.

During last years He designed software architectures for big NTT DATA's customers and contributed to implement remote working models for international and multi-country teams, leveraging on scaled agile methodology and adopting DevOps best practices and test first approaches (Behavior-Driven Development, Test-Driven Development).

Overview

A common language is the most obvious binding element in any society

Michael Howard



To make the collaboration amidst Business Stakeholder, Business Analysts and Developers more effective, we propose an **automation platform** supporting the **adoption of the BDD practices** and paving the way to have a **common language** and avoid misunderstandings when defining and developing business requirements.

Main Goal

1 Writing in Natural Language

Often, requirements are misunderstood due to a lack of an expressive and accepted language between Business Analysts and Devs. Our solution is to allow Business Analysts to use a natural language to make requirements able to display the desired behaviour of a system, the so called practice of **behaviour-driven development**.



Using Domain-based Dictionaries 2



Every language in the World has its own dictionary. BuDDy implements specific dictionaries featuring business domains such as e-commerce and web portals. Dictionaries are **extensible** to make BuDDy usable on every business domain; further, they can be interpreted by an **inferential engine** capable of making quality checks completely automated.

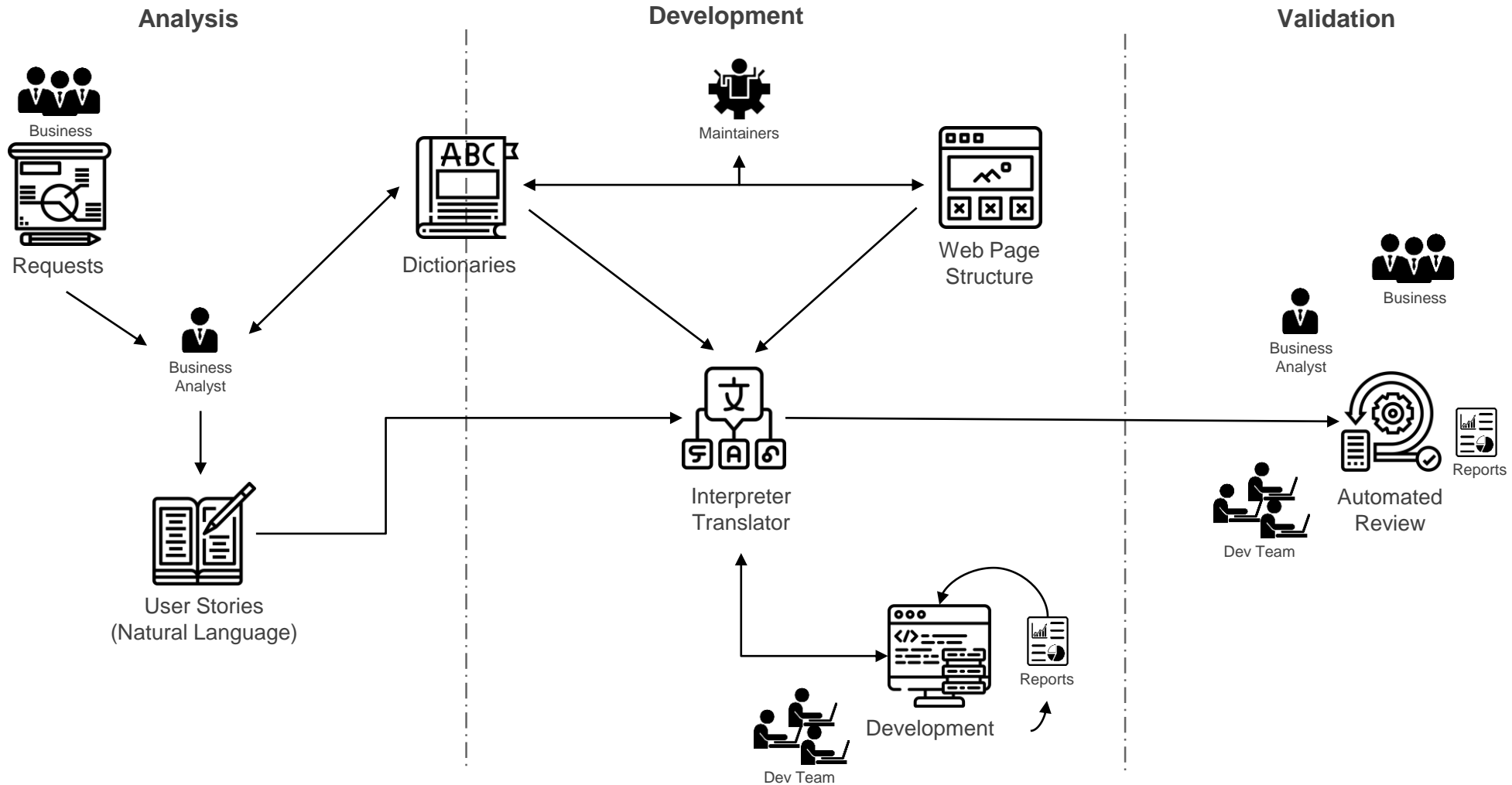
3 Automating reviews

Starting from the well-known Requirements Management System (eg Jira), **BuDDy** paves the way towards **automated tests** interpreted by the inferential engine. Tests are done on the basis of the business behaviour expressed into the requirements by means of the dictionaries and then **reports** are produced for **continuous improvement**.



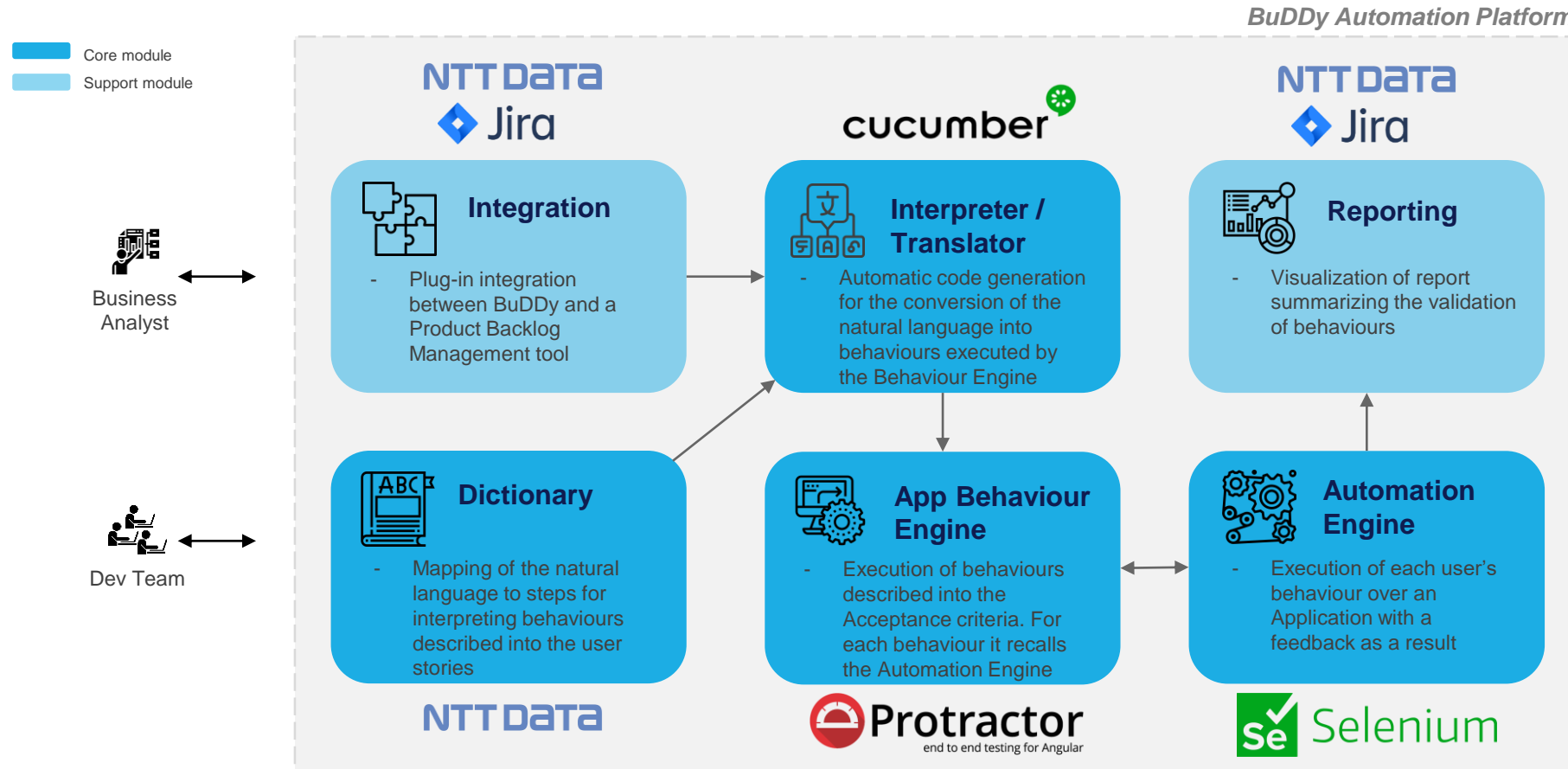
BuDDy

A complete journey from requirements to implementation and test



BuDDy Automation Platform

Architecture and Submodules



Main Benefits on Delivery Process



Saving time due to less effort spent in interpreting requirements



Saving costs due to less re-works



Optimization of Dev Cycle due to a standardization of Agile Software Development Life Cycle stages



Increase of Quality due to the application of the BDD practices for the compliance of Acceptance Criteria



Increase customer satisfaction due to a higher transparency and an optimization of agreements between NTT DATA and Customers.

