

Plan de cours

420-W61-SF

Projet synthèse – Développement

0-10-3

Hiver 2022

Équipes:	01	02, 03, 04	05, 06	07, 08
Professeurs :	Benjamin Lemelin	François Paradis	Yannick Mazières	Guillaume Simard
Bureaux :	C-242.00	C-242.00	C-112.05	C-112.03
Téléphones :	418-659-6600 #5806	418-659-6600 #5808	418-659-6600 #5815	418-659-6600 #5803
Courriels :	blemelin@csfoy.ca	fparadis@csfoy.ca	ymazieres@csfoy.ca	gsimard@csfoy.ca

Préambule

Le Cégep de Sainte-Foy souhaite être un milieu inclusif pour les personnes qui y étudient, y travaillent et y évoluent.

Chacun(e) devrait également se sentir confortable de s'exprimer et de demander qu'on s'adresse à elle ou à lui dans le genre de son choix ou de façon neutre si c'est là son besoin.

Présentation

Thématique

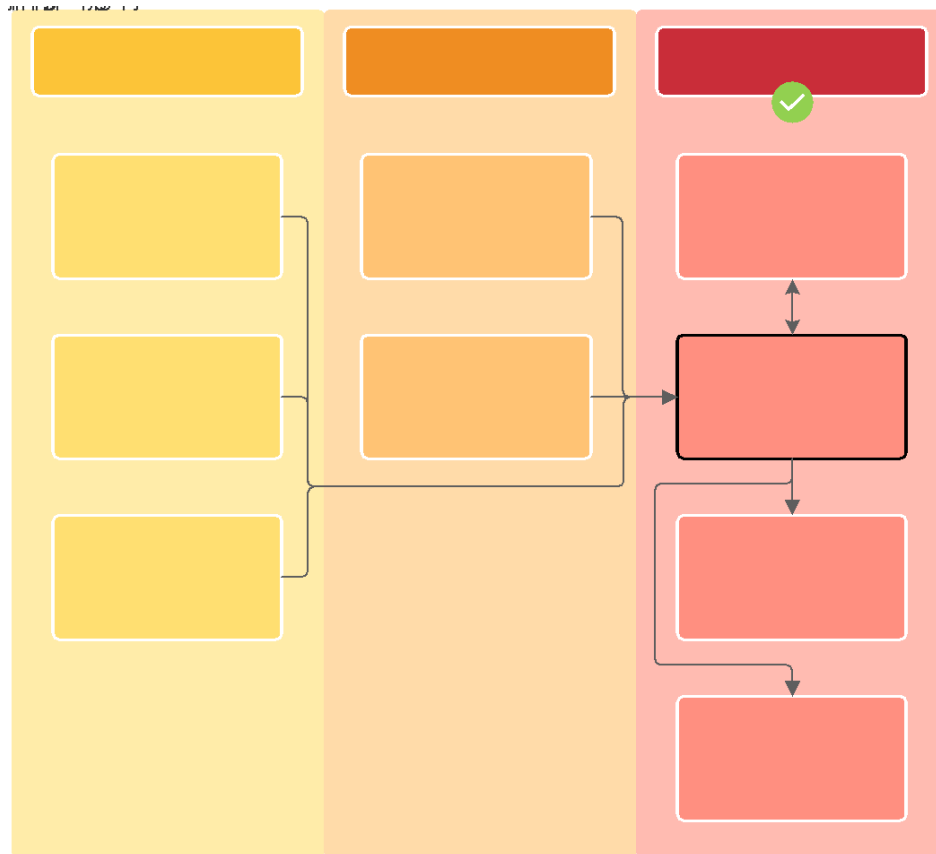
Ce cours permet à l'étudiante ou l'étudiant de contribuer, au sein d'une équipe Agile, au développement d'une application d'envergure pour répondre aux besoins d'un client.

Elle ou il installe et configure un environnement technologique supportant un processus d'intégration continue automatisé, puis effectue la conception et le développement des fonctionnalités en appliquant les méthodes et pratiques de développement Agile.

