

Enoncé

Document C : Contexte

Vous travaillez sur le portail Web « Manny’ s food service customer », un service de restauration en ligne. Différentes US ont été rédigées de manière imparfaite (colonne 1. Contre-exemples) et les bonnes pratiques de « Suscheck-Fuqua » leur ont été appliquées pour les transformer (colonne 2. Exemples).

Document T : Tableau « Suscheck-Fuqua » - 9 bonnes pratiques concernant les récits utilisateurs (ou user stories)

BP ID	Texte de la bonne pratique
A	<p>Excessive ‘So That’ (Misplaced Requirement) Problem : The team typically expects details of a story to be found before the conjunction “so that,” which is used to explain the story’s value. When the product owner writes requirements in the “so that” section, it is easy to miss part of the real requirement since it’s hidden after “so that.” You can tell if there is a problem when the “so that” section is complex or has multiple parts. In this case, the true story may be so big that it can’t get done in a sprint. Improvement: Move the misplaced needs (from « so that » to « I want ») and analyze a bit more to find a real business value to be written after « so that ».</p>
B	<p>Odyssey (Ultra-Huge Story) Problem : If team members can’t even estimate a story at a gross level, they may take on too much work in the iteration and not get to done. An Odyssey is beyond an epic. It’s something compounded or diffused to the point of having no discernable value (which an epic has). Such a story will lead to long conversations with the product owner, or even a failure to get to done. The team’s velocity will be unpredictable, and team members might be frustrated with a constantly evolving story. The story is too very large to be estimated and hard to implement in a sprint. It’s difficult to see the value that holds this story (or epic) together. Improvement: Stories containing “and” or “or” are likely candidates for splitting into several smaller stories. Improvement: Split the large story in small ones.</p>
C	<p>Waterfall Problem : Stories that contain only analysis, design, or technical aspects lead to waterfall development in two-week phases. The story has no business value and a user who is not really a system end user. There is only the technical side of the equation. This story will likely lead to coding, and no testing since it has no business value. Improvement: Remove this developer story and split it in many tasks</p>
D	<p>Rigidity (Inflexible) Problem : Stories with too much detail are often inflexible, leaving little room for creativity, better solutions, or dynamic scope control during development. Avoid this by postponing decisions on details that constrain the solution or that specify an implementation. Defer these decisions to a later but still responsible moment in order to maintain maximum flexibility. The product owner is likely to get exactly what he asks for instead of what he really needs. Improvement : There may be better solutions that leave more flexibility. Specifying this level of technical detail handcuffs the programmers and possibly limits innovation.</p>
E	<p>For Whom? (Non-User) Problem : There are a lot of different types of non-user stories, like using a specific name, a role, or the system. Improvement : It’ s better to write user stories for the role that actually wants the benefit or value provided by the user story.</p>
F	<p>For Whom? (Generic User) Problem : the most overused and overly vague persona: “As a user” and its variants. Another non-user is “the system.” Beginning the user story with the phrase “As the system” may enable teams to use a waterfall approach. The system doesn’ t care if business value is delivered or not. With the waterfall approach, no business value is delivered until the end. Improvement : Knowing the specific role for a story helps us understand the context for the story, leading to better value and focus.</p>
G	<p>Parakeet Value (valeur perroquet) Problem : Beware when the story’ s “so that” phrase is a restatement of the story’ s “I want” phrase. This points to a lack of analysis depth. The real goal of the story is not at all clear, making it easy to be off target and develop software that doesn’ t maximize business value. Such stories are often not analyzed well enough. Improvement : remove the « So that » part and change it to real business value</p>
H	<p>Techie Value Problem : Beware of user stories where “so that” is a technical capability, not any value to the end user. This is very similar to no business value, but the value listed is technical. Improvement: The improvement would be to delete this story.</p>

Question

- Q1. Indiquer dans la colonne « 2 – bonne pratique (BP) » le code (1 lettre) de la bonne pratique appliquée pour transformer le contre-exemple en un exemple (prises parmi les 8 bonnes pratiques explicitées dans le tableau « Suscheck-Fuqua »).
- Q2. Pour les cas 3 à 9, proposer une correction de la User's Story en application du *pattern*.

Document Q

0.ID	1. Contre-exemple (Naif)	2. BP	3. Exemple (Amélioré)
1	As a Manny's food service customer, I need to save, copy, print, and email my list so that I can edit it again, check a received shipment against a printed list, and send the list to a restaurant.		<ul style="list-style-type: none"> • As a Manny's food service customer, I need to save my list so that I can reorder from the list to create more accurate food orders. • As a Manny's food service customer, I need to copy my list so that I can use it as a starting point for creating another list. • As a Manny's food service customer, I need to print my list so that I can check a received shipment against the printed list. • As a Manny's food service customer, I need to email my list so that I can have someone who doesn't use the system review my list.
2	As a tester, I want to have detailed test plans so that when the system is completed, I can test the system.		Remove this developer story and split it in many tasks
3	As a business user, I would like a report of item profitability so that I can identify and highlight profitable items and consider what to do about underperforming items.		
4	As a Manny's food service customer I need to save my list so that later I can email the list for other uses.		
5	As a Manny's food service customer, I want to see different food item types displayed in different colors—RGB = #FF0000 for meats, #A52AFA for grains, and #808000 for vegetables and fruits—so that I can quickly identify my food items by food type.		
6	As a the marketing manager, I would like logins to time out and log off after a preset number of seconds in case users leave their computers unattended.		
7	As a developer, I want to finalize the database table changes and additions for the release so that we don't have to make changes to the model later.		
8	As the system, I need to store customer account info and their order lists in the database.		
9	As a customer ordering food, I want to locate previous food order lists so that I can see all the lists that I have.		