

Le récit utilisateur (user story)

Table des matières

I. Contexte	3
II. Les user stories en pratique	3
A. Les méthodes agiles et les user stories	3
B. Qu'est-ce qu'une user story ?.....	3
C. Pourquoi est-ce important de rédiger des user stories ?	4
D. Introduction à la méthodologie Agile	5
E. Exercice : Quiz	5
III. La structure des user stories	6
A. La règle du qui, quoi, pourquoi ?	6
B. Les critères d'acceptation d'une user story.....	7
C. Comment intégrer les user stories aux spécifications techniques ?	8
D. Exercice : Quiz.....	9
IV. L'essentiel	10
V. Auto-évaluation	10
A. Exercice	10
B. Test.....	10
Solutions des exercices	11

I. Contexte

Durée : 1 h

Environnement de travail : PC et connexion internet

Prérequis : aucun

Contexte

La rédaction de *user stories* est très souvent utilisée par les équipes qui utilisent des méthodes de gestion de projet agiles telle que le Scrum, par exemple. Elles sont utilisées pour décrire les fonctionnalités d'un produit du point de vue de l'utilisateur en langage naturel. Elles permettent de faciliter le développement en sprint, ou itérations. Dans ce cours, vous allez donc vous préparer à la programmation professionnelle grâce à l'utilisation d'une méthode de gestion de projet, car cette compétence est capitale en entreprise. Aussi, il est bon de savoir que développer vos projets informatiques en utilisant cette méthodologie vous permettra également de réaliser des économies en matière de coût de projet et de temps de réalisation du projet. Dans un premier temps, nous allons donc définir ensemble ce qu'est une user story. Puis dans un second temps, nous passerons directement à la pratique et créerons nos premières user stories. Enfin, nous découvrirons des méthodes de testing qui nous permettront de mesurer l'efficacité des user stories.

II. Les user stories en pratique

A. Les méthodes agiles et les user stories

Il existe un lien très fort entre les user stories et les méthodes agiles.

Les méthodes agiles sont des approches basées sur les itérations. C'est-à-dire que nous développons une fonctionnalité par sprint plutôt qu'en un seul bloc. Par essence, ces méthodes sont également plus collaboratives et articulées autour d'un seul et même objectif : livrer continuellement des fonctionnalités.

Les user stories sont une composante clé des méthodes agiles, car elles permettent de décrire les besoins des utilisateurs de manière concise et compréhensible par l'ensemble de l'équipe en charge d'un projet.

En effet, elles permettent à l'équipe de développement de travailler en étroite collaboration non seulement avec les clients, les utilisateurs, mais plus globalement avec toute l'équipe en charge du projet : designers, webdesigners ou marketeurs par exemple. Les user stories sont donc un excellent moyen de communiquer les besoins de l'utilisateur, car celles-ci sont écrites en ce qu'on appelle du langage métier, plutôt qu'en jargon informatique.

Définition Langage métier

On appelle « langage métier » toute explication réalisée en langage naturel.

B. Qu'est-ce qu'une user story ?

Le lien entre les méthodes de gestion de projet agiles et les user stories étant fait, nous pouvons désormais nous intéresser à l'essence même des user stories. Voyons plus en détail ce qu'est une user story et quelle est son utilité à travers des cas pratiques.

Une user story est une méthode que l'on utilise afin de décrire les besoins de l'utilisateur dans le cadre du développement de produits informatiques. C'est une technique de communication simple et très efficace qui permet à l'ensemble de l'équipe en charge d'un projet de comprendre les exigences concernant le produit. Grâce aux user stories, l'équipe développe les fonctionnalités produit du point de vue de l'utilisateur final.

Pourquoi est-ce si efficace ?

Comme nous l'avons vu précédemment, la particularité des user stories est qu'elles décrivent les fonctionnalités d'un produit informatique du point de vue de l'utilisateur final plutôt que d'un point de vue technique. Ainsi, en se concentrant sur les besoins de l'utilisateur pour élaborer un logiciel par exemple, nous pouvons nous assurer de créer un logiciel qui répond véritablement aux attentes du client. Par leur structure, elles permettent également de décomposer des besoins complexes en petites tâches et elles respectent la méthode de développement en sprint si spécifique aux méthodes de gestion de projet agiles. Cette approche est également essentielle lors de la planification d'un projet, car cette découpe en petites tâches permet un suivi plus facile de l'avancement d'un projet. Enfin, dans le cadre de la gestion de projet, les user stories sont idéales car, de par leur vocabulaire simple et non technique, elles sont également compréhensibles par l'ensemble de l'équipe chargée d'un projet. Par conséquent, toutes les parties prenantes du projet peuvent se comprendre et donc collaborer.