Les équipes d'étudiantes et d'étudiants travaillent sur leurs projets à temps plein pendant 7 semaines.

Préalables

Ce cours a comme préalables absolus les cours *Développement d'applications Web* (420-W40-SF), *Développement de services Web* (420-W41-SF), *Infrastructure technologique et virtualisation* (420-W4A-SF), *Développement mobile et objets connectés* (420-W50-SF), *Programmation de jeux vidéo II* (420-W51-SF). Il a aussi comme préalable concomitant le cours *Projet synthèse – Équipe agile* (420-W60-SF). Enfin, il est préalable concomitant aux cours *Stage – Interactions en milieu de travail* (420-W70-SF) et *Stage – Développement* (420-W71-SF).



Calendrier

Ce calendrier est donné à titre indicatif et peut être modifié en fonction des circonstances.

Semaines	Sprint	Contenus	Livraisons
1 24 janvier	0	Installation et configuration de l'environnement de travail Rencontre avec le client et appropriation du catalogue	
2 31 janvier	1	Création du squelette de l'application Réalisation d'un récit utilisateur	Activité 4 – Environnement (Sommatif)
3 7 février	2	Identification, estimation et priorisation des récits utilisateur Début Itération <ul style="list-style-type: none"> Planification itération Développement Rencontres quotidiennes (<i>Scrum Meeting</i>) 	
4 14 février	2	Fin Itération <ul style="list-style-type: none"> Développement Rencontres quotidiennes (<i>Scrum Meeting</i>) Revue de code Présentation itération au client Rétrospective itération 	Activité 2 – Travail d'équipe (Sommatif) Activité 3 – Exécution des tâches (Formatif) Activité 5 – Conception (Formatif)

			Activité 6 – Développement (Formatif)
5 21 février	3	Identification, estimation et priorisation des récits utilisateur Début itération <ul style="list-style-type: none"> • Planification itération • Développement • Rencontres quotidiennes (<i>Scrum Meeting</i>) 	
6 28 février	3	Fin Itération <ul style="list-style-type: none"> • Développement • Rencontres quotidiennes (<i>Scrum Meeting</i>) • Revue de code • Présentation itération au client • Rétrospective itération 	Activité 5 – Conception (Sommatif) Activité 6 – Développement (Sommatif)
7 7 mars	4	Préparation aux présentations orales Livraison au client Présentation grand public (mardi le 8 mars 2022) Finalisation et fermeture du projet	Activité 1 – Cahier de réflexion (Sommatif) Activité 2 – Travail d'équipe (Sommatif) Activité 3 – Exécution des tâches (Sommatif)

Travaux

Chaque membre d'une équipe de travail doit apporter une contribution équitable à l'ensemble de l'épreuve synthèse de programme. Un facteur personnel pourra être attribué à chaque membre basé sur sa contribution et son professionnalisme. Tout membre ayant manqué à son devoir pourra aussi être convoqué à une rencontre d'évaluation.

Évaluations

Évaluation	Contexte de réalisation	Type	Pondération
Préproduction – Individuel <i>L'étudiant configure son environnement de travail local permettant de développer l'entièreté du projet.</i>	Individuel	Sommatif	5%
Préproduction – Équipe <i>L'équipe met en place les environnements destinés aux tests, aux démonstrations et aux déploiements dans une perspective d'intégration continue.</i>	En équipe	Sommatif	15%
Conception et analyse <i>L'équipe analyse les fonctionnalités à produire pour constituer un catalogue de produit qui sera révisé, modifié et bonifié à chaque début d'itération.</i>	En équipe	Sommatif	20%
Développement <i>L'équipe développe les fonctionnalités du catalogue de produit et effectue des livraisons à la fin de chaque itération.</i>	En équipe	Sommatif	60%

Évaluations formatives

Afin de donner de la rétroaction, deux revues formatives sont prévues :

- Une revue du catalogue de produit au courant de la semaine 4.
- Une revue de code au courant de la semaine 4 ou 5.

Les revues formatives sont obligatoires, et chaque équipe est responsable de planifier les rencontres avec son professeur superviseur. Le format et la durée des rencontres est laissé à la discrétion du professeur superviseur.

