

User Stories

Ce document décrit la manière d'utiliser des User Stories dans le cadre de notre projet de développement P_DEV.

Une User Story est une description simple d'un besoin, exprimée par un utilisateur, pour déterminer les fonctionnalités à développer. Elle vise une fonctionnalité précise et décrit comment l'utilisateur interagira avec le système une fois celui-ci terminé

Une User Story possède un titre (court), qui permet de la distinguer des autres sans risque de se tromper. Elle s'articule ensuite en quatre parties: **Description**, **Tests d'acceptance**, **Tâches** et **DoD**.

Description

La description énonce le besoin lui-même, mais aussi **qui** a ce besoin et avec quelle **intention**.

On utilise généralement la structure suivante pour formuler la description:

En tant que ...
Je veux ...
Pour ...

Ne soyons pas dogmatiques, cette structure n'a pas besoin d'être conservée à la lettre ! Elle doit servir avant tout de rappel à ce qui doit figurer dans la story.

Exemple 'à la lettre' :

En tant qu'utilisateur,
je veux m'authentifier auprès du système
pour pouvoir accéder à mes données.

Variante 1, où on considère l'intention du login comme évidente:

En tant qu'utilisateur,
je veux m'authentifier auprès du système

Variante 2, où on n'utilise pas la structure "En tant que ..." :

Les utilisateurs peuvent s'authentifier sur le système

Tests d'acceptance

Les tests d'acceptance sont le résultat de l'analyse fonctionnelle. Ils décrivent de manière précise, vérifiable et complète le comportement attendu du système dans le cadre de la story.

Pour chaque test, on énonce:

1. Le contexte de départ
2. L'action effectuée par l'utilisateur
3. La réaction attendue du système

Pour simplifier l'écriture des tests, on peut faire référence à des maquettes

Voici quelques exemples de tests bien formulés

Sur la page d'accueil du site (maquette 1),
quand je clique le bouton 'Login',
le formulaire d'authentification s'ouvre (maquette 2)

Dans le formulaire d'authentification avec des champs vides (maquette 2)
quand le clique 'Ok'
le formulaire reste ouvert et des messages d'erreur sont affichés (maquette 3)

Dans le formulaire d'authentification avec des champs contenant des identifiants incorrects
quand le clique 'Ok'
le formulaire reste ouvert et des messages d'erreur sont affichés (maquette 4)

Dans le formulaire d'authentification avec des champs contenant des identifiants valables
quand le clique 'Ok'
Je retourne à l'accueil. Mon nom d'utilisateur et un bouton de déconnexion sont visibles (maquette 5)

--- ATTENTION ---

Il est de la plus haute importance de valider les tests d'acceptance avec le client avant d'aller plus loin!!!

Si vous ne le faites pas, vous courez le risque de faire un travail inutile.

Tâches

La liste des tâches est le résultat de l'analyse technique. Elle décrit le travail à effectuer pour arriver au point où on pourra exécuter chacun des tests d'acceptance sur le système.

Pour chaque tâche, estimez et notez le temps dont vous pensez avoir besoin pour sa réalisation.

Exemple de liste suite à l'analyse technique:

- Créer le formulaire d'authentification (15min)
- Script de modification de base de données pour ajouter le champ 'password' (15min)
- Traitement du formulaire bien rempli (10min)
- Traitement du formulaire mal rempli (10min)

Durant le développement, il est très probable que vous allez identifier des tâches auxquelles vous n'avez pas pensé durant l'analyse technique. J'appelle cela des "surprises". Ajoutez-les à la suite de la liste, avec un tag "surprise".

Lorsqu'une tâche est terminée, notez le temps qu'elle vous a effectivement pris. Si ce temps est très différent de votre estimation, ajoutez un tag:

- "facile" si le temps effectif est moins que la moitié du temps estimé
- "durdur" si le temps effectif est plus du double du temps estimé

Ainsi, votre liste de tâches à la fin de la story pourrait ressembler à ceci:

- Créer le formulaire d'authentification (15min/10min)
- Script de modification de base de données pour ajouter le champ 'password' (15min/20min)
- Traitement du formulaire bien rempli (10min/10min)
- Traitement du formulaire mal rempli (10min/30min) **durdur**
- Mise à jour du MLD dans la documentation (5min/5min) **surprise**

Moins vous aurez de **surprise**, **durdur** et **facile** à la fin de vos stories, meilleur vous serez !

DoD

La DoD (Definition of Done) est une liste de points qui doivent être vérifiés pour que l'on puisse déclarer avoir terminé le travail sur la story.

Exemple:

- Tests d'acceptance validés en présence du client
- Documentation technique mise à jour
- Commits Git publiés (push)
- Revue de code effectuée avec un collègue