

## 9 Lab: Desenvolvimento orientado por user-stories

### Tópicos

Os métodos ágeis defendem a utilização de ciclos curtos de desenvolvimento, designados “iterações”. Cada iteração inclui atividades de refinamento da especificação e do desenho, implementação, integração dos novos incrementos na base anterior e a validação frequente com o cliente. O método SCRUM, não sendo um método de desenvolvimento de software mas de gestão de projetos, ajusta-se bem às ideias dos métodos ágeis, havendo uma correspondência direta de conceitos (*sprint* da SCRUM ~ iteração do *Agile*).

### Tarefas

Este Lab dá continuidade aos conceitos explorados na apresentação sobre SCRUM realizada pela Nokia, na aula TP.

Uma parte essencial de um projeto que aplica métodos ágeis/SCRUM é a gestão do *backlog*, a lista do trabalho para ser feito.

Neste Lab, vamos usar o conceito de *user story* para modelar cada entrada do *backlog* (i.e., o *backlog* é uma pilha de *user stories*, ordenadas por prioridade). As *user stories* podem ser derivadas dos casos de utilização ([saber mais](#)).

Conceito de *user story*:

*A story represents one small concrete deliverable (or task) that provides verifiable business value.*

*Typically, on a software project, it is a concise description of some functionality (from the perspective of the user); it must be understood by everyone on the team (developers, product owners, business stakeholders, testers) and describes something that can be verified and accepted by the project's customer. It must be clear to everyone what 'done' means.*

*A story can also be seen as one way (out of several) to walk through a use case.*

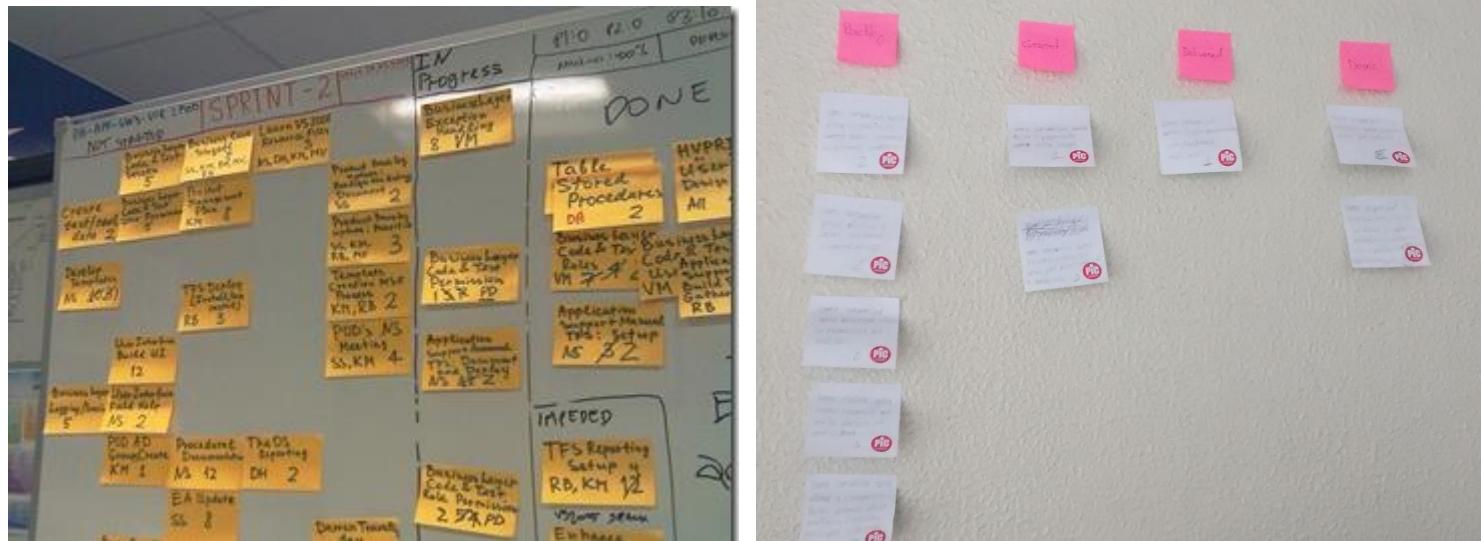
*E.g.: 'Customer adds a product to his shopping cart'; 'Shopper pays items with VISA credit card'; 'Shopper shares multiples items in his Wish List with friends using Google+'.*

#### E9.1

As a frequent flyer, I want to rebook a past trip, so that I save time booking trips I take often