Évaluation finale à caractère synthèse

Étant donné la nature synthèse de ce cours, l'ensemble des activités d'évaluation constituent une évaluation sommative à caractère synthèse. L'évaluation finale à caractère synthèse totalise donc 100% de la note finale.

Conditions de réussite

Pour obtenir la note de passage, une moyenne supérieure ou égale à 60 % doit être obtenue pour l'ensemble des évaluations.

Médiagraphie

Aucun livre obligatoire. Des références pratiques seront données sur des sites pertinents de façon *ad hoc* en cours de session. Les notes de cours, énoncés et remises de travaux ainsi qu'autres documents sont déposés sur un support électronique.

Compétences

Articulation

Ce cours est le quatrième cours de cinq cours porteurs de la compétence 00SU, développée aussi en session 1 dans le cours Programmation Web et jeux vidéo, en session 2 dans le cours Programmation Web dynamique, en session 4 dans le cours Développement d'applications Web, et finalement en session 6 dans le cours Stage - Développement.

Ce cours est le dernier cours de deux cours porteurs de la compétence 00SY, développée aussi en session 5 dans le cours Conception d'applications en contexte Agile.

Progression des apprentissages

Compétence 00SU

Le cours aborde les éléments de la compétence en lien avec l'installation de l'infrastructure technologique nécessaire au développement d'une application, la conception des fonctionnalités, la qualité du code et de l'application. Les habiletés et contenus de ces éléments de compétence sont essentiellement les mêmes que ceux des compétences de développement d'applications mobiles (00SS), pour des objets connectés (00SX) ou de jeux vidéo (00SW).

Les projets réalisés par les équipes d'étudiants peuvent donc être diversifiés, les éléments de la compétence s'appliquant à des projets de différents types (Web, mobile, jeux vidéo, bureau, avec des objets connectés, etc.) et de différentes natures.

Compétence 00SY

Tous les éléments de la compétence ont été couverts dans le cours Conception d'applications en contexte Agile en session 5. Dans ce cours, l'étudiant met en œuvre ces apprentissages et participe à la constitution du catalogue des fonctionnalités à réaliser, en établissant les critères d'acceptation et la complexité des récits utilisateurs.

Au terme du cours, l'étudiante ou l'étudiant est en mesure de :

- Installer et exploiter un environnement de développement supportant un processus d'intégration continue.
- Élaborer la conception détaillée de récits utilisateurs du catalogue.
- Réaliser des récits utilisateurs en respectant une architecture de qualité et les bonnes pratiques de programmation.
- Réaliser les tests unitaires et d'acceptation pour certifier de la qualité de l'application.
- Respecter les bonnes pratiques de partage des sources et de gestion des versions.

Le travail au sein d'une équipe est exigé dans ce cours de par la nature même d'un processus de développement Agile.

Voir tableau des compétences détaillées en annexe (Annexe – Compétences détaillées).

Pédagogie et apprentissage

Pédagogie

L'épreuve synthèse de programme s'effectue dans un contexte très proche de ce qui se fait réellement en entreprise. Il est réalisé en équipe, s'échelonne sur 7 semaines, compte 5 sprints (dont 3 de développement) et produit 3 livraisons. Chaque équipier doit contribuer de manière significative dans le but de livrer un produit final complet et de qualité.

Chaque équipe se verra assigner un professeur superviseur pour l'ensemble de l'épreuve synthèse. Ces professeurs agiront principalement comme observateurs et évaluateurs, l'autonomie et la débrouillardise étant de premier ordre. Le rôle secondaire du professeur superviseur dépendra de la nature du projet à réaliser, sans empiéter sur les compétences à évaluer : directeur de produit, guide technique, expert technologique, maître agile, etc.

Durant les sprints de développement, le professeur superviseur agira comme conseiller sur différents plans : méthodes agiles, conseils architecturaux, bonnes pratiques, etc. Ces conseils seront le fruit des diverses observations du professeur, des mêlées quotidiennes et des rétrospectives d'équipe. L'objectif est de fournir régulièrement de la rétroaction aux équipes.

