

# Specification-By-Example with Gherkin

FROM STAKEHOLDER EXAMPLES TO LIVING DOCUMENTATION

---

Christian Hassa - ch@techtalk.at - @chr99ha  
Scrum Gathering Barcelona, 2<sup>nd</sup> of October 2012

# About me

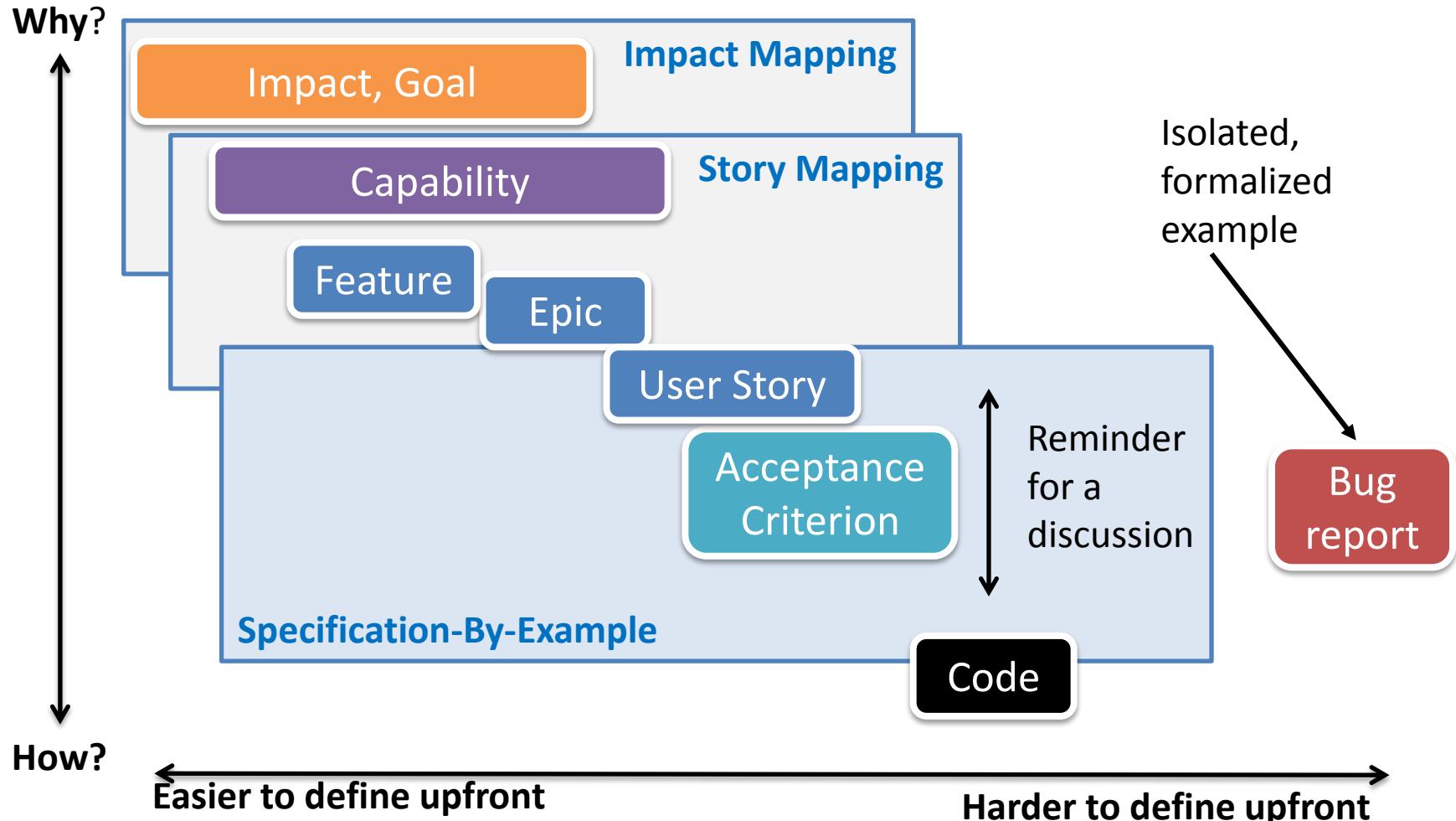
- Managing Partner at TechTalk:  
agile consulting and delivery
- Working in: Vienna, Budapest, Zurich

Vienna/Austria



# How do we specify requirements?

# Discovering what matters to customers



# Collecting Acceptance Criteria

As a potential customer  
**I want to collect books in a shopping cart**  
So that I can order several books at once.

*"Imagine this story is already implemented: how would you verify it?"*

*"I would try to put a book into the shopping cart ..."*

Books can be placed into shopping cart.

*"I would try to remove a book from the shopping cart..."*

Books can be removed from shopping cart.

*"I'd check whether the shopping cart is empty, when I enter the shop ..."*

Shopping cart should be empty when entering the shop.

*"I would try to add the same book again to the shopping cart ..."*

Adding same book again to shopping cart should increase quantity.

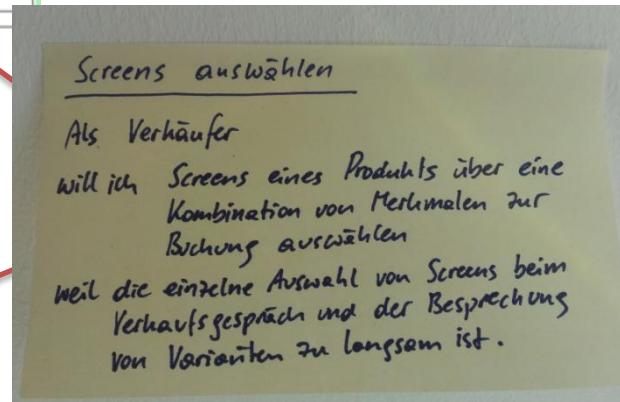


# Why do we need examples?

# Examples for user stories

**UI wire frames,  
existing UI**

**rules, key examples**



ID	A	B	C	D	E
1		Attribute 1	Attribute 2	Blacklist 1	Blacklist 2
2	1 Wien		Wohngebiet	Verkehr	Mobiltelef
3	2 NÖ			Wohngebiet	
4	3 OÖ				
5	4 Stmk				
6	5 Bgld			Büros	
7	6 Kärnten			Bahnhof	
8	7 Sbg			Uni/Schule	
9	8 Tirol			Bahnhof	
10	9 Vbg			Uni/Schule	Elektroartikel
11	10 Vbg				
12	11 Vbg				
13	12 Vbg			Büros	
14	13 Vbg			Büros	

**existing artifacts,  
samples**

# Exercise

As a potential customer  
*I want to collect books in a shopping cart*  
So that I can order several books at once.



- Work in groups of 2-3
- Discuss examples of how you would try out the shopping cart
- Draw/write down necessary examples you need during the discussion  
(UI mockup(s), concrete books, ...)
- Write down the list of discussed acceptance criteria
- *DON'T bother with formalizing examples yet*  
*e.g. Given/When/Then*

# Specification-by-Example

Examples ...

- make abstract descriptions  
*better understandable*

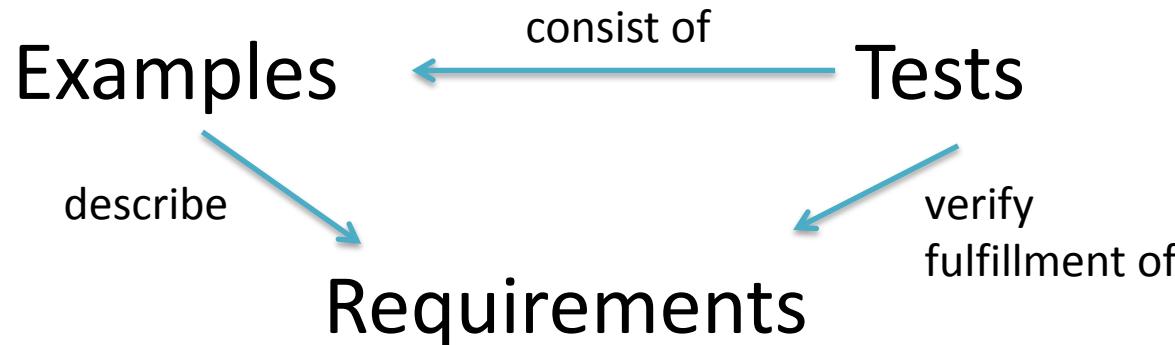
However ...

- examples are usually not formally  
*exchanged or documented*

**an example  
would be handy  
right about now**

*www.exampler.com*

Brian Marick



# Discussion of acceptance criteria

We would like to encourage new users to buy in our shop.  
Therefore we offer 10% discount for their first order.