C. Pourquoi est-ce important de rédiger des user stories ?

Lors de la création d'un projet informatique, nous réalisons ce que nous appelons des spécifications techniques. Le problème avec les spécifications techniques, c'est qu'elles sont souvent complexes et difficiles à comprendre pour l'ensemble de l'équipe en charge d'un projet.

En effet, sur un même projet, il peut y avoir un web designer et des développeurs. Les spécifications techniques sont souvent compréhensibles par les développeurs uniquement. Ce qui pose un problème, puisque cela peut priver l'équipe d'avis très précieux qui pourraient provenir de profils non techniques, comme les designers. C'est là qu'interviennent les user stories.

Elles sont généralement plus faciles à comprendre pour l'ensemble des personnes qui travaillent sur le projet. De plus, en décrivant les besoins des utilisateurs en langage métier, plutôt qu'en employant du jargon technique, il est plus facile de comprendre le besoin réel de l'utilisateur final. Cela évite de s'éparpiller, car le développement est réalisé en gardant constamment en tête les besoins de l'utilisateur final.

En quelques mots, les user stories permettent donc de :

1. **Faciliter la collaboration dans une équipe** : comme nous l'avons vu, les user stories sont très utiles dans le cadre de la collaboration entre des membres qui travaillent sur un seul et même projet, car toute l'équipe peut comprendre les fonctionnalités développées et leur utilité. Cela permet donc de favoriser les retours tout au long du processus de développement.
2. **Faire économiser du temps** : l'autre force des user stories est leur incroyable utilité dans la gestion de projet, notamment concernant la gestion du temps de réalisation d'un projet. En effet, développer un produit informatique peut demander beaucoup de temps et de ressources. Souvent, ce n'est pas lié à la complexité technique du projet, mais plutôt à sa mauvaise définition. Après coup, il arrive que les équipes se rendent compte que le produit ne correspond pas aux besoins de l'utilisateur final. Certains investissent donc des mois de développement pour un résultat qui ne correspond pas aux réels besoins de l'utilisateur. La force des user stories tient en un mot : itération. Par exemple, dans le développement de logiciel, il est crucial d'obtenir des retours réguliers et fréquents des utilisateurs tout au long du développement. Ceci permet de s'assurer que le produit final correspond bien à leurs besoins et attentes. Exemple : après avoir créé des user stories, l'équipe développe peu à peu les fonctionnalités. À chaque étape, un panel d'utilisateurs est invité à tester la fonctionnalité développée et à donner son avis. Cette manière de faire permet d'obtenir des retours plus réguliers et tout au long du processus de développement. C'est plus efficace que d'attendre la fin du projet pour obtenir des commentaires, car grâce au développement successif des fonctionnalités et des nombreux tests qui en découlent, il sera plus facile de rectifier le tir si nécessaire.
3. **Faire économiser de l'argent** : grâce aux user stories, nous pouvons également réduire les coûts de développement et accélérer celui-ci en choisissant de se concentrer d'abord uniquement sur les fonctionnalités les plus importantes pour l'utilisateur final. Cela permet de créer ce qu'on appelle un MVP, pour *Minimal Valuable Product*.

Définition MVP

Un MVP est une version d'un produit ou d'un logiciel qui comporte uniquement les fonctionnalités principales d'un produit. L'objectif est de recueillir les commentaires des utilisateurs pour guider le développement. Cela évite d'allouer du temps et de l'énergie à des fonctionnalités qui ne sont pas si importantes que cela pour l'utilisateur final. Se faisant, comme nous nous concentrons sur l'essentiel, nous pouvons également accélérer le processus de développement.

D. Introduction à la méthodologie Agile

Avant de découvrir comment construire concrètement vos user stories, nous allons clôturer ce chapitre en réalisant une petite introduction à la méthodologie Agile.

E. Exercice : Quiz

[solution n°1 p.13]

Question 1

MVP signifie *Minimum Valuable Product*. C'est le fait de créer un produit digital, pas entièrement terminé mais qui est pleinement fonctionnel.

- ☐ Vrai
- ☐ Faux

Question 2

Les user stories permettent de retranscrire les besoins d'un utilisateur en langage naturel plutôt qu'en langage technique.

- ☐ Vrai
- ☐ Faux

Question 3

Qu'est-ce qu'une user story ?

