

**Libellé : Développement et Qualité des Logiciels**

Libelle	ECTS	Semestre	CM	TD	Total	mois de stage	heures de projet	Compétences attendues (à renseigner pour chaque UE et module)	Bloc de compétences (indiquer à quel bloc ces compétences sont rattachées)
<b>Communication et environnement professionnel</b>	<b>6</b>	<b>5</b>	<b>21</b>	<b>96</b>	<b>117</b>			<b>être plus qu'un développeur</b>	communiquer et comprendre son environnement professionnel (être plus qu'un développeur)
Préparation à l'insertion professionnelle				15				écrire un CV et une lettre de motivation	
Tutoriel sur un produit logiciel				9				vulgariser une information technique et expliquer l'utilisation d'un logiciel	
Web documentaire				15				analyser une situation de communication et assembler différents types de médias	
Anglais : communication verbale				12				parler d'un projet avec un interlocuteur étranger	
Anglais : communication écrite basique				9				écrire des mails et programmer en Anglais	
Anglais : compréhension d'articles techniques				9				lire des docs et des ouvrages techniques en Anglais	
Anglais : compréhension d'exposés techniques				9				comprendre des conférences techniques en Anglais	
Droit de l'informatique : notions générales de droit du contrat appliquées à l'informatique			10,5					connaître la législation concernant les données personnelles et les droits d'auteur	
Droit social : notions générales du droit du travail et problématiques liées à l'informatique			10,5					savoir lire et comprendre un contrat de travail	
Outils d'un gestionnaire de projets				18				gérer un projet en terme de ressources, compétences et activités critiques	
<b>Modélisation, Conception, Programmation</b>	<b>6</b>	<b>5</b>	<b>33</b>	<b>87</b>	<b>120</b>			<b>être un programmeur utilisant les techniques d'aujourd'hui</b>	développer en utilisant les techniques actuelles (être un développeur d'aujourd'hui)
Paradigmes de l'orienté Objet			12					comprendre les principes du développement logiciel à objets	
Conception Orientée Objet (UML)			21					utiliser un langage de modélisation pour concevoir et communiquer sur des architectures logicielles	
Langage Java				33				programmer dans le langage à objets le plus répandu	
Architectures Client Serveur				12				comprendre les architectures logicielles d'aujourd'hui	
Programmation Web				18				programmer dans les langages PHP et Javascript	
Gestion des cas exceptionnels				12				gérer proprement les cas d'erreurs	
Interopérabilité et paramétrage (XML)				12				paramétrer des logiciels, interagir avec des outils dédiés au développement logiciel	
<b>Méthodes agiles</b>	<b>6</b>	<b>5</b>	<b>42</b>	<b>60</b>	<b>102</b>			<b>aller sur la voie de l'excellence technique</b>	
Scrum et eXtreme Programming			21					maîtriser les principales méthodes agiles de développement logiciel	
Test Driven Development				12				maîtriser le mantra du bon programmeur	
Clean Code et Refactoring				9				augmenter la qualité intrinsèque de son code	

Libelle	ECTS	Semestre	CM	TD	Total	mois de stage	heures de projet	Compétences attendues (à renseigner pour chaque UE et module)	Bloc de compétences (indiquer à quel bloc ces compétences sont rattachées)
Bonnes pratiques réutilisables (Design Patterns)				9				connaître les référentiels des bonnes pratiques et savoir les utiliser à bon escient	concevoir du code de qualité (être un bon développeur)
Exigences agiles et user stories				21				capter les besoins clients sous la forme d'histoires utilisateurs	
Usine Logicielle			12					assurer la mise en production en continu d'un logiciel	
Amélioration continue d'un processus de développement industriel			9					connaître une démarche d'amélioration continue d'un processus de développement	
Qualité d'un code				9				utiliser des outils fournissant des métriques sur le code	
<b>Tests</b>	<b>6</b>	<b>5</b>	<b>24</b>	<b>51</b>	<b>75</b>			<b>avoir confiance et inspirer confiance</b>	
Théorie des tests				9				comprendre le monde des tests et des testeurs	
Tests unitaires (Framework JUnit)				6				réaliser des tests unitaires sur du code applicatif	
Frameworks de tests			12					connaître d'autres frameworks pour automatiser les tests d'acceptation et d'Interfaces Hommes Machines	
Paradigmes du testeur Agile			12					avoir le niveau d'un testeur agile certifié	
Interfaces Homme Machine et tests d'utilisabilité				15				comprendre ce qu'est l'utilisabilité d'un produit logiciel	
Complexité des algorithmes et structures de données				21				rendre un code plus performant	
<b>Nouveaux paradigmes de programmation</b>	<b>6</b>	<b>5</b>	<b>36</b>	<b>0</b>	<b>36</b>			<b>être un programmeur utilisant les techniques de demain</b>	
Génération de code à partir de modèles			21					modéliser un langage métier afin de générer automatiquement du code	
Langages spécifiques à un domaine (DSL)			15					implémenter un langage de programmation métier	
<b>Projet</b>	<b>10</b>	<b>6</b>					<b>150</b>	Développer collaborativement en mode "full Agile" un produit logiciel	s'immerger dans un contexte professionnel (être un développeur responsable)
<b>Stage</b>	<b>20</b>	<b>6</b>				<b>4</b>		Libérer ses compétences dans un cadre professionnel	