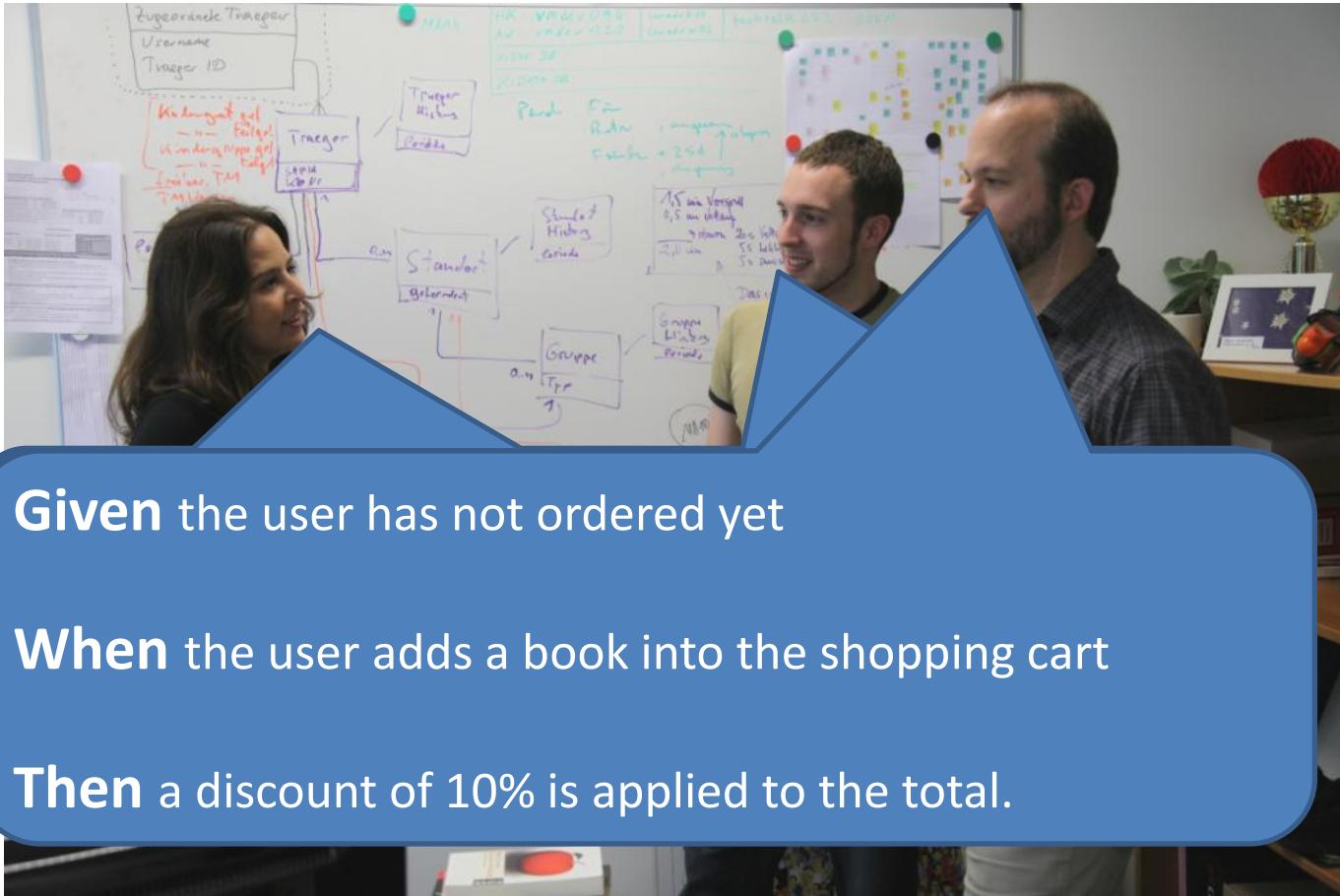
```
public void TestInitialOrderDiscount()
{
    Customer newCustomer = new Customer();
    Order newOrder = new Order(newCustomer);
    newOrder.AddBook(
        Catalog.Find("ISBN-0955683610")
    );
    Assert.Equals(33.75,
        newOrder.Subtotal);
}
```

Register as “bart\_bookworm”  
Go to “/catalog/search”  
Enter “ISBN-0955683610”  
Click “Search”  
Click “Add to Cart”  
Click “View Cart”  
Verify “Subtotal” is “\$33.75”



Original idea for the illustration: George Dinwiddie  
<http://blog.gdinwidiee.com>

# ... illustrated with formalized examples



Original idea for the illustration: George Dinwiddie  
<http://blog.gdinwidiee.com>

# Discover hidden assumptions

Actually, this not quite right:  
Books on sale should be excluded .



Original idea for the illustration: George Dinwiddie  
<http://blog.gdinwidiee.com>

# Collaboration: 3 amigos



Original idea for the illustration: George Dinwiddie  
<http://blog.gdinwidiee.com>

# Abstract acceptance criteria

As a shop visitor

*I want to collect books in my shopping basket  
so that I can purchase multiple books at once.*

Books can be added to the shopping basket

Books can be removed from the shopping basket

Shopping basket is initially empty

The same book can be added multiple times to the shopping basket

# Examples for acceptance criteria

As a shop visitor

***I want to collect books in my shopping basket  
so that I can purchase multiple books at once.***

Books can be added to the shopping basket

***Given my shopping basket is empty***

***When I add the book "Harry Potter" to my shopping basket***

***Then my shopping basket should contain 1 copy of "Harry Potter"***

# Examples for acceptance criteria

As a shop visitor

**I want to collect books in my shopping basket  
so that I can purchase multiple books at once.**

B

The same book can be added multiple times to the shopping basket

*Given my shopping basket contains 1 copy of "Harry Potter"*

*When I add the book "Harry Potter" to my shopping basket*

*Then my shopping basket should contain 2 copies of "Harry Potter"*

# Structure of examples

**Title:** Describes intention/abstract acceptance criterion

**Arrange:** Context, initial state of the system

**Act:** Execution of the feature

**Assert:** Assertion of observable behaviour

Triple-A  
constraint  
“Checks”

The same book can be added multiple times to the shopping basket

*Given my shopping basket contains 1 copy of "Harry Potter"*

*When I add the book "Harry Potter" to my shopping basket*

*Then my shopping basket should contain 2 copies of "Harry Potter"*

*And I should see the warning: "Book already existed in basket"*

Chaining  
up steps

Evolution of a domain specific  
test and specification language.



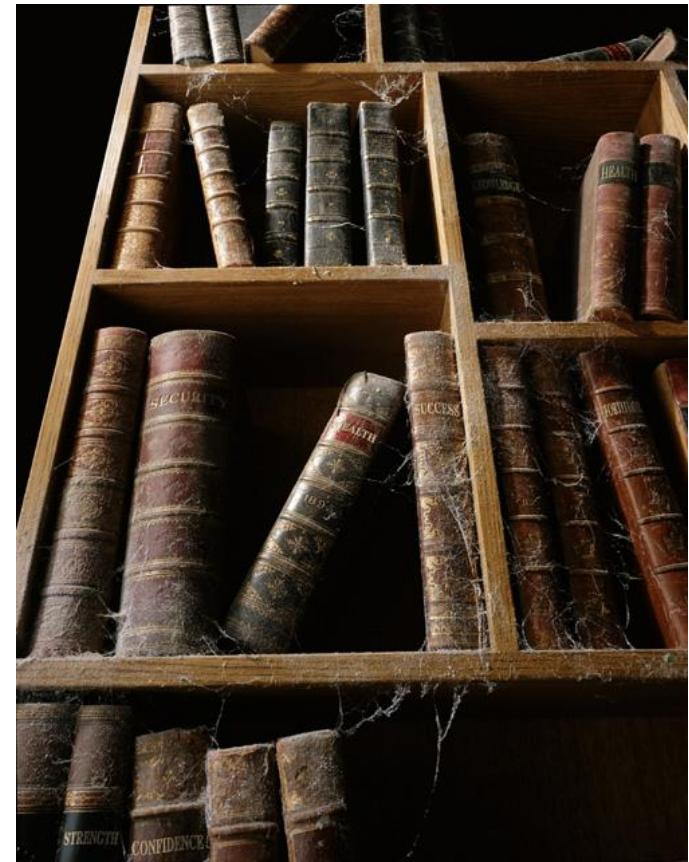
# Exercise

- Work in same previous groups of 2-3
- Pick an acceptance criteria from the previously compiled list
- Illustrate with examples formalized into
  - Given: Context, initial state
  - When: utilization of the feature
  - Then: assertions of observable behaviour
- Validate that scenario title (acceptance criterion) still describes intention of formalized scenario

How long do we  
care  
about the  
examples?

# Purpose of the examples

- Shared understanding:  
acceptance criteria
- Documentation:  
Look-up detail aspects  
of the system
- Regression-tests:  
Understand what  
assumptions have been violated

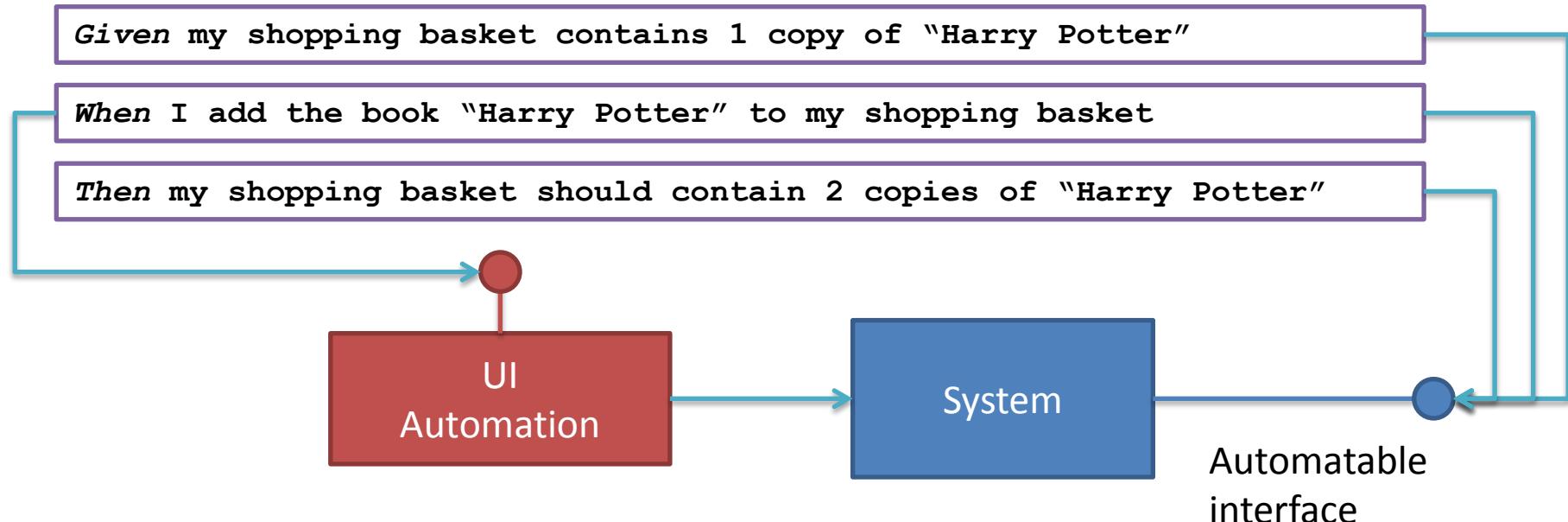


# Continuous validation with automation

„Step Definitions“ are binding individual steps to an automatable interface of the application.