- ☐ C'est le storytelling d'un utilisateur
- ☐ C'est une fonctionnalité décrite depuis le point de vue de l'utilisateur
- ☐ Aucune de ces réponses

Question 4

Pourquoi les user stories font-elles partie des méthodes de développement agiles ?

- ☐ Elles respectent un des principes de base de la méthodologie agile qui consiste à développer un produit digital par itération ou sprint
- ☐ Elles permettent aux équipes de développement de se concentrer sur les tâches les plus complexes et les plus importantes pour l'utilisateur final, plutôt que de gaspiller leur temps sur des tâches moins critiques
- ☐ Elles permettent de décrire les fonctionnalités d'un produit de manière très technique, ce qui est essentiel dans la création d'un produit informatique

Question 5

C'est important de garder constamment en tête les besoins de l'utilisateur final lors de la conception d'un projet informatique.

- ☐ Vrai
- ☐ Faux

III. La structure des user stories

A. La règle du qui, quoi, pourquoi ?

Plusieurs structures prêtes à l'emploi permettent de définir des user stories. Bien qu'il soit possible de créer ses propres structures en fonction des besoins spécifiques d'un projet, celles-ci peuvent également servir de guide pour s'assurer que nos propres user stories seront efficaces.

Outre les structures prêtes à l'emploi existantes, il y a une règle très connue qui, si vous la respectez, vous permettra de vous assurer que votre user story sera efficace.

Mais avant cela, la question que nous pourrions nous poser est la suivante : qu'est-ce qu'une user story efficace ?

Une user story efficace est une user story **claire, concise et facile à comprendre** par l'ensemble de l'équipe projet.

Pour ce faire, celle-ci doit respecter la règle suivante : Qui ? Quoi ? Pourquoi ?

La règle du qui, quoi, pourquoi est la plus courante. Elle est applicable quelle que soit la structure que vous choisirez. Elle sert véritablement de fil conducteur.

- Le « *qui* » désigne l'utilisateur ou la personne qui a besoin de la fonctionnalité. Cela peut être un utilisateur, un client, etc. Pour faire simple, elle répond à la question : qui a besoin de cette fonctionnalité ?
- Le « *quoi* » : il décrit le besoin de l'utilisateur /la fonctionnalité. Cette partie doit être écrite du point de vue de l'utilisateur en utilisant un langage simple et compréhensible (ce qu'on appelle du langage métier). Elle répond à la question : qu'est-ce que l'utilisateur veut accomplir ?
- Enfin le « *pourquoi* ». Cette partie décrit la raison pour laquelle l'utilisateur a besoin de cette fonctionnalité. Elle doit très clairement expliquer le bénéfice que l'utilisateur retire de cette fonctionnalité. Elle répond à la question : pourquoi l'utilisateur a-t-il besoin de la fonctionnalité ?

En combinant ces 3 éléments, la user story peut décrire clairement et de manière succincte le besoin d'un utilisateur.

Voyons maintenant quelques exemples de structure :

- En tant que [rôle], je souhaite [fonctionnalité], afin de [bénéfice],
- En tant que [rôle], je souhaite [fonctionnalité] pour [raison],
- En tant que [rôle], je souhaite [fonctionnalité] afin de pouvoir [tâche à accomplir],
- En tant que [rôle], je souhaite [fonctionnalité] parce que [raison],
- En tant que [rôle] je souhaite [fonctionnalité] pour [bénéfice], afin de [raison].

Exemple

Illustrons ce cas pratique avec la user story suivante.

En tant qu'utilisateur, je veux pouvoir m'inscrire sur le site web pour pouvoir accéder aux fonctionnalités réservées aux membres, car cela me permettra d'interagir sur le forum.

Ce que nous pouvons voir, c'est que dans cette user story, nous retrouvons les 3 composantes citées précédemment :

- Qui : l'utilisateur,
- Quoi : s'inscrire sur le site web,
- Pourquoi : accéder aux fonctionnalités réservées aux membres et interagir sur le forum.

Dans cette user story, le besoin de l'utilisateur est bel et bien communiqué de manière claire et concise.

Méthode

Pour réaliser vos user stories de manière efficace, plusieurs outils sont à votre disposition. L'un d'entre eux est Miro. Découvrons dès maintenant comment il fonctionne.

B. Les critères d'acceptation d'une user story

Ceci permet notamment de mieux cerner les attentes du client, et cela peut également faciliter l'étape de validation du projet. Néanmoins, il y a certaines conditions à remplir afin qu'une user story soit considérée comme étant terminée et efficace.