Le professeur superviseur assistera à la majorité des mêlées quotidiennes et des rétrospectives de l'équipe, mais ne sera pas présent chaque fois dû à des contraintes horaires. Il ne sera pas non plus présent pendant toutes les périodes de travail. Bien qu'un professeur soit attribué à une équipe, il peut aussi venir en support à une autre équipe au besoin.

Les professeurs interviendront périodiquement pour introduire et commenter des concepts et notions nécessaires à la réalisation des projets. Il s'agira surtout d'inventions *ad hoc* en cours de session afin de former les équipes sur certaines compétences nécessaires à la réalisation de leurs projets individuels.

Comportement et attitude

Durant ce cours, chaque étudiant doit démontrer un comportement et une attitude professionnels jugés essentiels pour une intégration réussie dans sa profession. Ainsi, il doit :

- Être présent à tous les cours chaque semaine;
- Être assidu aux cours et ponctuel;
- Faire montre en tout temps d'un comportement respectueux de son entourage;
- Respecter les directives relatives à l'utilisation des ressources matérielles et logicielles;
- Apporter une contribution équitable dans le cas des travaux en équipe.

Évaluations des apprentissages

Tous les travaux et examens sont évalués selon les règles régies par la politique d'évaluation des apprentissages (P.E.A.) du Cégep de Sainte-Foy. En août 2020, des modifications temporaires ont été apportées à la Politique afin de tenir compte du contexte d'enseignement particulier entraîné par la crise sanitaire.

Sujet	Règle
Évaluation du français Article 6.1.8	[...] Dans toutes les activités d'évaluation sommative, chaque erreur de langue est soustraite du résultat de l'étudiant à raison de 0,5 % de la pondération de l'activité en cause jusqu'à concurrence de 10 % de cette pondération. Lorsque le contexte de réalisation de l'activité d'évaluation permet aux étudiants le recours aux outils de référence linguistique, le département peut déterminer une pénalité pour les erreurs de langue qui va au-delà de 10 % de la note, jusqu'à un maximum de 20 %. [...]
Note de passage Article 6.1.13	La note de passage est de 60 %. Dans le respect des normes et des règles prévues dans la présente politique, cette note traduit l'atteinte minimale des objectifs d'un cours. [...]
Évaluation de la présentation et de la composition des travaux Article 6.1.9	Les étudiants doivent présenter leurs travaux écrits en conformité avec les normes de présentation adoptées par le Cégep et les règles départementales qui viennent les préciser et les compléter. [...] Dans le cas où le type de travail ne se prête pas à ces normes de présentation, les étudiants suivent les consignes du professeur. http://cegep-ste-foy.qc.ca/presentationtravaux
Remises des travaux Article 6.1.11	Les rapports et les travaux exigés des étudiants doivent être remis au professeur à la date, à l'heure et selon les modalités précisées par le professeur. Une journée ouvrable de retard est tolérée entraînant une pénalité de 15 % de la note. Au-delà de ce délai, le travail est refusé et la note « 0 » est attribuée
Présence lors des activités d'évaluation sommative (examens) Article 6.1.10	Que ce soit en présentiel ou en ligne, l'étudiant est tenu de faire et de remettre chacune des activités d'évaluation sommative au moment prévu au plan de cours. Dans le cas d'une activité d'évaluation sommative en présentiel ou en mode synchrone, une absence à l'activité entraîne un échec à cette activité, à moins d'une raison de force majeure. Dans le cas d'une activité d'évaluation sommative en mode asynchrone, la période de retard prévue à l'article 6.1.11 ne s'applique pas puisque l'activité n'est pas un travail. Ainsi, la remise de cette activité d'évaluation sommative après le moment prévu à cet effet entraîne un échec à cette activité, à moins d'une raison de force majeure.
Plagiat et fraude Article 6.1.12	Tout acte de plagiat, de tricherie et de fraude sera sanctionné. Constitue notamment un plagiat, une tricherie ou une fraude tout acte de copier, de fournir ou de recevoir volontairement de l'information lors d'un examen, de reproduire en tout ou en partie le travail d'une autre personne, qu'il s'agisse d'un document imprimé, multimédia ou électronique, sans y faire expressément référence, de remplacer un étudiant ou de se faire remplacer lors d'un examen, de remettre un travail réalisé par une autre personne, d'obtenir, posséder ou utiliser frauduleusement des questions ou réponses d'examen, d'utiliser du matériel, des applications, des sites Web ou des ressources non autorisés, de se faire aider par une autre personne lorsqu'il est demandé de réaliser l'évaluation seul, de falsifier les résultats de travaux ou d'examens.