Automation does not necessarily have to bind to the UI.

Automatability of system is supported/evolving with development.



„Living documentation“



# Demo



<http://www.specflow.org>

## Gherkin automation for .NET

- Visual Studio plugin (VS-Gallery)
- NuGet Package

What does that  
mean for  
testing?

# UI automation becomes expensive when ...

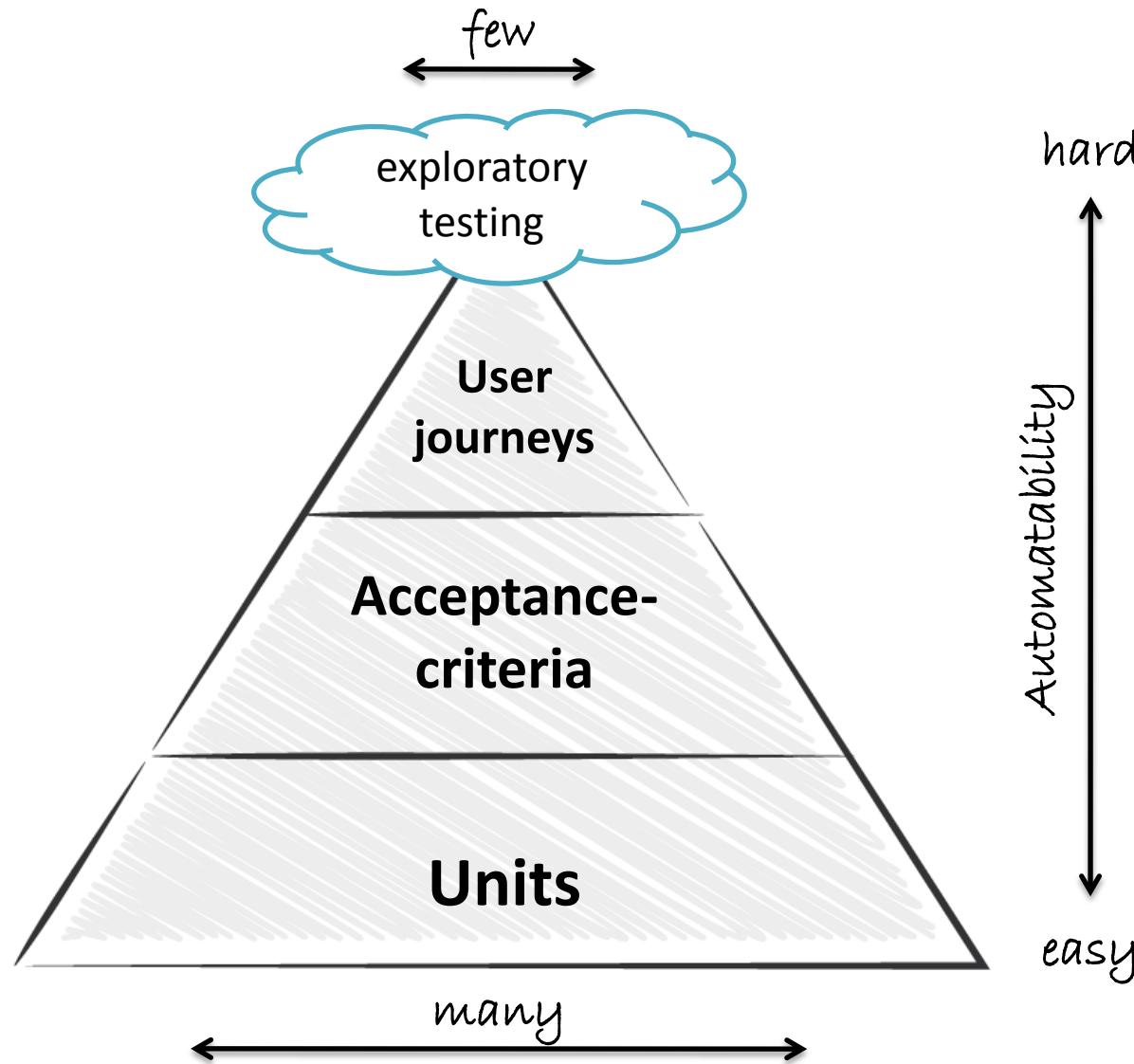


- trying to automate manual tests **structure**
- making tests unreadable when automating them **readability**
- automating after completing development **point in time**

# Structure

	Manual tests	Automated Check
Asserts	Multiple combined features	Single aspect of a single feature
Structure	ACT-ASSERT ACT-ASSERT ACT-ASSERT- ... 	ARRANGE – ACT – ASSERT
	<b>Dependent</b> on other features <b>Long test path</b> with high chance to break	<b>Independent</b> of other features <b>Short test path</b> with lower chance to break

# Test automation pyramid



# Readability

```
// Go to web page 'http://localhost:40001/' using new browser instance
BrowserWindow localhostBrowser = BrowserWindow.Launch(
    new System.Uri(this.RecordedMethod1Params.Url));

// Click 'Register found item' link
Mouse.Click(uIFundstückerkassenHyperlink, new Point(56, 9));

// Click 'Save' button
Mouse.Click(uISpeichernButton, new Point(44, 14));

int fundNr1 = int.Parse(uIFundNr127Pane.InnerText.Substring(9))

// Click 'Register found item' link
Mouse.Click(uIFundstückerkassenHyperlink, new Point(63, 7));

// Click 'Save' button
Mouse.Click(uISpeichernButton, new Point(34, 11));

int fundNr2 = int.Parse(uIFundNr128Pane.InnerText.Substring(9));

Assert.IsTrue(fundNr1 + 1 == fundNr2);

// Click 'Close' button
Mouse.Click(uICloseButton, new Point(26, 11));
```



# A readable test case

**Scenario:** New found items should receive a consecutive number for the current year

**Given** the previous found item of the current year had the number 145

} *arrange*

**When** I register a new found item

} *act*

**Then** the last found item of the current year should have the number

} *assert*  
146

# When to test (point in time)

defining the product

business view

Acceptance criteria  
(ATDD, BDD)

Exploratory tests  
Workflow tests

Unit Tests  
(TDD)

Performance, Scalability,  
Usability, Security, ...

technical view

Agile Testing Quadrants: Brian Marick

criticizing the product

New dimension: defining the product

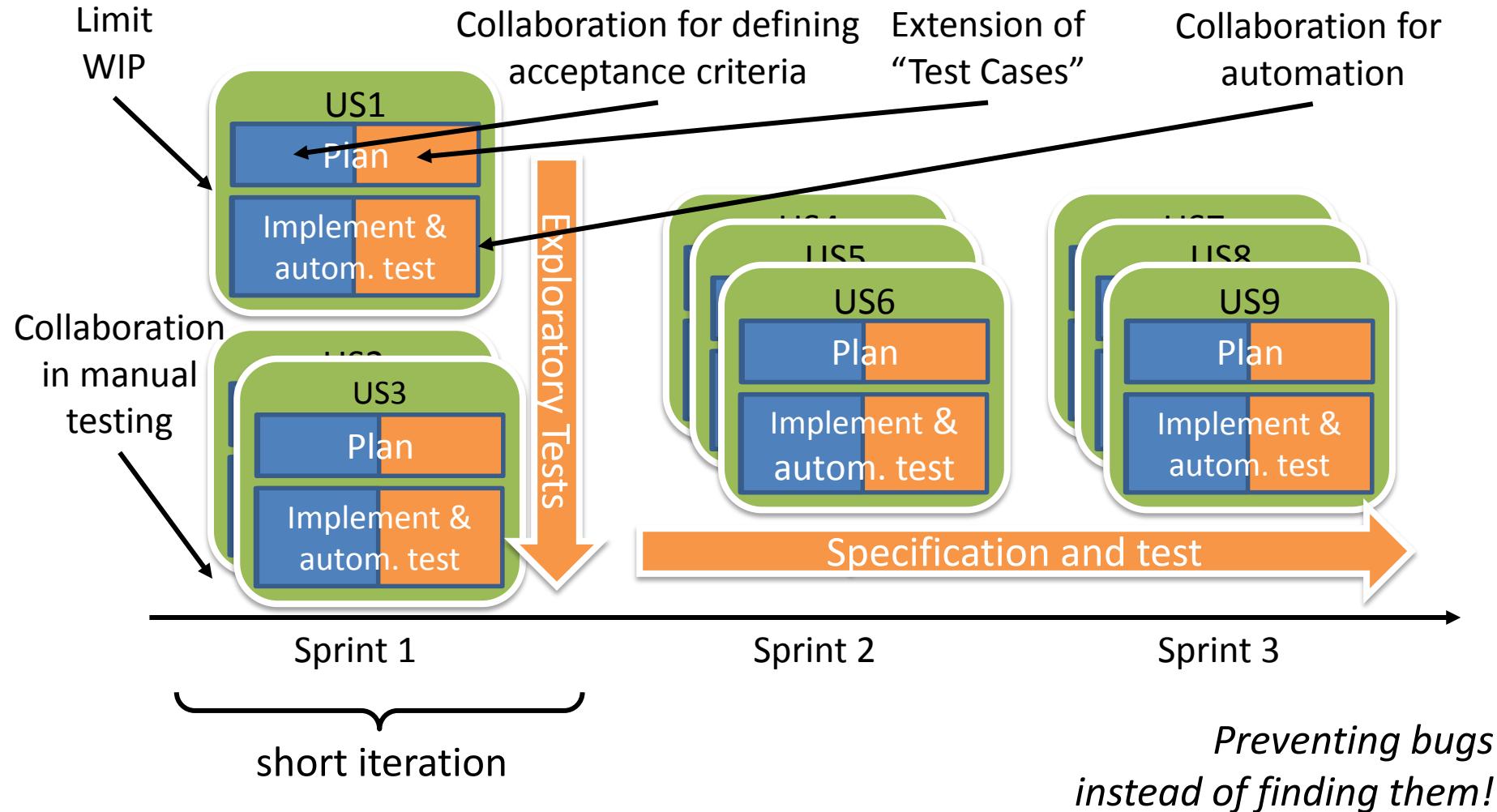
Synergy: Specification of requirements **and** tests