L'une des conditions importantes des user stories est que celles-ci doivent être testables. C'est-à-dire qu'elles doivent pouvoir être validées ou invalidées lors de la phase de test. Pour être pertinentes, elles doivent être claires, spécifiques, mesurables et réalisables. Pour vous aider, vous pouvez vous inspirer du moyen mnémotechnique **INVEST** inventé par Bill Wake, consultant en développement logiciel et coach agile.

Selon lui, pour être considérées comme efficaces, les user stories doivent être :

I : Indépendantes les unes des autres pour être réalisées.

N : Négociables et donc suffisamment flexibles pour être ajustées au fil des retours.

V : avoir de la Valeur et une utilité concrète pour l'utilisateur final.

E : Estimables. C'est-à-dire qu'elles doivent être suffisamment précises pour pouvoir être estimées et chiffrées en matière de temps et de ressources nécessaires pour les réaliser.

S : *Small* (signifie « petit » en anglais), c'est-à-dire qu'elles doivent être suffisamment concises.

T : Testables. Les user stories doivent être testables pour permettre de valider leur réalisation et leur conformité aux besoins exprimés par le client.

Ces critères d'acceptation peuvent également inclure des exigences fonctionnelles telles que l'accessibilité, la sécurité ou encore la compatibilité de la fonctionnalité. Elles doivent aussi être modifiables et évolutives en fonction des retours des utilisateurs et des modifications apportées pendant le développement. Il est très important de les réviser régulièrement afin de s'assurer qu'elles restent pertinentes et en rapport avec les besoins de l'utilisateur. Ces critères d'acceptation sont une partie essentielle de la user story, car ils permettent de valider leur efficacité en situation réelle.

Complément

Une autre méthode d'acceptation, qui cette fois-ci permet également de réaliser des tests automatisés, est d'utiliser le **langage Gherkin**.

Il s'agit de suivre la structure suivante :

- Étant donné que...
- Quand...
- Alors...

Pour illustrer le langage Gherkin, reprenons l'exemple précédent. Pour rappel, la user story était la suivante : « *En tant qu'utilisateur, je veux pouvoir m'inscrire sur le site web pour pouvoir accéder aux fonctionnalités réservées aux membres, car cela me permettra d'interagir sur le forum* ».

Voici à quoi pourrait ressembler un test d'acceptation en langage Gherkin pour cette user story :

« Étant donné que je me suis inscrit sur le site web depuis la page d'accueil, quand je renseigne mes identifiants de connexion (mes nom d'utilisateur et mot de passe), et qu'ils sont corrects, alors j'ai accès au contenu réservé aux membres et je peux interagir sur le forum. »

C. Comment intégrer les user stories aux spécifications techniques ?

Comme nous l'avons vu tout au long du cours, les user stories sont extrêmement utiles en programmation. Mais pour en tirer un maximum de bénéfices, il est conseillé de les intégrer aux spécifications techniques du projet.

Définition Spécifications techniques

Une spécification technique est un document qui comprend l'ensemble des détails liés à la façon dont une fonctionnalité doit être implémentée. Elle comprend souvent des informations sur les langages de programmation qui vont être utilisés dans le cadre du projet, les bibliothèques et frameworks à utiliser par exemple. Ces spécifications techniques vont aider les développeurs à comprendre comment créer et implémenter chaque fonctionnalité.

Concernant les user stories, il sera important de les séparer clairement des spécifications techniques. Par exemple, si les spécifications techniques sont comprises dans un tableau, il est possible de créer une section distincte qui comprendra les spécifications techniques et une section à part destinée aux user stories.

Il est important que celles-ci respectent les critères énoncés dans les parties précédentes, à savoir : être écrites en langage métier donc dans un langage naturel ; vérifier qu'elles soient compréhensibles par tous les membres de l'équipe projet, y compris les non-techniques. Mais un point central des user stories est ici encore plus crucial. Il s'agit de leur **longueur**. Il est très important qu'elles soient concises, car si les user stories sont trop longues dans un tableau, par exemple, cela pourrait vite devenir ingérable et illisible.

Il est également important de s'assurer que les **spécifications techniques** sont clairement liées à la user story correspondante en utilisant la nomenclature de votre choix. Cela permet notamment à l'équipe de développeurs de prendre connaissance des exigences fonctionnelles (donc *via* la user story) et techniques en même temps (les caractéristiques purement techniques comme le langage de programmation de la fonctionnalité, par exemple). Ceci facilitera grandement le développement de la user story correspondante.