- a) Suponha que a sua equipa se reuniu para a “*Sprint planning meeting*”. Para começar, a equipa vai precisar de um conjunto de *stories* que representam o trabalho (funcionalidade a desenvolver). É aqui que o *Product Owner* e a equipa vão colocando questões para clarificar o trabalho em cada uma delas. As ambiguidades sobre os requisitos são rapidamente dirimidas. Crie várias *stories* para o seu projecto (>10). Sugestão: procure fluxos significativos “dentro” dos casos de utilização. Escreva cada *story* num *post-it* (sem a parte dos critérios de aceitação). Veja a secção “Como escrever uma *story*” adiante.
- b) Prepare uma área na parede/painel/quadro, com 4 colunas (imaginárias)

identificadas: "Backlog", "Current", "Delivered", Done" (pode-se usar a parede da sala ou as mesa).



- c) Para cada *story* do seu projeto, estime uma pontuação (1, 2, 3 ou 4). A pontuação reflete o esforço relativo que a equipa antecipa com o desenvolvimento daquela *story* (1: fácil ... 4: muito difícil).
- d) Priorize as *stories* (no topo, as mais prioritárias); a prioridade deve refletir a importância que têm para o negócio e conveniência de validar cedo as *features* críticas. Distribua as *stories* pelas colunas do *backlog* e *current*, em que *current* são as *stories* que vão ser implementadas na presente iteração. A soma dos pontos das *stories* na coluna *current* não pode passar um patamar de **velocidade** configurado no projeto (admita que o limite é 10, i.e. a equipa consegue trabalhar à velocidade de 10 pontos por iteração).
- e) Escreva, para duas *stories*, os respetivos critérios de aceitação. Veja a secção "Como escrever uma *story*".
- f) Admita que se passaram alguns dias e algumas das *stories* foram implementadas. Algumas foram implementadas e entregues, mas aguardam o OK do *Product Owner*; outras, já têm o OK e estão consideradas feitas. Mova os post-its correspondentes respetivamente para as colunas "Delivered" e "Done".

## E9.2

Vamos procurar replicar o *workflow* anterior, mas com uma ferramenta ([PivotalTracker](#)).

- a) Cada membro da equipa deve criar um perfil no PivotalTracker (livre), com e email da UA.
  - b) Apenas um membro: criar um novo projeto no PivotalTracker (um para o grupo). Adicione os outros membros na configuração do projeto (utilize o email para identificar).
  - c) Configurar a duração de cada iteração, nas configurações do projeto (e.g.: 2 semanas por iteração).
  - d) Adicione as *stories* ao "icebox" (as *stories* devem ser do tipo *feature*). Para cada *story*, registe, pelo menos:
    - O título (que vai aparecer nas listas)
    - Na descrição: uma narrative curta no formato *As a [role] I want [feature] So that [benefit]*
    - Os critérios de aceitação, para algumas *stories*, podem ser também inseridos no campo de descrição.
- [Saber mais sobre o [formato de uma \*story\* completa](#)]

- e) Depois, estime o esforço (dar pontos às *stories*). Priorize as *stories*, movendo-as para o Backlog e ordenando na pilha para refletir as prioridades. O Pivotal vai automaticamente selecionar um conjunto de *stories* para encher o “Current” (o current corresponde ao topo do backlog, para a iteração em curso).
- f) Simule o seguinte cenário:  
Um dos *developers* da equipa inicia o desenvolvimento de uma *story* usando o botão “Start”.  
Enquanto trabalha nesta *story*, surgem dúvidas, que geram um *thread* de discussão no contexto desta *story* (dentro da ficha da *story*).  
Passado algum tempo, o *developer* para o desenvolvimento (botão Stop) e integra com o resto do projeto (botão Delivery).  
Outro membro da equipa, funcionando como o *product owner*, faz a aceitação do trabalho (botão Accept).

Experimente livremente o [workflow do PivotalTracker](#).

### E9.3

Os critérios de aceitação definidos nas *stories* são usados para criar testes de aceitação. Os testes de aceitação simulam a utilização da aplicação ao nível do utilizador final, ativando o mesmo tipo de inputs e seleções, e verificam se o sistema produz, na interface do utilizador, os resultados esperados.

Considere a seguinte *user story* e os seus critérios de aceitação:

**Story:** Pesquisa por nome de autor

Como um visitante do site da livraria

Quero pesquisar por nome de autor

De modo a ver bibliografia e novidades de um autor.

**Cenário 1:** Pesquisa com sucesso

Dado que estou na página de entrada da Fnac.pt

E insiro o nome do autor “Valério Romão” no campo de pesquisa

Quando seleciono o botão para iniciar pesquisa

Então a página de resultados inclui “Valério Romão” no título

E existe um livro chamado “Autismo” na lista

E existe um livro chamado “Cair Para Dentro” na lista.