# Cross-functional teams



Co-creation  
Fast feedback

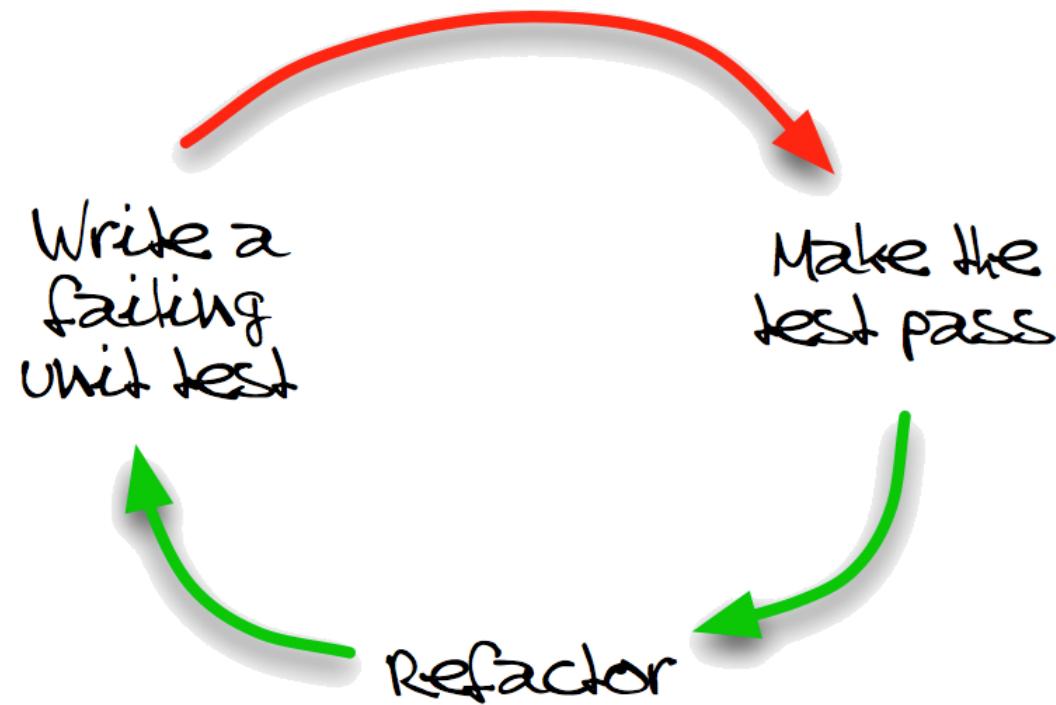
# Cross-functional work



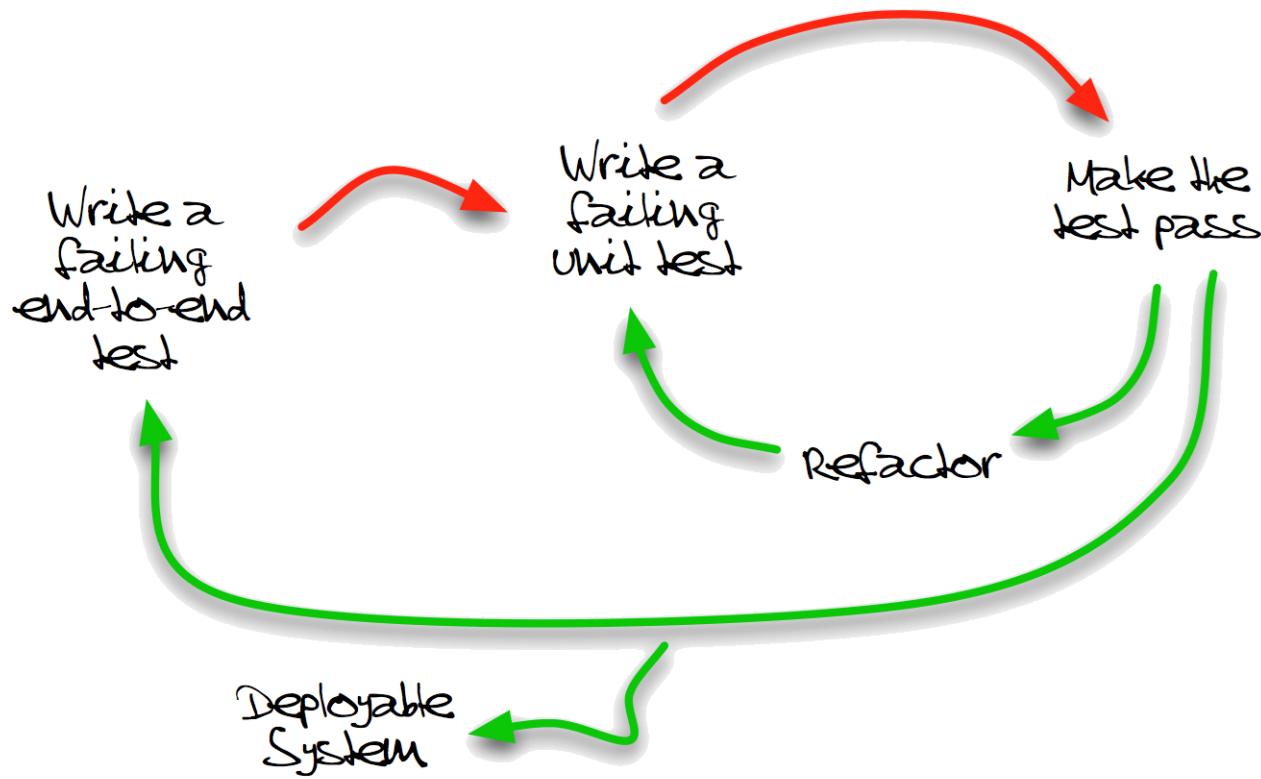


What does that  
mean for  
development?

# Workflow - TDD



# Extending TDD for business



# Transparency for stakeholders

## AD Project Dashboard

### Last Finished Sprint results

- Done report: [here](#)
- Application: [http://\[REDACTED\].techtalk.at:8890/](http://[REDACTED].techtalk.at:8890/)
- Test files:
  - Product & Booking import sample: [\[REDACTED\] Testdaten\\_01.xlsx](#)

### In Progress

- Build status: see on the right side
- Current done report: [on build server here](#)
- Current sprint report: [on build server here](#)
- Sprint burndown: [generated every day at noon](#) or [on reporting server here \(internal\)](#)
- Latest application (internal): [http://ttvm-\[REDACTED\]:8889/](http://ttvm-[REDACTED]:8889/)
- Latest smoke-tested application (internal): [http://ttvm-g-\[REDACTED\]:8888/](http://ttvm-g-[REDACTED]:8888/)

### Infrastructure summary

- Project Sharepoint Site: [https://projects.techtalk.at/projects/\[REDACTED\]](https://projects.techtalk.at/projects/[REDACTED])
- Build Server: [http://\[REDACTED\].techtalk.at:8111/](http://[REDACTED].techtalk.at:8111/)
- SpecLog: ttvm-tt-sfee-net4:4243 [\[REDACTED\]](#)

The dashboard features a logo for TechTalk with the tagline "IT WORKS - WE CARE". It displays two build status cards. The first card for "Acceptance" shows a green checkmark, build #176, and artifacts from 17 Jun 11 16:34. The second card for "CI" shows a green checkmark, build #121, and artifacts from 17 Jun 11 14:28.

# Current sprint report: all sprint scenarios

The screenshot shows a web browser window with the URL [.techtalk.at:8111/viewLog.html?buildId=149&buildTypeId=bt5&tab=SpecFlow\\_Reports](http://.techtalk.at:8111/viewLog.html?buildId=149&buildTypeId=bt5&tab=SpecFlow_Reports). The page title is "AcceptanceTests Test Execution Report". The top navigation bar includes links for "Projects", "Agents (1)", "Build Queue (0)", "Overview", "Changes (1)", "Tests", "Build Log", "Build Parameters", "Artifacts", "SpecFlow Reports", and a search bar. The main content area displays the test execution summary and detailed results for various features.

### AcceptanceTests Test Execution Report

Generated by SpecFlow at 05/05/2011 15:19 (see <http://www.specflow.org/>).

#### Summary

Features	Success rate	Scenarios	Success	Failed	Pending	Ignored
5 features	0%	50	0	0	50	0

#### Feature Summary

Feature	Success rate	Scenarios	Success	Failed	Pending	Ignored
<a href="#">Calendar views</a>	0%	8	0	0	8	0
<a href="#">Create booking</a>	0%	30	0	0	30	0
<a href="#">Create reservation</a>	0%	5	0	0	5	0
<a href="#">Show availability of spot length</a>	0%	3	0	0	3	0
<a href="#">View bookings related to utilization value</a>	0%	4	0	0	4	0

#### Feature Execution Details

Feature: [Calendar views](#)

Scenario	Status	Time(s)
Should allow paging backwards in month	pending	15.413
Should allow paging forwards in month	pending	0.187
Should allow switching to month view	pending	0.147
Should allow switching to the booking range	pending	0.437
Should display maximum 31 days when	pending	0.150

Scenario: Should display utilization in green when spot length invalid  
@specified

Given I am on the screen selection page  
And the calendar shows the booking range from '2011-05-11' to '2011-05-11'  
When I enter 'x' for the spot length  
Then the displayed overall utilization for all screens should be green for every day

# Starting with first scenario (AC)

The screenshot displays a web-based application interface for a Test Execution Report. At the top, a browser window shows the URL [.techtalk.at:8111/viewLog.html?buildId=151&tab=SpecFlow\\_Reports&buildTypeId=bt5](http://.techtalk.at:8111/viewLog.html?buildId=151&tab=SpecFlow_Reports&buildTypeId=bt5). The main content area is titled "AcceptanceTests Test Execution Report" and indicates it was generated by SpecFlow at 05/06/2011 10:44. The interface includes tabs for Overview, Changes (1), Tests, Build Log, Build Parameters, Artifacts, and SpecFlow Reports. A navigation bar shows "CI > #7 (06 May 11 10:43)". Below the tabs, there are links for "#6 | All history | #8".