Voyons dorénavant une des méthodes d'organisation les plus courantes.

Cette méthode consiste à inclure les spécifications techniques dans la description de la user story. Cette approche peut être utilisée pour les personnes qui préfèrent garder toutes les informations au même endroit. Ainsi, elles peuvent prendre connaissance de l'ensemble des exigences en un coup d'œil. Par exemple, pour une user story qui décrit la fonctionnalité « *ajouter un article au panier* », les spécifications techniques pourraient inclure des informations sur la manière dont les données seront stockées, les fonctionnalités de validation ou encore, de sécurité.

Dans le cas présent, les spécifications techniques ainsi que la user story sont comprises dans la même section. Ainsi, les spécifications techniques sont incluses dans la description de la user story elle-même. Cette approche permet de lier de manière claire les spécifications techniques à la user story correspondante. Ceci facilite la compréhension et la communication entre les différents membres de l'équipe projet.

Cette méthode n'est qu'un exemple parmi tant d'autres. Quelle que soit la méthode choisie, il est simplement important que le lien entre la user story et ses spécifications techniques soit clairement identifiable par l'ensemble de l'équipe projet.

Méthode

Pour faciliter la création de vos user stories, il est possible de créer des « personas ». Dans la vidéo suivante, nous allons découvrir ce concept et créer votre premier persona grâce à l'outil gratuit disponible à cette adresse : <http://www.hubspot.fr/make-my-persona>.

D. Exercice : Quiz

[solution n°2 p.14]

Question 1

Quels sont les critères d'acceptation d'une user story selon le moyen mnémotechnique INVEST ?

- ☐ I pour indépendantes
N pour novatrices
V pour valeur ajoutée
E pour économiques
S pour sécurisées
T pour techniques
- ☐ I pour indépendantes
N pour négociables
V pour valeur ajoutée
E pour estimable
S pour small
T pour testable
- ☐ I pour inventives
N pour négociables
V pour valeur ajoutée
E pour estimable
S pour small
T pour testable

Question 2

Pourquoi est-il important que les user stories soient testables ?

- ☐ Il est important qu'elles soient testables car un des avantages principaux d'une user story est sa flexibilité et sa possibilité d'être mise à jour en fonction des différents retours collectés. Tester ses users stories, c'est récolter des avis et vérifier leur faisabilité réelle
- ☐ Il est important qu'elles soient testables afin de respecter le *product backlog* et ses délais
- ☐ Il est important qu'elles soient testables afin de ralentir le développement d'une fonctionnalité

Question 3

Qu'est-ce que le langage Gherkin et comment peut-il aider à la validation des user stories ?

- ☐ Le langage Gherkin permet de valider la user story selon la règle du qui, quoi et pourquoi
- ☐ Le langage Gherkin est une méthode de validation basée sur 3 composantes : étant donné, quand, alors
- ☐ Aucune de ces réponses

Question 4

Quelle est l'approche la plus commune pour intégrer les spécifications techniques dans les user stories ?

- ☐ Une approche qui consiste à réaliser 2 tableaux, l'un comprenant les spécifications techniques et le second les user stories, ceux-ci devront expressément être dans des feuilles de tableur distinctes
- ☐ L'approche qui consiste à inclure les 2 tableaux dans la même feuille de tableur
- ☐ Une approche qui consiste à inclure les spécifications techniques dans le titre de la user story
- ☐ Une approche qui consiste à créer une section à part, mais en la référençant tout de même de manière claire à la bonne spécification technique

Question 5

Comment la règle du qui, quoi, pourquoi peut-elle aider à écrire des user stories efficaces ?

- ☐ Elle peut aider à créer des user stories efficaces, car elle fournit une structure qui a fait ses preuves et comprend les informations essentielles à la bonne définition du besoin de l'utilisateur final
- ☐ Elle peut aider à créer des user stories efficaces, car elle permet de clarifier les rôles et les responsabilités de chaque membre de l'équipe projet
- ☐ Elle peut aider à créer des user stories efficaces, car elle permet de garantir que l'ensemble des parties prenantes du projet ont une compréhension commune de ce qui est nécessaire à la réussite du projet

IV. L'essentiel

Pour conclure, les user stories peuvent réellement être des facilitateurs de projets informatiques. Que vous travailliez en équipe ou non, grâce à leur méthodologie, vous êtes désormais plus efficace dans la création de vos projets informatiques. Comme nous l'avons vu tout au long du cours, en plus de l'aspect collaboratif, réaliser des user stories a de nombreux avantages. Elles permettent :