**Cenário 2:** Pesquisa sem resultados

Dado que estou na página de entrada da Fnac.pt

E insiro o nome do autor “askjfdenk kjewnjknkdsjnf kjdsakfjjnj” no campo de pesquisa

Quando seleciono o botão para iniciar pesquisa

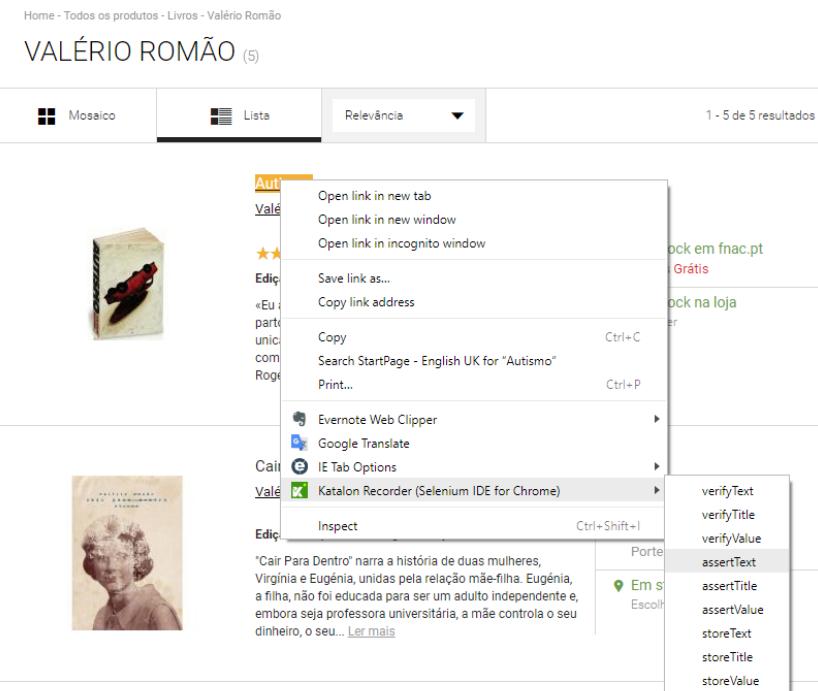
Então a página de resultados inclui “askjfdenk kjewnjknkdsjnf kjdsakfjjnj” no título

E existe a menção “Não há resultados para a tua pesquisa” na página

Vamos agora criar um plano de teste que possa automatizar o teste de aceitação (que será executado muitas vezes).

- a) No Chrome/Chromium/Firefox, instale o *plug-in* [Katalon Recorder](#). Poderá ver como se utiliza esta ferramenta [neste vídeo](#).
- b) Crie um novo caso de teste (*test case*) no Katalan, chamado “pesquisa-valério-5”.

- Selecione este teste e inicie o modo de gravação (Record).
- No browser, execute passo-a-passo o cenário 1 da *user story* exemplificada: ir ao site da [www.fnac.pt](http://www.fnac.pt), inserir o texto de pesquisa, selecionar a “lupa” para pesquisar, etc. As ações devem ser gravadas pelo Katalan. Pode fazer Play do teste para confirmar as respetivas ações.
  - Tratando-se de um teste, precisamos de incluir elementos que verifiquem se o resultado é o esperado. Uma forma fácil é, durante o modo de gravação, selecionar



elementos da página web e no menu de contexto (do rato) aceder à opção “Katalon Recorder” para incluir “asserts”.

O resultado deve ficar algo parecido com o ilustrado a seguir.

Test Suites					Command	Target	Value
livaria-fnac-suite*	...				open	https://www.fnac.pt/	
• results-valerio-5 *					type	id=Fnac_Search	Valério Romão
Untitled Test Case *					click	//button[@type='submit']	
no-results *					click	//div[3]/div/div[2]	
					assertTitle	Valério Romão, uma pesquisa em Livros na Fnac.pt	
					assertText	link=Autismo	Autismo
					assertText	link=Cair Para Dentro	Cair Para Dentro

- Depois de ter confirmado que consegue escrever um teste para o cenário de sucesso, considere também o cenário em que não há resultados.
- Execute o “test suite” para confirmar que criou uma pequena base com testes de aceitação.

