

# R3.03 Analyse - Activités

Franck Silvestre

Version 1.4, 13 octobre 2022

# Table of Contents

1. Activité I - Rédiger des User Story .....	1
1.1. Introduction .....	1
1.1.1. Objectifs .....	1
1.1.2. Modalités .....	1
1.1.3. Le cas du site de "L'Alliance pour l'Ostéopathie Autrement". .....	1
1.2. Premiers pas dans le rôle de Product Owner .....	1
1.2.1. Lecture active du cahier des charges .....	1
1.2.2. Questions/Réponses .....	1
1.3. Rédaction des Users Stories .....	1
1.3.1. Utilisation du board Github Projects .....	2
1.3.2. Rédaction d'US pour le site ALOSA .....	2
2. Activité 2 - Détailler des US .....	2
2.1. Introduction .....	2
2.1.1. Objectifs .....	2
2.1.2. Modalités .....	2
2.2. Études d'exemples de tests d'acceptation .....	2
2.3. Rédaction des tests d'acceptations de vos US .....	3
2.3.1. Rédaction des tests d'acceptation d'une US .....	3
2.3.2. Rédaction des tests d'acceptation de l'ensemble des US .....	3
3. Activité 3 - Définition de Fini (DdF) .....	3
3.1. Introduction .....	3
3.1.1. Objectifs .....	3
3.1.2. Modalités .....	3
3.2. Rédaction d'une DdF .....	4
4. Activité 4 - Diagramme de classes UML .....	4
4.1. Introduction .....	4
4.1.1. Objectifs .....	4
4.1.2. Modalités .....	4
4.2. Dictionnaire de données .....	4
4.3. Diagramme de classes UML .....	4

# 1. Activité I - Rédiger des User Story

## 1.1. Introduction

### 1.1.1. Objectifs

- Se confronter aux problèmes que soulève la traduction d'un cahier des charges d'un client en quelque chose d'exploitable par une équipe de développement.
- Rédiger des **User Stories**.

### 1.1.2. Modalités

Alternance de travail en binôme et individuel sur étude de cas.

### 1.1.3. Le cas du site de "L'Alliance pour l'Ostéopathie Autrement".

Vous êtes "consultant fonctionnel" au sein d'une agence web. On vient de vous confier un nouveau projet consistant à développer le site Web de "L'Alliance pour l'Ostéopathie Autrement" (ALOSA). Vous êtes donc le ou la consultant•e fonctionnel•le côté maîtrise d'œuvre sur ce projet. Le projet doit se dérouler en mode agile en utilisant la méthode Scrum. Dans le jargon Scrum, vous êtes donc le Product Owner. En tant que Product Owner, votre rôle est de fournir de toutes les manières possibles les informations permettant à l'équipe de développement de développer le site Internet ALOSA.

Le cahier des charges fourni par le client est disponible dans les ressources de l'activité.

## 1.2. Premiers pas dans le rôle de Product Owner

### 1.2.1. Lecture active du cahier des charges

1. Lisez individuellement le cahier des charges en relevant toutes les questions et remarques que celui suscite, toujours dans l'idée de pouvoir transmettre le maximum d'éléments permettant à l'équipe de développement de faire le travail.
2. Mettez en commun vos questions et remarques avec votre binôme.

### 1.2.2. Questions/Réponses

Activité élastique : Du CDC au développement...



Vous pouvez réfléchir en binôme, mais les réponses fournies doivent être individuelles.

## 1.3. Rédaction des Users Stories

### 1.3.1. Utilisation du board Github Projects

Pour cette activité, vous aurez besoin d'utiliser votre compte Github.

1. Inscrivez-vous au devoir Site\_Alosa via Github Classroom et RDV sur le projet créé. [Lien d'inscription Github Classroom](#)



Coordonnez-vous au sein du binôme pour vous inscrire dans la même **team** pour partager le même projet.

2. Suivez les instructions de votre enseignant pour ajouter des labels applicables aux issues que vous allez créer. Créez au moins les labels **Feature** et **US**.
3. Cliquez sur l'onglet Github Projects disponible dans le projet Github "Site\_Alosa".
4. En suivant les instructions de votre enseignant :
  - a. Créez un nouveau **project** et lié au repository Site\_Alosa.
  - b. Ajoutez la colonne "Product backlog" et placez là en première position.
  - c. Créez une **feature** de test et une **US** associée à cette **feature**.