BuDDy Core features

This is what we call
"BuDDy Hello World"

A sample user story
describing a typical login
flow

Acceptance Criteria are
written according to the
"classic" rule-based
paradigm

Projects /  TDD/Buddy / TDDBUD-75 /  TDDBUD-98

Login to BuDDy Dashboard

 Attach

 Create subtask

 Link issue



 Acceptance Tests



Description

As a developer, I want to access the BuDDy Dashboard, so that I can be operative on the projects where I have permissions.

Acceptance criteria

1. Every user should be registered to make a Login
2. To login, the user should enter:
 - a. username
 - b. password
3. After entering credentials, user should submit his/her login
4. When user does not exist on the users database, then a message is shown "Sorry, login failed, please check your credentials"
5. When a user does not enter his/her credentials (username or password) as registered into the users database, then a message is shown "Sorry, login failed, please check your credentials"

BuDDy Core features



Acceptance Tests

Feature: BuDDy Dashboard

Scenario Outline: Wrong Login to BuDDy Dashboard

add tag

```
Given I am on "BuDDyLoginPage" page
When I type "<username>" in "username"
And I type "<password>" in "password"
And I click "loginButton"
Then I should see the "wrongLogin"
```

Examples:

```
|username|password|
|mario.candela|passwordNotValid|
|usernameNotValid|rightPassword|
|usernameNotExisting|rightPassword|
```

Scenario Outline: Login to BuDDy Dashboard

add tag

```
Given I am on "LoginPage" page
When I type "<username>" in "username"
And I type "<password>" in "password"
And I click "loginButton"
Then I should not see the "wrongLogin"
```

Examples:

```
|username|password|
|mario.candela|rightPassword|
```

AUTOMATED

AUTOMATED

The “new way” of defining Acceptance Criteria introduced by BuDDy

Acceptance Criteria are written in “Gherkin” formal language

BuDDy Core features



BuDDy is able to translate the behaviour steps written by the Business Analyst

A user interface has been developed for supporting the definition of the structure of the applications pages

The screenshot displays the BuDDy web application interface. The top navigation bar includes the BuDDy logo, a 'Light' theme selector, and user information for Mario Candela. The left sidebar contains navigation options: Dashboard, Projects, Dictionaries, and Features(BETA). The 'Current Project' section shows 'CRD' selected, with options for Edit, Page Objects, and Dictionaries.

The main content area is divided into two sections:

- CRD - Page Objects:** A table listing page objects with columns for Name, Type, Stock, Items, and Actions. A search bar for 'Page Object name' and a 'CREATE NEW PAGE OBJECT' button are also present.
- Components:** A table listing components with columns for Actions, Name, Selector, Type, Stock, and Items. A search bar for 'Component name' and a '+ COMPONENT' button are also present.

Name	Type	Stock	Items	Actions
LoginPage	Page	-	4	↶ ↷
PersonalDashboard	Page	-	6	↶ ↷
WidgetCustomer	Element	Single	4	↶ ↷
CustomerDashboardTab	Element	Single	2	↶ ↷
PartnerNotAvailableModal	Element	Single	1	↶ ↷
ExternalInformationTab	Element	Single	3	↶ ↷
WorkItems	Element	Single	-	↶ ↷
WorkItemsTab	Element	Single	8	↶ ↷

Actions	Name	Selector	Type	Stock	Items
+ COMPONENT					
✎ 🗑	12thMarch	crd-customer-dashboard > crd-card-create-fa > crd-modal > div > div > div > div.crd-modal-container > div.ng-star-inserted > form > div > div:nth-child(1) > p-calendar > span > div > div > div.ui-datepicker-calendar-container.ng-tns-c46-29.ng-star-inserted > table > tbody > tr:nth-child(2) > td:nth-child(6) > a	Element	Single	0
✎ 🗑	ActionsComponent	.flyout.relative.flyout-bottom.flyout-left.flyout-visible	Element	Single	0