**Summary**

Features	Success rate	Scenarios	Success	Failed	Pending	Ignored
5 features	0%	51	0	1	50	0

**Feature Summary**

Feature	Success rate	Scenarios	Success	Failed	Pending	Ignored
<a href="#">Calendar views</a>	0%	8	0	0	8	0
<a href="#">Create booking</a>	0%	31	0	1	30	0
<a href="#">Create reservation</a>	0%	5	0	0	5	0
<a href="#">Show availability of spot length</a>	0%	3	0	0	3	0
<a href="#">View bookings related to utilization value</a>	0%	4	0	0	4	0

**Feature Execution Details**

Feature: [Calendar views](#)

Scenario	Status	Time(s)
Should allow paging backwards in month	pending	15.554
Should allow paging forwards in month	pending	0.288
Should allow switching to month view	pending	0.189
Should allow switching to the booking range	pending	0.157
Should display maximum 31 days when	pending	0.161

Scenario: Should display utilization in green when spot length invalid  
@automated @in\_progress

Given I am on the screen selection page  
And the calendar shows the booking range from '2011-05-11' to '2011-05-11'  
When I enter 'x' for the spot length  
Then the displayed overall utilization for all screens should be green for every day

# Finishing the first scenario (AC)

The screenshot shows a web browser window with the URL [.techtalk.at:8111/viewLog.html?buildId=162&tab=SpecFlow\\_Reports&buildTypeId=bt5](http://.techtalk.at:8111/viewLog.html?buildId=162&tab=SpecFlow_Reports&buildTypeId=bt5). The page title is "AcceptanceTests Test Execution Report". The report was generated by SpecFlow at 05/09/2011 10:49 (see <http://www.specflow.org/>).

**Summary**

Features	Success rate	Scenarios	Success	Failed	Pending	Ignored
5 features	2%	50	1	0	49	0

**Feature Summary**

Feature	Success rate	Scenarios	Success	Failed	Pending	Ignored
<a href="#">Calendar views</a>	0%	8	0	0	8	0
<a href="#">Create booking</a>	3%	30	1	0	29	0
<a href="#">Create reservation</a>	0%	5	0	0	5	0
<a href="#">Show availability of spot length</a>	0%	3	0	0	3	0
<a href="#">View bookings related to utilization value</a>	0%	4	0	0	4	0

**Feature Execution Details**

Feature: [Calendar views](#)

Scenario	Status	Time(s)
Should allow paging backwards in month	pending	12.176
Should allow paging forwards in month	pending	0.227
Should allow switching to month view	pending	0.208
Should allow switching to the booking range	pending	0.168
Should display maximum 31 days when	pending	0.173

Scenario: Should display utilization in green when spot length invalid

@automated @done

Given I am on the screen selection page  
And the calendar shows the booking range from '2011-05-11' to '2011-05-11'  
When I enter 'x' for the spot length  
Then the displayed overall utilization for all screens should be green for every day

# Progressing scenario after scenario

AcceptanceTests Test Execution Report  
Generated by SpecFlow at 05/09/2011 11:49 (see <http://www.specflow.org/>).

### Summary

Features	Success rate	Scenarios	Success	Failed	Pending	Ignored
5 features	4%	50	2	0	48	0

### Feature Summary

Feature	Success rate	Scenarios	Success	Failed	Pending	Ignored
<a href="#">Calendar views</a>	0%	8	0	0	8	0
<a href="#">Create booking</a>	7%	30	2	0	28	0
<a href="#">Create reservation</a>	0%	5	0	0	5	0
<a href="#">Show availability of spot length</a>	0%	3	0	0	3	0
<a href="#">View bookings related to utilization value</a>	0%	4	0	0	4	0

### Feature Execution Details

#### Feature: Calendar views

Scenario	Status	Time(s)
Should allow paging backwards in month view	pending	14.029
Should allow paging forwards in month view	pending	0.182
Should allow switching to month view	pending	0.171
Should allow switching to the booking range view	pending	0.169
	pending	0.176

# Progressing scenario after scenario

The screenshot displays a Test Execution Report for 'AcceptanceTests' generated by SpecFlow on 05/09/2011 at 13:18. The report includes a summary table, a feature summary table with bar charts, and detailed execution details for each scenario.

**Summary**

Features	Success rate	Scenarios	Success	Failed	Pending	Ignored
5 features	6%	51	3	0	48	0

**Feature Summary**

Feature	Success rate	Scenarios	Success	Failed	Pending	Ignored
<a href="#">Calendar views</a>	0%	8	0	0	8	0
<a href="#">Create booking</a>	10%	31	3	0	28	0
<a href="#">Create reservation</a>	0%	5	0	0	5	0
<a href="#">Show availability of spot length</a>	0%	3	0	0	3	0
<a href="#">View bookings related to utilization value</a>	0%	4	0	0	4	0

**Feature Execution Details**

**Feature: Calendar views**

Scenario	Status	Time(s)
Should allow paging backwards in month view	pending	13.217
Should allow paging forwards in month view	pending	0.191
Should allow switching to month view	pending	0.324
Should allow switching to the booking range view	pending	0.202
Should display maximum 31 days when selecting the booking range view	pending	0.172

# Progressing scenario after scenario

The screenshot shows a web browser displaying a CI report for build #167. The URL is [.techtalk.at:8111/viewLog.html?buildId=167&tab=SpecFlow\\_Reports&buildTypeId=bt5](http://.techtalk.at:8111/viewLog.html?buildId=167&tab=SpecFlow_Reports&buildTypeId=bt5). The page title is "AcceptanceTests Test Execution Report".

**Summary:**

Features	Success rate	Scenarios	Success	Failed	Pending	Ignored
5 features	8%	51	4	0	47	0

**Feature Summary:**

Feature	Success rate	Scenarios	Success	Failed	Pending	Ignored
<a href="#">Calendar views</a>	0%	8	0	0	8	0
<a href="#">Create booking</a>	13%	31	4	0	27	0
<a href="#">Create reservation</a>	0%	5	0	0	5	0
<a href="#">Show availability of spot length</a>	0%	3	0	0	3	0
<a href="#">View bookings related to utilization value</a>	0%	4	0	0	4	0

**Feature Execution Details:**

**Feature: Calendar views**

Scenario	Status	Time(s)
Should allow paging backwards in month view	pending	12.952
Should allow paging forwards in month view	pending	0.248
Should allow switching to month view	pending	0.258
Should allow switching to the booking range view	pending	0.355
Should display maximum 31 days when selecting the booking range view	pending	0.171

# Implementing user stories in parallel

The screenshot shows a web browser displaying a test execution report from a Continuous Integration (CI) system. The URL is [.techtalk.at:8111/viewLog.html?buildId=207&buildTypeId=bt5&tab=SpecFlow\\_Reports](http://.techtalk.at:8111/viewLog.html?buildId=207&buildTypeId=bt5&tab=SpecFlow_Reports). The page title is "AcceptanceTests Test Execution Report".

Key elements of the interface include:

- Header:** Shows "Welcome, Guest user" and "Logout".
- Top Navigation:** Projects (dropdown), Agents (1), Build Queue (0).
- Breadcrumbs:** CI > #42 (13 May 11 17:09).
- Tab Bar:** Overview, Changes (7), Tests, Build Log, Build Parameters, Artifacts, SpecFlow Reports (selected).
- Toolbar:** Navigation icons (back, forward, search, etc.) and links for "#41", "All history", and "#43".
- Summary Section:** "AcceptanceTests Test Execution Report" generated by SpecFlow at 05/13/2011 17:13.
- Feature Summary Table:** Shows 5 features with a success rate of 22% (22% green, 78% purple). Details: Scenarios 51, Success 11, Failed 4, Pending 36, Ignored 0.
- Feature Details Tables:** "Feature Summary" table for US110, US112, US114, US115, and US116. "Feature Execution Details" table for US110 showing 5 scenarios: Should allow paging backwards in month view, Should allow paging forwards in month view, Should allow switching to month view, Should allow switching to the booking range view, and Should display maximum 31 days when selecting the booking range view. All scenarios are marked as "success" with times ranging from 1.482 to 42.743 seconds.

# First user story ready for testing

The screenshot displays a Test Execution Report for 'AcceptanceTests' generated by SpecFlow on 05/19/2011 at 16:47. The report includes a summary table, a feature summary table with bar charts, and detailed execution details for a specific feature.

**Summary**

Features	Success rate	Scenarios	Success	Failed	Pending	Ignored
5 features	49%	51	25	0	26	0

**Feature Summary**

Feature	Success rate	Scenarios	Success	Failed	Pending	Ignored
<a href="#">US110 - Calendar views</a>	88%	8	7	0	1	0
<a href="#">US112 - Create reservation</a>	0%	5	0	0	5	0
<a href="#">US114 - Show availability of spot length</a>	100%	3	3	0	0	0
<a href="#">US115 - Create booking</a>	48%	31	15	0	16	0
<a href="#">US116 - View bookings related to utilization value</a>	0%	4	0	0	4	0

**Feature Execution Details**

**Feature: US110 - Calendar views**

Scenario	Status	Time(s)
Should allow paging backwards in month view	success	24.682
Should allow paging forwards in month view	success	2.373
Should allow switching to month view	success	1.901
Should allow switching to the booking range view	success	3.581
Should display maximum 31 days when selecting the booking range view	success	1.725

# Manual testing can start even earlier

The screenshot shows a web-based Continuous Integration (CI) tool interface. The URL is [.techtalk.at:8111/viewLog.html?buildId=295&tab=SpecFlow\\_Reports&buildTypeId=bt5](http://.techtalk.at:8111/viewLog.html?buildId=295&tab=SpecFlow_Reports&buildTypeId=bt5). The page title is "CI > #99 (23 May 11 09:43)". The navigation bar includes "Projects", "Agents (1)", "Build Queue (0)", "Welcome, Guest user", and "Logout". Below the navigation is a breadcrumb trail: "CI > #99 (23 May 11 09:43)". The main content area displays a table of test results for "Feature: US115 - Create booking". The table has columns for "Scenario", "Status", and "Time(s)". A horizontal orange box highlights a group of test cases related to invalid times. Another horizontal orange box highlights a group of test cases related to overbooking calculations.