- De mieux comprendre les besoins des utilisateurs,
- De mieux communiquer leurs besoins à une équipe,
- D'obtenir des retours rapides et fréquents des utilisateurs tout au long du processus de développement,
- De réduire les coûts de développement et d'accélérer son processus de développement,
- Et enfin, d'avoir une méthode de développement agile, car en créant un MVP et en récoltant des avis des utilisateurs, on peut rapidement rectifier le tir et ajouter / modifier des fonctionnalités qui correspondent davantage aux besoins de l'utilisateur final.

V. Auto-évaluation

A. Exercice

Vous êtes développeur web travaillant au sein d'une SSII. Votre mission est de créer 1 user story liée au besoin suivant : le client souhaite que les utilisateurs puissent s'inscrire puis acheter des cours en ligne.

Question 1

[solution n°3 p.15]

Rédigez une user story en utilisant la règle du « *qui, quoi, pourquoi* ».

Question 2

[solution n°4 p.16]

Comment pouvez-vous vérifier que cette user story est testable ?

B. Test

Exercice 1 : Quiz

[solution n°5 p.16]

Question 1

Dans la règle du « *qui, quoi, pourquoi* », que désigne le « *qui* » ?

- ☐ Le client
- ☐ L'utilisateur du produit final
- ☐ L'équipe en charge du développement

Question 2

Quelles sont les caractéristiques principales d'une user story ?

- ☐ Elle doit être technique et compréhensible par l'ensemble de l'équipe chargée du projet
- ☐ Elle doit être négociable, claire et concise
- ☐ Elle doit être claire, concise et compréhensible par l'ensemble de l'équipe chargée du projet

Question 3

Qu'est-ce qu'une spécification technique ?

- ☐ Un document qui donne des indications sur le design d'une fonctionnalité
- ☐ Un fichier comprenant l'ensemble des spécifications techniques liées au projet ou à une fonctionnalité en particulier
- ☐ Un fichier qui regroupe l'ensemble de l'équipe chargée du projet

Question 4

Choisissez la user story la plus efficace :

- ☐ En tant que personne, je peux acheter un produit en ligne
- ☐ En tant qu'utilisateur, depuis la page produit je peux accéder au bon de commande pour acheter le produit et remplir mes coordonnées de livraison
- ☐ En tant qu'utilisateur, je peux ajouter un produit au panier pour l'acheter depuis la page Bon de commande

Question 5

Choisissez la user story testée en langage Gherkin la plus efficace :

- ☐ Étant donné que je suis un utilisateur, quand j'ajoute un produit à mon panier et si celui-ci a un stock de marchandises supérieur à 0, alors je peux acheter le produit depuis le bon de commande
- ☐ Étant donné que je suis un utilisateur, quand j'achète un produit alors je peux être livré
- ☐ Étant donné que je suis un visiteur, quand je navigue sur le site web, je peux acheter un produit


Solutions des exercices

Exercice p. 5 Solution n°1**Question 1**

MVP signifie *Minimum Valuable Product*. C'est le fait de créer un produit digital, pas entièrement terminé mais qui est pleinement fonctionnel.

☒ Vrai

☐ Faux


 Un MVP est un produit minimal viable qui est un avant-goût du produit digital informatique final. Il comporte les fonctionnalités essentielles du produit.

Question 2

Les user stories permettent de retranscrire les besoins d'un utilisateur en langage naturel plutôt qu'en langage technique.

☒ Vrai

☐ Faux

 Les user stories permettent de décrire le besoin d'un utilisateur en langage métier. Cela est crucial dans le cadre de la collaboration sur un projet, mais également pour les équipes de développement.


Question 3

Qu'est-ce qu'une user story ?

☐ C'est le storytelling d'un utilisateur

☒ C'est une fonctionnalité décrite depuis le point de vue de l'utilisateur

☐ Aucune de ces réponses

 Une user story est une fonctionnalité d'un produit digital décrite depuis le point de vue de l'utilisateur final. Elle est formulée en langage naturel et décrit de manière simple le besoin d'un utilisateur. Cela servira par la suite de guide aux équipes chargées du développement.


Question 4

Pourquoi les user stories font-elles partie des méthodes de développement agiles ?

