

## Identification du module

<b>Numéro de module</b>	<b>426</b>
<b>Titre</b>	Développer un logiciel avec des méthodes agiles
<b>Compétence</b>	Appliquer des méthodes agiles pour le développement de logiciels dans les cycles de révision

---

### Objectifs opérationnels

1. Mettre en œuvre une fonctionnalité donnée, dans le cadre d'un projet logiciel, avec une méthode agile.
  2. Réaliser et tester par étapes les fonctionnalités à l'aide de pratiques agiles dans les cycles de révision prescrits, et présenter en résumé la version logicielle.
  3. Mettre en œuvre, de manière ciblée, des échantillons de développement existants et/ou des composants logiciels testés pour résoudre le problème.
  4. Refléter les résultats et le déroulement du travail lors d'un cycle de révision, en déduire les conclusions pour la suite du déroulement.
  5. Mettre à disposition les documents du projet et les codes sources du programme dans un système de gestion des versions.
  6. Formuler de manière compréhensive le code source programme selon les conventions.
- 

<b>Domaine de compétence</b>	Application Engineering
<b>Objet</b>	Version logicielle fonctionnelle, testée et documentée.
<b>Version du module</b>	1.0
<b>Créé le</b>	11.02.2021

## Connaissances opérationnelles nécessaires

<b>Numéro de module</b>	<b>426</b>
<b>Titre</b>	Développer un logiciel avec des méthodes agiles
<b>Compétence</b>	Appliquer des méthodes agiles pour le développement de logiciels dans les cycles de révision

### Objectifs opérationnels et connaissances opérationnelles nécessaires

1	1.1	Connaître une méthode agile pour le développement logiciel.
	1.2	Connaître le contenu et la structure d'un User-Story.
	1.3	Connaître des valeurs et des principes de méthodes agiles.
	1.4	Connaître les limites des méthodes linéaires et agiles.
2	2.1	Connaître le déroulement et les éléments d'un cycle des versions.
	2.2	Connaître le déroulement et le contenu d'une présentation des versions.
	2.3	Connaître l'importance des versions fonctionnelles de logiciels.
	2.4	Connaître les pratiques (Best Practices) du développement logiciel agile comme par ex. Design, Metapher, Refactoring, Testdrive Design, Continuous Integration, Pair Programming, Coding Standards.

	2.5	Connaître les outils appropriés au développement agile pour les UnitsTests, System Integration, System Build, test d'acceptance, etc.
3	3.1	Connaître l'importance et la structure des échantillons significatifs de développement.
	3.2	Connaître l'utilité de la réutilisation des composants logiciels existants
	3.3	Connaître un procédé pour l'évaluation de composants logiciels ainsi que pour résoudre des problèmes donnés.
4	4.1	Connaître un procédé pour refléter les résultats d'un cycle des versions.
	4.2	Connaître des possibilités pour refléter son propre déroulement du travail dans un cycle des versions.
	4.3	Connaître des possibilités pour optimiser la suite du procédé sur la base de la réflexion (SCRUM, ceremony, product backlog issues).
5	5.1	Connaître les avantages et utilités d'un système de gestion des versions utilisé communément.
	5.2	Connaître les fonctionnalités et l'utilisation d'un système de gestion des versions, ainsi que les possibilités d'y intégrer divers objets.
	5.3	Connaître les différences entre un système de gestion des versions avec accès exclusif et accès parallèle.
6	6.1	Connaître les concepts, les règles et procédures d'un Clean-Code.
	6.2	Connaître l'importance du Refactoring pour l'amélioration intuitive de la lisibilité d'un code source programme.

Version du module 1.0  
Crée le 11.02.2021

Niveau d'exigences	Niveau C	Description	Verbes typiques des activités
<b>Savoir</b>	<b>C1</b>	Restituer des informations et les retrouver dans des situations similaires.	Désigner, noter, énumérer, nommer, restituer.
<b>Comprendre</b>	<b>C2</b>	Non seulement restituer des informations, mais les comprendre.	Décrire, expliquer, commenter, reformuler, démontrer, caractériser
<b>Appliquer</b>	<b>C3</b>	Appliquer des informations circonstanciées dans différentes situations.	Appliquer, établir, exécuter, calculer, utiliser, traduire, transposer
<b>Analyser</b>	<b>C4</b>	Décomposer une situation en ses divers éléments, établir les relations entre ces éléments et en identifier les tenants et les aboutissants.	Interpréter, analyser, résoudre, différencier, décomposer, identifier, examiner, comparer, diviser, contrôler, mesurer
<b>Synthétiser</b>	<b>C5</b>	Combiner les éléments d'une situation pour former un tout, ou concevoir la solution d'un problème.	Justifier, noter, structurer, mettre en place, élaborer, projeter, développer, concevoir, combiner, construire, optimiser, planifier, rédiger, établir, élaborer
<b>Evaluer</b>	<b>C6</b>	Evaluer des informations et des situations déterminées selon des critères prédéfinis ou selon ses propres critères.	Apprécier, évaluer, qualifier

### Niveaux d'exigences (taxonomie)

L'indication du niveau d'exigences des objectifs évaluateurs en reflète le degré de difficulté. On distingue six niveaux de compétences (C1 à C6) Le tableau ci-dessous les présente en détail.