Scenario	Status	Time(s)
Confirmation of overbooking should save the booking	success	9.504
Invalid date range should cause validation error on the booking details page	success	6.988
Invalid dates should cause validation error on the booking details page("31/05/2011",System.String[])	success	6.218
Invalid dates should cause validation error on the booking details page("2011.05.31",System.String[])	success	6.127
Invalid dates should cause validation error on the booking details page("//",System.String[])	success	6.183
Invalid spot lenght should cause validation error on the booking details page("xxx",System.String[])	success	6.032
Invalid spot lenght should cause validation error on the booking details page("201",System.String[])	success	5.839
Invalid spot lenght should cause validation error on the booking details page("-1",System.String[])	success	6.277
Invalid spot lenght should cause validation error on the booking details page("0",System.String[])	success	5.854
Invalid times should cause validation error on the booking details page("xxx",System.String[])	pending	0.274
Invalid times should cause validation error on the booking details page("1800",System.String[])	pending	0.428
Invalid times should cause validation error on the booking details page("25:00",System.String[])	pending	0.380
Invalid times should cause validation error on the booking details page("08:00AM",System.String[])	pending	0.271
Invalid times should cause validation error on the booking details page("12:61",System.String[])	pending	0.310
Navigation from booking details to utilization view takes over the booking details and the selected attributes and time ranges	success	82.914
Navigation from utilization view to booking details takes over the booking details and the selected attributes and time ranges	success	76.330
None of the required fields filled should cause all required validation errors displayed	success	4.440
None of the required fields filled should cause all required validation errors displayed on select product configuration page	success	1.526
No time ranges selected should raise a validation error	pending	0.347
Overbooking calculation should handle different booking types("Reservation","Pre-reservation","no",System.String[])	pending	0.337
Overbooking calculation should handle different booking types("Booking","Pre-reservation","no",System.String[])	pending	0.321
Overbooking calculation should handle different booking types("Booking","Reservation","yes",System.String[])	pending	0.346
Overbooking calculation should handle different booking types("Reservation","Booking","yes",System.String[])	pending	0.263

# Already done work can break again

The screenshot shows a web browser displaying a test execution report from a Continuous Integration (CI) system. The URL in the address bar is [.techtalk.at:8111/viewLog.html?buildId=301&tab=SpecFlow\\_Reports&buildTypeId=bt5](http://.techtalk.at:8111/viewLog.html?buildId=301&tab=SpecFlow_Reports&buildTypeId=bt5). The page title is "AcceptanceTests Test Execution Report".

The report is generated by SpecFlow at 05/23/2011 16:37 (see <http://www.specflow.org/>).

**Summary**

Features	Success rate	Scenarios	Success	Failed	Pending	Ignored
5 features	53%	51	27	0	24	0

**Feature Summary**

Feature	Success rate	Scenarios	Success	Failed	Pending	Ignored
<a href="#">US110 - Calendar views</a>	88%	8	7	0	1	0
<a href="#">US112 - Create reservation</a>	20%	5	1	0	4	0
<a href="#">US114 - Show availability of spot length</a>	100%	3	3	0	0	0
<a href="#">US115 - Create booking</a>	52%	31	16	0	15	0
<a href="#">US116 - View bookings related to utilization value</a>	0%	4	0	0	4	0

**Feature Execution Details**

**Feature: US110 - Calendar views**

Scenario	Status	Time(s)
Should allow paging backwards in month view	success	24.295
Should allow paging forwards in month view	success	2.427
Should allow switching to month view	success	2.273
Should allow switching to the booking range view	success	3.493
Should display maximum 31 days when selecting the booking range view	success	1.877

# Already done work can break again

The screenshot shows a web browser window with the URL [.techtalk.at:8111/viewLog.html?buildId=308&tab=SpecFlow\\_Reports&buildTypeId=bt5](http://.techtalk.at:8111/viewLog.html?buildId=308&tab=SpecFlow_Reports&buildTypeId=bt5). The page title is "AcceptanceTests Test Execution Report". The report was generated by SpecFlow at 05/24/2011 15:59 (see <http://www.specflow.org/>).

**Summary**

Features	Success rate	Scenarios	Success	Failed	Pending	Ignored
5 features	58%	59	34	1	24	0

**Feature Summary**

Feature	Success rate	Scenarios	Success	Failed	Pending	Ignored
<a href="#">US110 - Calendar views</a>	88%	8	7	0	1	0
<a href="#">US112 - Create reservation</a>	69%	13	9	0	4	0
<a href="#">US114 - Show availability of spot length</a>	67%	3	2	1	0	0
<a href="#">US115 - Create booking</a>	52%	31	16	0	15	0
<a href="#">US116 - View bookings related to utilization value</a>	0%	4	0	0	4	0

**Feature Execution Details**

Feature: [US110 - Calendar views](#)

Scenario	Status	Time(s)
Should allow paging backwards in month view	success	21.132
Should allow paging forwards in month view	success	2.408
Should allow switching to month view	success	2.151
Should allow switching to the booking range view	success	3.729
Should display maximum 31 days when selecting the booking range view	success	1.920

# See what is temporarily not working

The screenshot shows a web browser displaying a Continuous Integration (CI) dashboard for a build. The URL is [.techtalk.at:8111/viewLog.html?buildId=308&tab=SpecFlow\\_Reports&buildTypeId=bt5](http://.techtalk.at:8111/viewLog.html?buildId=308&tab=SpecFlow_Reports&buildTypeId=bt5). The dashboard includes tabs for Projects, Agents (1), Build Queue (0), Overview, Changes (1), Tests, Build Log, Build Parameters, Artifacts, and SpecFlow Reports. The SpecFlow Reports tab is active, showing the results for CI build #108 (24 May 11 15:51). A red error icon indicates a failure.

**Feature: US114 - Show availability of spot length**

Scenario	Status	Time(s)
Should display utilization in green when spot length available	success	2.794
Should display utilization in green when spot length invalid	success	12.665
Should display utilization in red when spot length not available	failure [show]	2.826

**Feature: US114 - Show availability of spot length**

**@current\_sprint**  
LAST UPDATED: 2 MINUTES AGO

**Background**

Given a product with the following properties

Id	Name	BookingVariant	SecondsAvailable
Gas station	Gas stations in Austria	Loop	200

**And the following screens**

Screen
S1

**And the following time ranges**

Name
Morning

**And the following bookings**

Start date	End date	Time range	Duration
2011-05-11	2011-05-11	Morning	100

**Scenario: Should display utilization in red when spot length not available**

**@automated @done**

Given I am on the screen selection page  
And the calendar shows the booking range from '2011-05-11' to '2011-05-11'  
When I enter '101' for the spot length  
Then the displayed overall utilization for all screens should be red for every day