BuDDy Core features



Clicking on "Run Button" from Jira User Story, BuDDy modules execute the acceptance tests corresponding to the behaviors defined by business analyst

Execution and results reports are automatically generated and directly integrated in Jira

Cucumberjs Report **buddy-framework** Passed: 1 Failed: 0
Generated a few seconds ago

1 Feature Passed 100%
1 Scenario Passed 100%

▼ Feature: Shop QA WKI Cucumberjs Report **buddy-framework** Passed: 0 Failed: 1
Generated a few seconds ago

▼ Scenario: Accept cookies

- ✓ Given I am on "ShopQAWKI" project
- ✓ Then I should be on "HomePage" page
- ✓ And "AcceptCookiesButton" should be visible
- ✓ When I click "AcceptCookiesButton"
- ✓ Then "AcceptCookiesButton" should not be visible

1 Feature Failed 100%
1 Scenario Failed 100%

▼ Feature: Shop QA WKI 1

▼ Scenario: Accept cookies 5 1

✓ Given I am on "ShopQAWKI" project	3s 177ms
✓ Then I should be on "HomePage" page	80ms
✓ And "AcceptCookiesButton" should be visible	118ms
✓ When I click "AcceptCookiesButton"	231ms
✓ And I wait 1 seconds	1s 9ms
✗ Then "AcceptCookiesButton" should be visible Show Error +	10s 29ms

BuDDy Advanced features – API Testing

BuDDy supports the definition of api-testing using the gherkin language through the Karate Framework

BuDDy Dashboard / BD-4
API JSONPlaceholder users endpoint

Buttons: Edit, Add comment, Assign, More, Start Progress, Done, Admin

Details
Type: Task
Priority: Medium
Labels: api
Status: **TO DO** (View Workflow)
Resolution: Unresolved

BuDDy
Gherkin
Feature: JSONPlaceholder
Scenario: Get users
Given url 'https://jsonplaceholder.typicode.com'
And path 'users'
When method get
Then status 200

Buttons: Save, Clear

People
Assignee: Unassigned
Reporter: nttdata
Votes: 0
Watchers: 1 Stop watching this issue

Dates
Created: 09/Dec/21 9:18 AM
Updated: 11 minutes ago

BuDDy
Select a test type: API test, End-to-end test, End-to-end regression test suite, API test, API regression test suite

Scenarios
2022-01-27 12:05:14 PM
[1:2] Get users

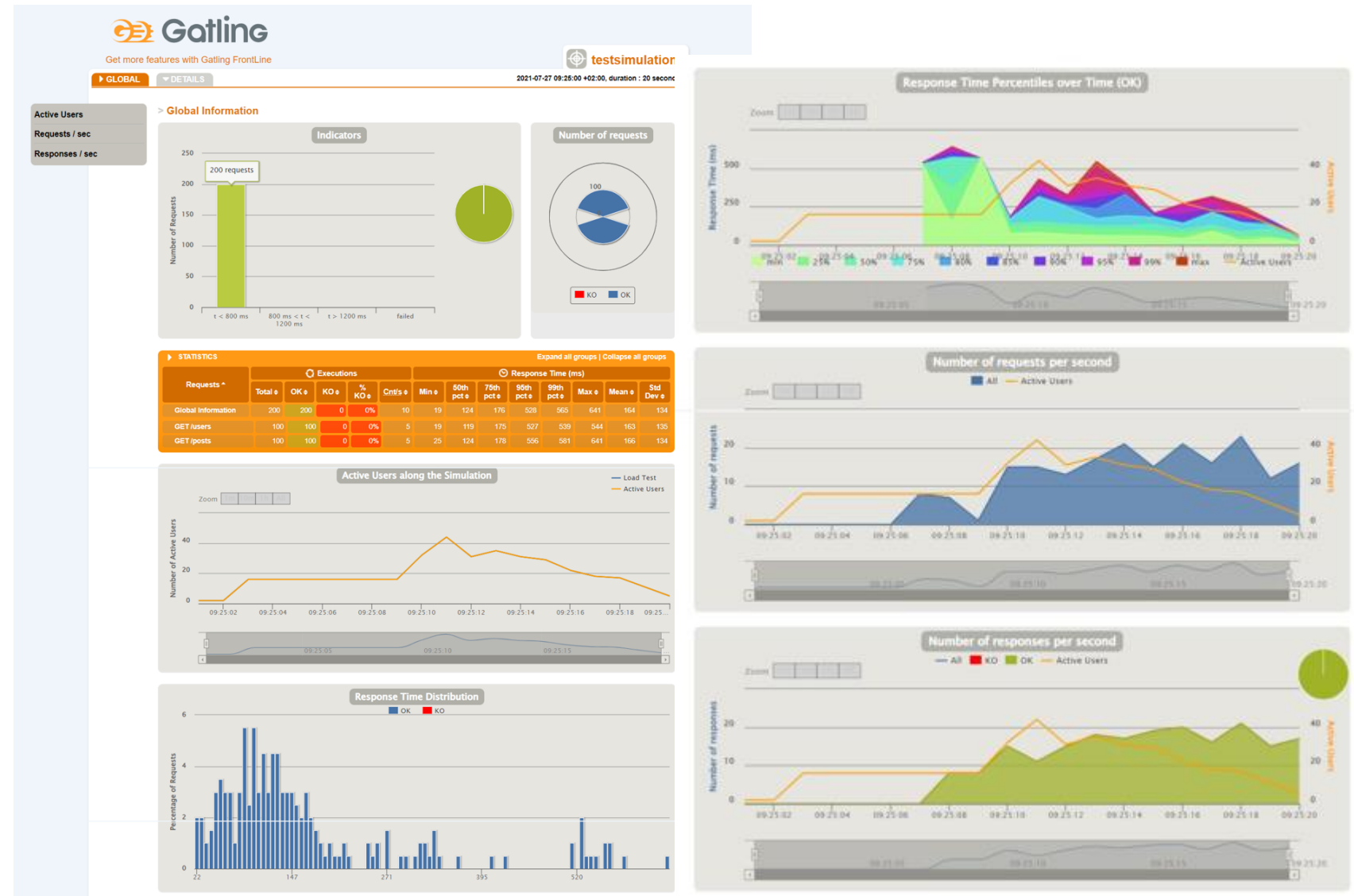
Summary | Tags | Feature: *buddy/features/BD-4.feature* | JSONPlaceholder