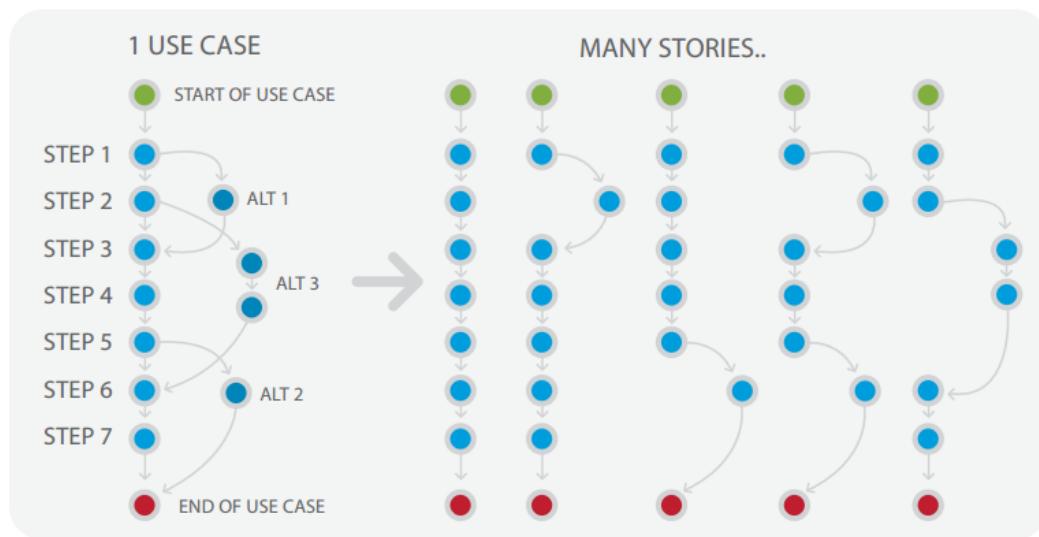
## Como escrever uma story

Um *story* descreve um pequeno incremento à solução, mas relevante para os utilizadores e para os *developers*.

Para identificar as *stories*, podemos começar por olhar para os casos de utilização (começando pelos mais importantes/centrais). A análise do fluxo principal e depois os fluxos alternativos deve dar várias *stories*.

A story é apresentada no formato *As a [role] I want [feature] So that [benefit]*

Uma *story* completa inclui também critérios de aceitação, que podem ser escritos com o estilo Given/Then/When.



**Figura 1:** um caso de utilização dá origem a várias formas de o percorrer (→ user stories).

**Tabela 1- Do caso de utilização para as stories.** Uma *story* deve incluir exemplos representativos que funcionarão como critérios de aceitação da sua implementação.

Use case (episode of use related to some actor's objective. → defined by the Analyst)	User stories (a unit of work to assign to the development team that reflects some useful increment from the business perspective. → defined collectively)	Acceptance criteria (specific sample situations to verify the expected functional behavior)
Add items to purchase	<b>Shopper can add an item to cart</b> <b>As a</b> shopper, <b>I want to</b> add a new product to the shopping cart, <b>so I</b> can build my list	<p>Scenario 1:  Given that I'm a logged in user  When I'm on the product page for "Perfume Gift Set"  And I click the "Buy" button  Then I should see the item count for my cart increment by 1  And the running total should be updated</p> <p>Scenario 2:  Given that I'm a logged in user  And there is no stock for "Perfume Gift Set"  When I'm on the product page for "Perfume Gift Set"  And I click the "Buy" button  Then the item count for my cart increment is not modified</p>

<b>Use case</b> (episode of use related to some actor's objective. → defined by the Analyst)	<b>User stories</b> (a unit of work to assign to the development team that reflects some useful increment from the business perspective. → defined collectively)	<b>Acceptance criteria</b> (specific sample situations to verify the expected functional behavior)
		<p>And a option to add the item to a wish list should pop up</p> <p>Scenario 3:</p> <p>Given that I'm an anonymous visitor When I'm on the product page for "Perfume Gift Set" And I click the "Buy" button Then I should see the item count for my cart increment by 1 And warning message should be displayed to notify the user he/she must create a profile.</p>
	<p><b>Shopper can modify itens in the cart</b></p> <p>As a shopper, I want to remove the products or change quantities in my shopping cart, so I can review my purchase</p>	...
	<p><b>Shopper can empty the cart</b></p> <p>As a shopper, I want to empty the shopping cart, so i can cancel a purchase.</p>	...

**Tabela 2: Esquema para definir critérios de aceitação da story.**

<b>Scenario 1: Title</b> <b>Given [context]</b> <b>And [some more context]...</b> <b>When [event]</b> <b>Then [outcome]</b> <b>And [another outcome]...</b>	Scenario 1: Account balance is negative Given the account's balance is below 0 And their is not a scheduled direct deposit that day When the account owner attempts to withdraw money Then the bank will deny it And send the account owner a nasty letter.
--	--