IT WORKS – WE CARE  
**TECHTALK**

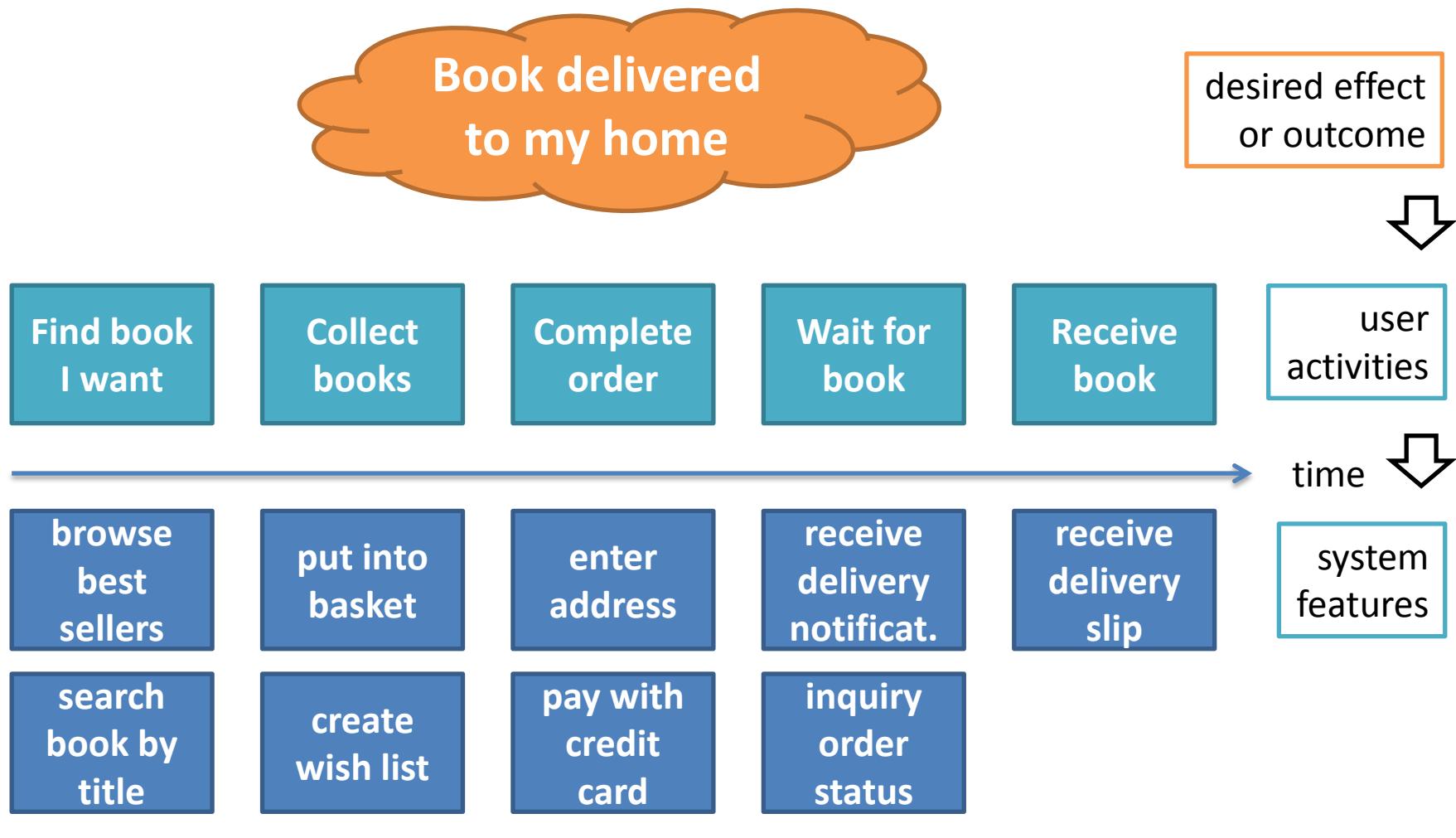
# Living documentation

# Story Maps

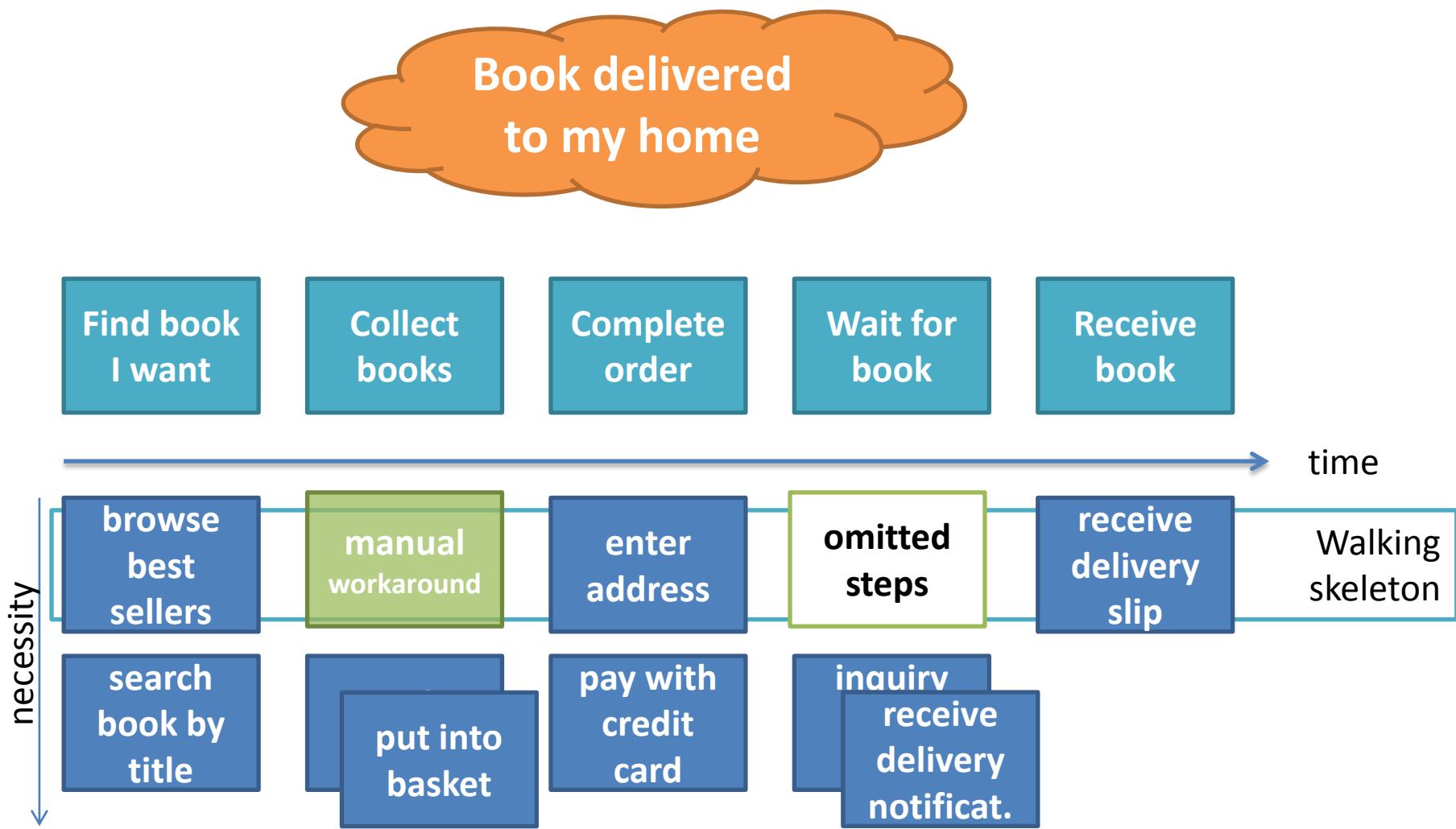
- Developed by Jeff Patton
- Skeleton for fine-grained user stories
- Built for a set of capabilities or outcome that the system should support



# Building story maps



# Establish a walking skeleton

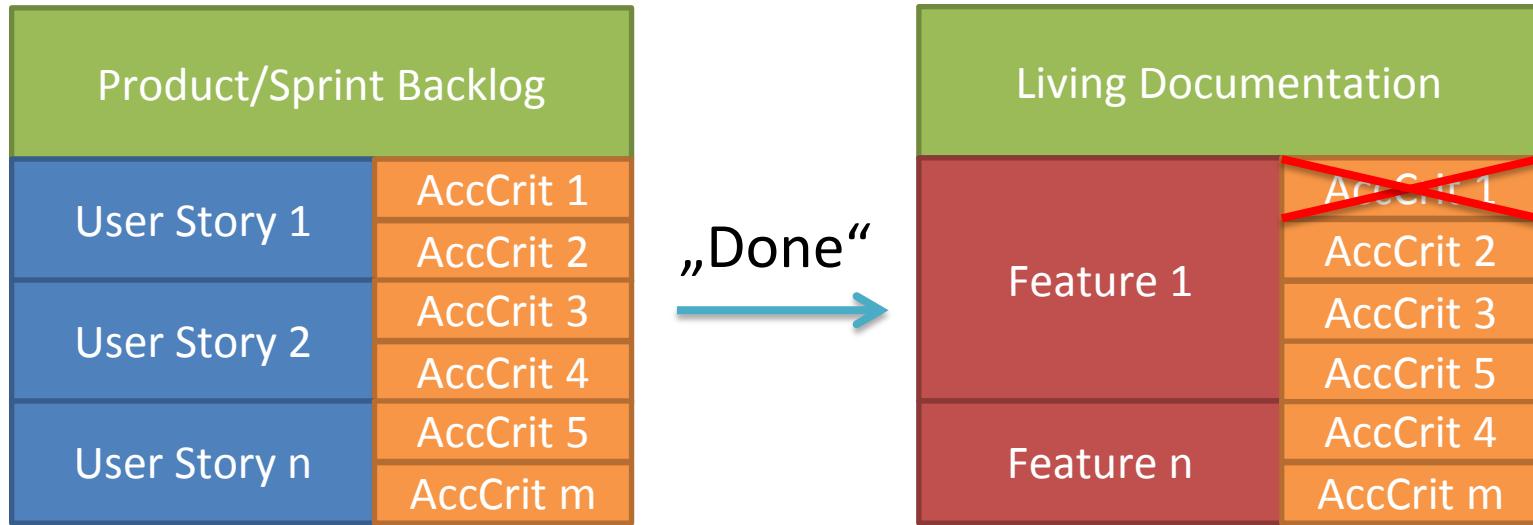




# Demo

- Story Maps as Living Documentation

# User Stories vs. Features



- Future options of the system
- Organized/refined according to priority, value, effort, risk, ...
- Next possible increments of the product (units of work)

- Current state of the system
- Organized/refined for functional overview
- Versioned and maintained together with source code

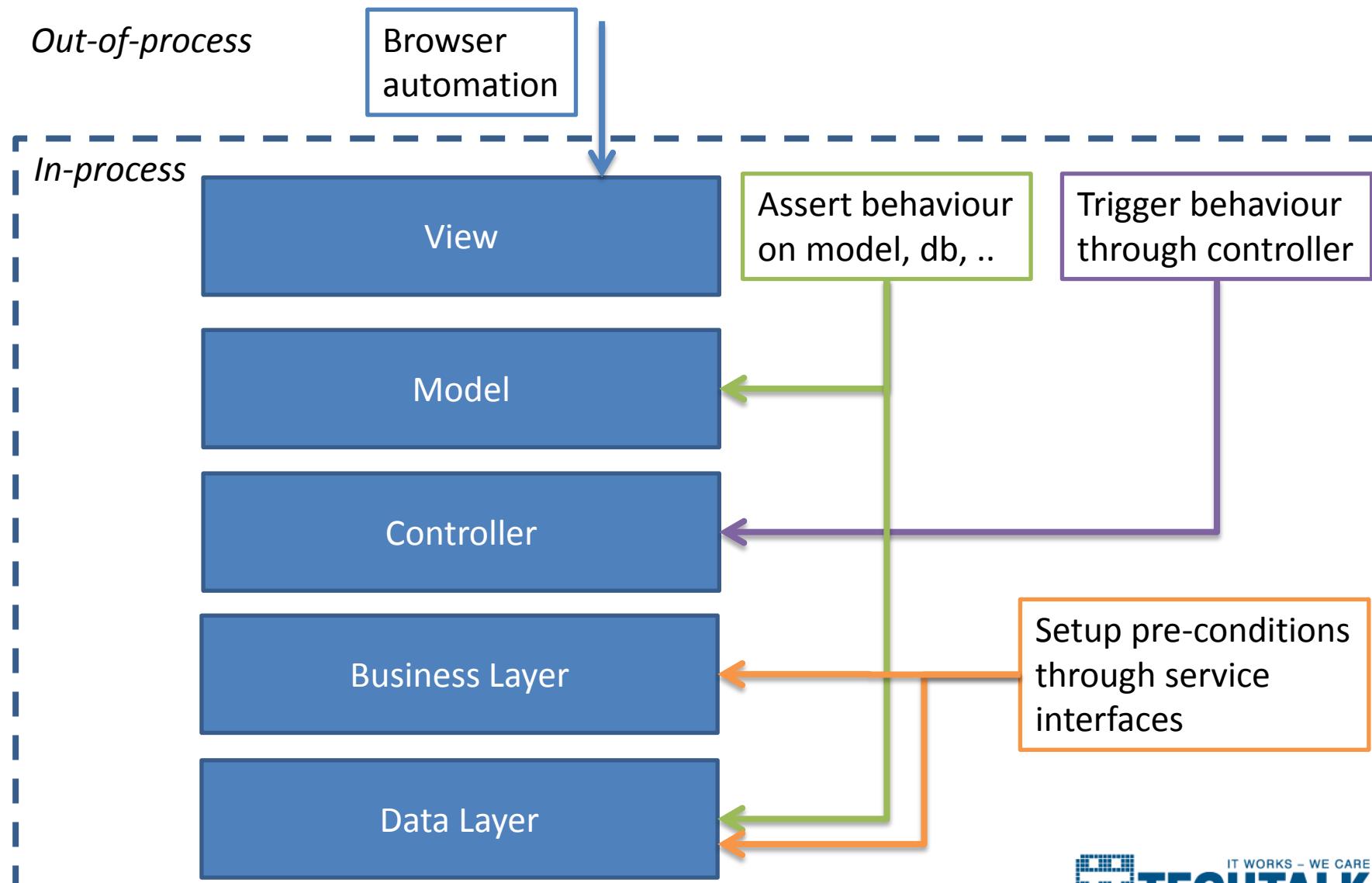


# Demo

- Feature Tree as Living documentation

# Additional thoughts

# Level of automation



# Performance

- Grouping tests
  - Current WIP
  - Completed work
- Database
  - In-memory
  - Templates for setup
- Parallel execution
- Smart execution order