### 1.3.2. Rédaction d'US pour le site ALOSA

1. Au sein de votre binôme, répartissez vous 1 partie du site ALOSA à spécifier sous forme d'US.
2. Directement dans le board Github Projects du projet :
  - Créez une **feature** correspondant à la partie que vous devez spécifier.
  - Rédigez 6 US attachées à cette **feature** (1 issue par US dans la colonne "backlog").
3. Relisez les US écrites par votre binôme en notant vos questions et vos remarques sur ces US que vous n'avez pas écrites.
4. Échangez avec votre binôme pour finaliser l'écriture des 12 US qui ont été rédigées.

## 2. Activité 2 - Détailler des US

### 2.1. Introduction

#### 2.1.1. Objectifs

- Rédiger les tests d'acceptations de **User Stories**.

#### 2.1.2. Modalités

Alternance de travail en binôme et individuel sur étude de cas.

### 2.2. Études d'exemples de tests d'acceptation

[Activité elastic : Exemples de tests d'acceptation...](#)

## 2.3. Rédaction des tests d'acceptations de vos US

Les tests d'acceptations seront ajoutés directement dans la description des US sur le board de votre projet.

### 2.3.1. Rédaction des tests d'acceptation d'une US

1. Think : Choisissez une des US que vous avez rédigée. Rédigez les tests d'acceptations pour cette User Story en distinguant bien le cas nominal des cas limites et des cas d'erreur.
2. Pair :
  - Étudiez et commentez, en utilisant les fonctionnalités du Board, les tests d'acceptation rédigés par votre binôme.
  - Échangez de vive voix avec votre binôme sur les commentaires postés sur vos US respectives.
3. Share : Questions/Réponses entre enseignant et étudiant•e•s.
4. Reprenez, si nécessaire, la rédaction de vos tests d'acceptation afin qu'ils prennent en compte l'ensemble des commentaires et échanges.

### 2.3.2. Rédaction des tests d'acceptation de l'ensemble des US

Pour chaque US que vous avez rédigée :

1. Rédigez les tests d'acceptation.
2. Étudiez et commentez, en utilisant les fonctionnalités du Board, les tests d'acceptations rédigés par votre binôme.
3. Échangez de vive voix avec votre binôme pour converger vers un consensus sur les modifications à apporter à l'ensemble des tests.
4. Prenez en compte les résultats de commentaires et des échanges pour finaliser la rédaction des tests d'acceptation.

## 3. Activité 3 - Définition de Fini (DdF)

### 3.1. Introduction

#### 3.1.1. Objectifs

Rédiger la première version d'une définition de fini pour un projet de développement d'application Web.

#### 3.1.2. Modalités

Alternance de travail en binôme et individuel sur projet Web de la SAÉ.

## 3.2. Rédaction d'une DdF



Travail individuel

1. À partir de la lecture du cahier des charges de l'application Web de votre SAÉ et du PMP de la SAÉ, rédigez une première version de DdF en prenant soin de distinguer la *check list* concernant les items de backlog de celle qui concerne l'incrément.



Vous pouvez créer une colonne "Définition de fini" sur le Board avec dans cette colonne une carte pour chaque *check list*.



Travail en binôme

2. Comparez vos checklists respectives et échangez sur les DdF obtenues.
3. Enrichissez vos *check lists*, si besoin, en tirant partie de vos échanges.

## 4. Activité 4 - Diagramme de classes UML

### 4.1. Introduction

#### 4.1.1. Objectifs

Passer des exigences à la modélisation.

#### 4.1.2. Modalités

Alternance de travail en binôme et individuel.

### 4.2. Dictionnaire de données



Travail individuel

1. À partir des user stories spécifiant le site ALOSA, constituez le dictionnaire de données.



Travail en binôme

2. Comparez et échangez sur vos dictionnaires de données respectifs.
3. Créez un fichier "DictionnaireDonnees.md" à la racine du projet sur Github et complétez ce document avec un dictionnaire de données unique résultant de vos échanges.

### 4.3. Diagramme de classes UML



Travail individuel

1. À partir du dictionnaire de données, constituez un diagramme de classes (sans les opérations).



Travail en binôme

2. Comparez et échangez sur votre diagramme de classes.
3. Déposez à la racine du projet sur Github un document au format PDF intitulé "DiagrammeClasses.pdf" contenant le diagramme de classes résultant de vos échanges.