☒ Elles respectent un des principes de base de la méthodologie agile qui consiste à développer un produit digital par itération ou sprint


☐ Elles permettent aux équipes de développement de se concentrer sur les tâches les plus complexes et les plus importantes pour l'utilisateur final, plutôt que de gaspiller leur temps sur des tâches moins critiques

☐ Elles permettent de décrire les fonctionnalités d'un produit de manière très technique, ce qui est essentiel dans la création d'un produit informatique

 Une user story fait partie des méthodes de développement dites agiles, de par leur respect du principe de développement en itération.

Question 5

C'est important de garder constamment en tête les besoins de l'utilisateur final lors de la conception d'un projet informatique.


- ☒ Vrai
- ☐ Faux
-  En gardant constamment en tête les besoins des utilisateurs, on peut s'assurer que le produit final sera fonctionnel et réellement utile à l'utilisateur final. Les user stories sont une méthode efficace pour rester focalisé sur les besoins des utilisateurs tout au long du processus de développement.

Exercice p. 9 Solution n°2

Question 1


Quels sont les critères d'acceptation d'une user story selon le moyen mnémotechnique INVEST ?

- ☐ I pour indépendantes
N pour novatrices
V pour valeur ajoutée
E pour économiques
S pour sécurisées
T pour techniques
- ☒ I pour indépendantes
N pour négociables
V pour valeur ajoutée
E pour estimable
S pour small
T pour testable
- ☐ I pour inventives
N pour négociables
V pour valeur ajoutée
E pour estimable
S pour small
T pour testable

 Ce moyen mnémotechnique permet de s'assurer de l'efficacité de la user story.


Question 2

Pourquoi est-il important que les user stories soient testables ?

- ☒ Il est important qu'elles soient testables car un des avantages principaux d'une user story est sa flexibilité et sa possibilité d'être mise à jour en fonction des différents retours collectés. Tester ses users stories, c'est récolter des avis et vérifier leur faisabilité réelle
- ☐ Il est important qu'elles soient testables afin de respecter le *product backlog* et ses délais
- ☐ Il est important qu'elles soient testables afin de ralentir le développement d'une fonctionnalité
-  Une des caractéristiques principales des user story est leur immense flexibilité. De plus, les différentes itérations permettent de récolter un certain nombre d'avis. Dans ce cas précis, il est donc important de mettre à jour les user stories. Enfin, la mise en place de tests permet de vérifier le bon fonctionnement des différentes user story.


Question 3

Qu'est-ce que le langage Gherkin et comment peut-il aider à la validation des user stories ?

- ☐ Le langage Gherkin permet de valider la user story selon la règle du qui, quoi et pourquoi
- ☒ Le langage Gherkin est une méthode de validation basée sur 3 composantes : étant donné, quand, alors
- ☐ Aucune de ces réponses
-  Le langage Gherkin est bien basé sur ces 3 composantes : étant donné, quand, alors.
Les réponses doivent être les plus précises possible afin de faciliter la phase de test.


Question 4

Quelle est l'approche la plus commune pour intégrer les spécifications techniques dans les user stories ?

- ☐ Une approche qui consiste à réaliser 2 tableaux, l'un comprenant les spécifications techniques et le second les user stories, ceux-ci devront expressément être dans des feuilles de tableur distinctes
- ☐ L'approche qui consiste à inclure les 2 tableaux dans la même feuille de tableur
- ☐ Une approche qui consiste à inclure les spécifications techniques dans le titre de la user story
- ☒ Une approche qui consiste à créer une section à part, mais en la référençant tout de même de manière claire à la bonne spécification technique
-  Cette approche est la plus courante. Quelle que soit l'approche choisie, il est également important de référencer très clairement les user stories aux spécifications techniques correspondantes.

Question 5

Comment la règle du qui, quoi, pourquoi peut-elle aider à écrire des user stories efficaces ?

- ☒ Elle peut aider à créer des user stories efficaces, car elle fournit une structure qui a fait ses preuves et comprend les informations essentielles à la bonne définition du besoin de l'utilisateur final
- ☐ Elle peut aider à créer des user stories efficaces, car elle permet de clarifier les rôles et les responsabilités de chaque membre de l'équipe projet
- ☐ Elle peut aider à créer des user stories efficaces, car elle permet de garantir que l'ensemble des parties prenantes du projet ont une compréhension commune de ce qui est nécessaire à la réussite du projet
-  Cette structure permet de définir les besoins de l'utilisateur final en se focalisant sur l'essentiel.