Scenario: [1:2] Get users		ms: 846
3	Given url 'https://jsonplaceholder.typicode.com'	1
4	And path 'users'	6
5	When method get	839
6	Then status 200	0

BuDDy Advanced features – Performance Testing

Each api-test can be converted into a performance test automatically.

Buddy provides a comprehensive report using the Gatling performance-testing framework

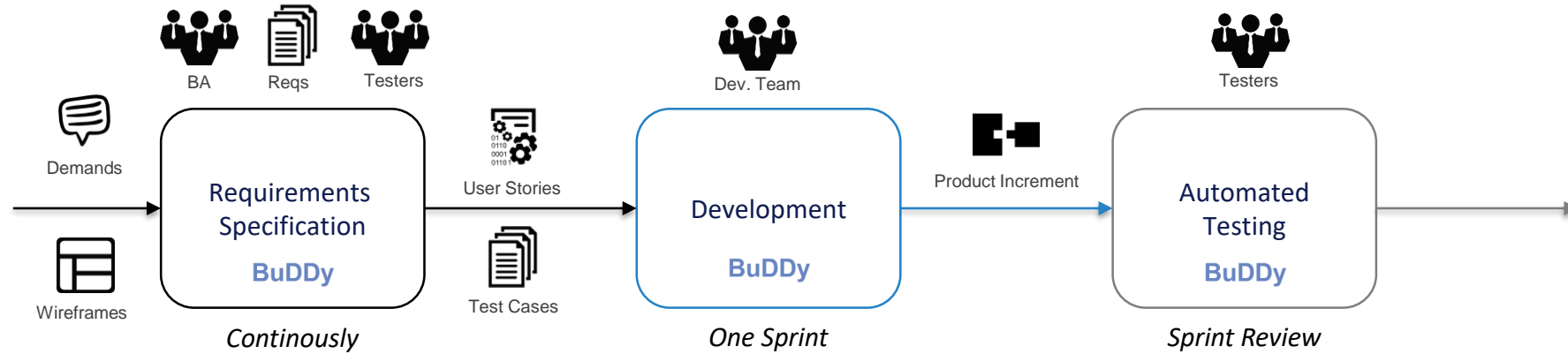


BuDDy in Action

Live Demo



BuDDy Adoption Status



There is an handover causing testers to write gherkin acceptance criteria starting from the "classic" rule-based ones.



Development is based on gherkin-based specs. A story is «done» only when all the test scenarios are executed correctly



It is not needed to spend a week to test the product increment → it could be done directly in the Sprint Review



Engagement of business analysts in writing gherkin AC to share a common language

*Let yourself be captivated by our **BuDDy**, Just try it.*

NTT DATA

THANK YOU

**AGILE/
DEVOPS
GLOBAL
CONFERENCE**