# Internal vs. external DSL

Feature: US01 - Book Search

LAST UPDATED: 4 HOURS AGO

*As a potential customer*

*I want to search for books by a simple phrase*

*So that I can easily allocate books by something I remember from them.*

## Background

Given the following books

Author	Title
Martin Fowler	Analysis...
Eric Evans	Domain...
Ted Pattison	Inside...
Gojko Adzic	Bridgin...

## Scenario: Title should be matched

When I search for books by the phrase 'Domain'

Then the list of found books should contain only: 'Domain Driven Design'

Example Source:

Liz Keogh

[https://github.com/lunivore/tictactoe-java/blob/master/scenarios/com/lunivore/tictactoe/scenarios/Three\\_in\\_a\\_row\\_wins.java](https://github.com/lunivore/tictactoe-java/blob/master/scenarios/com/lunivore/tictactoe/scenarios/Three_in_a_row_wins.java)

```
file | 24 lines (19 sloc) | 0.497 kb
1 package com.lunivore.tictactoe.scenarios;
2
3 import org.junit.Test;
4
5 import com.lunivore.tictactoe.scenarios.util.Scenario;
6
7 public class Three_in_a_row_wins extends Scenario {
8
9     @Test
10    public void horizontal_rows_win() {
11        given_a_game.with_players("Liz", "Sarah")
12            .which_looks_like(
13                "OX." + nl +
14                "OO." + nl +
15                "X.X" + nl);
16        when_i_go.in(2, 1);
17        then_the_game.should_look_like(
18            "OX." + nl +
19            "OOO" + nl +
20            "X.X" + nl);
21        and_it.should_say("Liz has won!");
22    }
23 }
```

# Non-functional acceptance criteria

**Given** there are 100,000 users registered on the system

**When** I create a new account

**Then** I should be taken to my dashboard within 5ms

**Given** 1000 users are hitting the homepage

simultaneously

**Then** each user should get a response within 2ms

Matt Wynne

<http://blog.mattwynne.net/2012/03/13/using-cucumber-for-load-testing>

# Tools

# Gherkin automation tools



[www.cukes.info](http://www.cukes.info)

Ruby, Java, JavaScript, C++



[www.specflow.org](http://www.specflow.org)

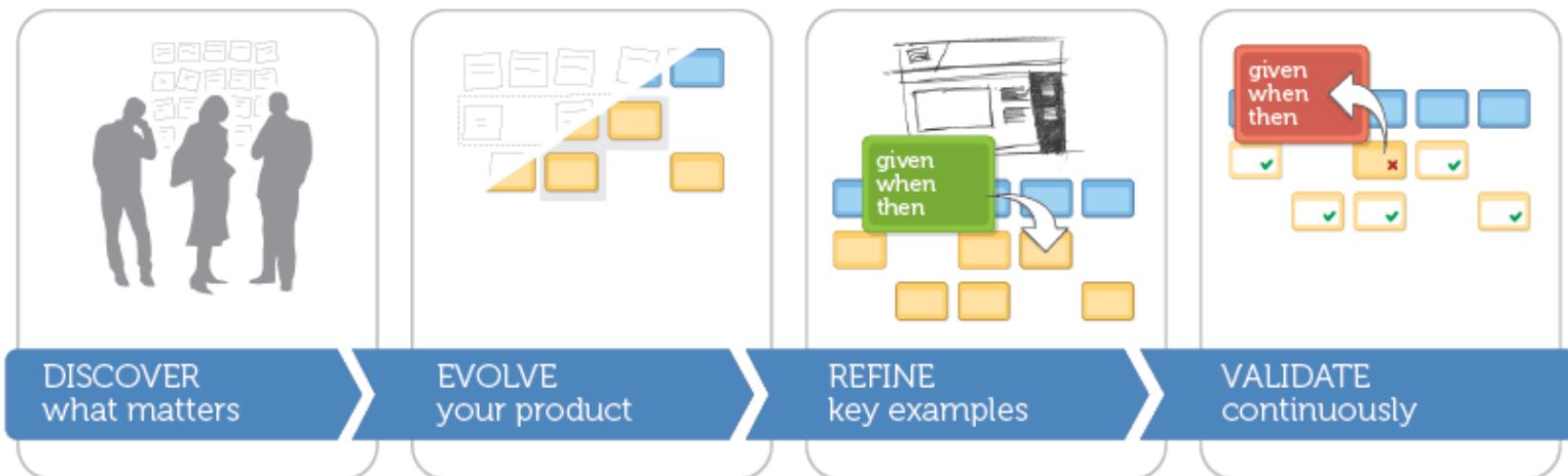
.NET, Mono, Silverlight, WP7



[www.behat.org](http://www.behat.org)

PHP

The tool you are missing to plan iteratively and validate continuously using Impact Maps, Story Maps, Specification-by-Example and Living Documentation. [Available now!](#)



Software that matters generates measurable impact. Collaborate visually with **impact maps** to quickly identify how actors and stakeholders can contribute to measurable goals. **Story maps** help you understand individual user scenarios and how they can be supported iteratively.

Prioritize and slice potential sets of features into valuable increments of your product. SpecLog provides a **virtual space** for your impact maps and story maps, and supports you with **product planning in distributed teams**. Your backlog becomes more than just a prioritized list.

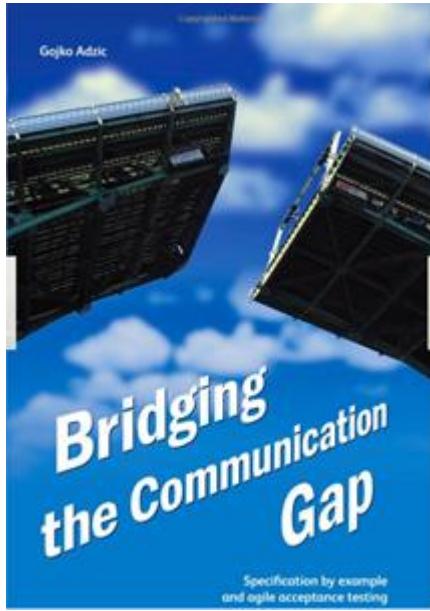
Use examples to develop a **shared understanding** about the domain. Add examples to illustrate user stories that are mapped in SpecLog. Start with UI scribbles, sample artifacts and acceptance criteria, and further refine them to formalized **Gherkin scenarios**.

Extend your system using **ATDD** where the Gherkin scenarios become automated business readable acceptance tests. Linking automated Gherkin scenarios to the maps maintained in SpecLog allows you to build a **continuously validated living documentation** of your system.

[twitter @speclog](#)

[www.speclog.net](http://www.speclog.net)

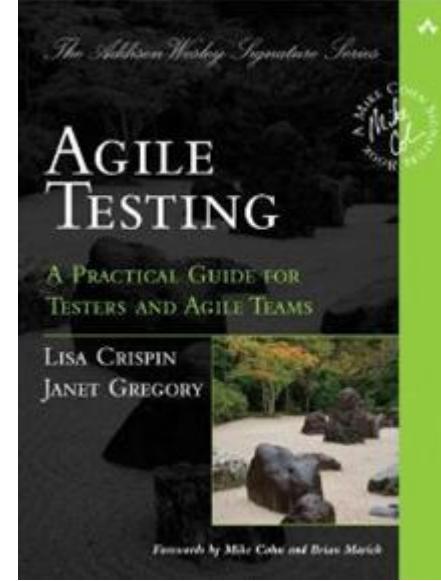
# Books



*Gojko Adzic*  
Bridging the  
Communication Gap



*Gojko Adzic*  
Specification by  
Example



*Lisa Crispin*  
*Janet Gregory*  
Agile Testing

## Customers & Business Stakeholders

- » Define measurable goals and desired impact
- » Identify how actors and stakeholders can support or obstruct these goals
- » Provide key examples of the domain
- » Validate formalized scenarios
- » Give feedback to delivered product increments

Discover what matters  
Deliver what matters,  
and not just what's specified



## Testers

- » Automate formalized scenarios for continuous validation
- » Deliver new increments of the system frequently

- » Extend formalized scenarios to validate edge cases and variations
- » Discover new scenarios and exceptions to be validated by business
- » Identify critical risks that may threaten the product
- » Explore unanticipated functionality

## Developers

- » Validate technical feasibility and propose technical solutions
- » Evolve and refine the architecture
- » Decompose formalized scenarios into unit tests and extend the product to fulfill them

## Business Analysts and UX Experts

- » Challenge requirements
- » Identify and structure user scenarios
- » Derive and prioritize potential features
- » Slice into sets of valuable increments for delivery
- » Design and validate the user experience



RESOURCES  
WWW.IMPACTMAPPING.ORG  
WWW.SPECIFICATIONBYEXAMPLE.COM