p. 10 Solution n°3

Afin de rédiger une user story qui correspondrait à la règle du « *qui, quoi, pourquoi* », vous pouvez commencer par isoler les éléments. Ensuite, vous pourriez faire appel à une des trames les plus couramment utilisées afin de structurer votre idée de manière précise, concise et simple à comprendre par l'ensemble de l'équipe en charge de la réalisation du projet, car ce sont des éléments de validation inhérents à la règle « *qui, quoi, pourquoi* ». Premièrement, le « *qui* » : ici, il s'agit d'un utilisateur. Ensuite, la question « *quoi* » : son besoin est de s'inscrire sur un site web. Enfin vient la question « *pourquoi* » : afin d'accéder aux cours dispensés sur le site web, mais accessibles uniquement aux personnes qui ont acheté le cours en question. En partant de ces divers éléments de réponses, la user story pourrait être la suivante : « *En tant qu'utilisateur, je souhaite pouvoir m'inscrire sur le site web pour accéder aux cours en ligne* ».


p. 10 Solution n°4

Pour vérifier que la user story « *En tant qu'utilisateur, je souhaite pouvoir m'inscrire sur le site web pour accéder aux cours en ligne* » soit testable, vous pouvez réaliser des tests inspirés du langage Gherkin. Ainsi, vous pourriez par exemple réaliser les tests suivants : « *Étant donné que l'utilisateur est sur la page d'inscription et que tous les champs requis sont correctement remplis, quand il clique sur le bouton d'inscription, alors un message de confirmation apparaît et le compte de l'utilisateur est créé avec succès* », « *Étant donné que l'utilisateur est sur la page de connexion et qu'il a un compte valide, quand il entre ses identifiants et qu'ils sont corrects, alors il est redirigé vers la page d'accueil et peut accéder à son compte* », ou encore « *Étant donné que l'utilisateur a acheté un cours en ligne et qu'il est connecté à son compte, quand il accède à la page des cours en ligne, alors il peut voir et accéder aux cours qu'il a achetés* ». Grâce à cela, on peut tester la fonctionnalité dans sa globalité.

Exercice p. 10 Solution n°5


Question 1

Dans la règle du « *qui, quoi, pourquoi* », que désigne le « *qui* » ?

- ☐ Le client
- ☒ L'utilisateur du produit final
- ☐ L'équipe en charge du développement
-  Le « *qui* » désigne l'utilisateur du produit final, c'est-à-dire la personne qui a besoin de la fonctionnalité développée.


Question 2

Quelles sont les caractéristiques principales d'une user story ?

- ☐ Elle doit être technique et compréhensible par l'ensemble de l'équipe chargée du projet
- ☐ Elle doit être négociable, claire et concise
- ☒ Elle doit être claire, concise et compréhensible par l'ensemble de l'équipe chargée du projet
-  Ceci est important, car la raison d'être d'une user story est de faciliter la collaboration entre les différents membres de l'équipe projet.

Question 3

Qu'est-ce qu'une spécification technique ?

- ☐ Un document qui donne des indications sur le design d'une fonctionnalité
- ☒ Un fichier comprenant l'ensemble des spécifications techniques liées au projet ou à une fonctionnalité en particulier
- ☐ Un fichier qui regroupe l'ensemble de l'équipe chargée du projet
-  Ce fichier donne des détails sur la manière dont la fonctionnalité doit être implémentée. Par exemple, le langage de programmation ou encore, le framework à utiliser.

Question 4

Choisissez la user story la plus efficace :

- ☐ En tant que personne, je peux acheter un produit en ligne
- ☐ En tant qu'utilisateur, depuis la page produit je peux accéder au bon de commande pour acheter le produit et remplir mes coordonnées de livraison
- ☒ En tant qu'utilisateur, je peux ajouter un produit au panier pour l'acheter depuis la page Bon de commande
- ☐ Cette user story est la plus efficace car elle est courte, compréhensible et simple.

Question 5

Choisissez la user story testée en langage Gherkin la plus efficace :

- ☒ Étant donné que je suis un utilisateur, quand j'ajoute un produit à mon panier et si celui-ci a un stock de marchandises supérieur à 0, alors je peux acheter le produit depuis le bon de commande
- ☐ Étant donné que je suis un utilisateur, quand j'achète un produit alors je peux être livré
- ☐ Étant donné que je suis un visiteur, quand je navigue sur le site web, je peux acheter un produit
- ☐ Cette user story en langage Gherkin est la plus efficace car elle est précise et compréhensible. Elle respecte également la structure étant donné/quand/alors.