	<p>En cas de plagiat, de tricherie, de fraude ou de coopération à un plagiat, une tricherie ou une fraude lors d'un examen (en présentiel ou en ligne) ou d'un travail, l'étudiant obtient la note « 0 » pour cet examen ou ce travail, sans exclure la possibilité d'autres sanctions compte tenu de la gravité de la faute ou du fait qu'il s'agisse d'une récidive. Chaque cas de plagiat ou de fraude doit être signalé à la Direction des études ou à la Direction de la formation continue et des services aux entreprises selon les modalités en vigueur.</p> <p>http://cegep-ste-foy.qc.ca/soisintegre</p>
La révision de notes Article 6.5	Tout étudiant a le droit de demander une révision de notes. Il est responsable de fournir tous les documents pertinents qui lui ont été remis.
L'incomplet Article 6.8	À la suite d'une demande écrite d'un étudiant et sans autre justification, le Cégep peut lui accorder un incomplet s'il est dans l'impossibilité de terminer sa session en raison de la COVID-19, sans autre justification. L'incomplet sans justification médicale peut être accordé à l'étudiant seulement pour des raisons exclusivement liées à la COVID-19.
Responsabilité de l'étudiant Articles 7.1.2 et 7.1.3	<p>L'étudiant a la responsabilité d'être présent, ponctuel, de participer avec intérêt aux cours, aux séances de laboratoire, aux stages ou autres activités d'apprentissage, et ce, qu'ils soient donnés en mode présentiel, à distance ou hybride, avec le matériel nécessaire, de se conformer aux règles départementales prévues à cet effet et, en cas d'absence, de s'informer de ce qu'il a manqué.</p> <p>L'étudiant a la responsabilité, dans la mesure du possible, de prévenir son professeur lorsqu'il doit s'absenter.</p>

Réussite scolaire

Le Cégep de Sainte-Foy a adopté un règlement pour favoriser la réussite scolaire. En raison de la COVID-19, des assouplissements temporaires ont été apportés à ce règlement afin de tenir compte du contexte d'enseignement à distance.

Sujet	Règle
PRESTATION DES COURS Article 6	[...] En raison de la COVID-19, l'assouplissement à l'article 1 du RREC permet que le travail personnel dirigé (travaux longs, périodes d'échanges en ligne et autres modalités) soit un type d'activité d'apprentissage au même titre que l'enseignement magistral, les laboratoires et les stages ; il peut être comptabilisé en tant que période d'enseignement. Ainsi, le cours est constitué des activités qui ont été ciblées et spécifiées explicitement par le professeur pour totaliser le nombre de périodes prévues par la ministre.
PRÉSENCE AUX COURS Article 7	La présence au cours et la réalisation des activités d'apprentissages du cours, tel que défini dans l'article 6, est obligatoire. Lorsque l'étudiant n'assume pas cette responsabilité et qu'il s'absente à plus de 15 % des périodes du cours, son professeur pourra alors cesser de corriger ses activités d'évaluation et pourra, dès lors, ne pas cumuler ses résultats.

Annexe - Compétences détaillées

00SU – Effectuer le développement d'applications Web transactionnelles			
	Critères de performance	Habiletés	Contenus
0 0 S U - 1	1.1 Analyse juste des documents de conception. 1.2 Détermination correcte des tâches à effectuer.	1.1.1 Analyser les documents de conception afin de déterminer les tâches de développement à réaliser. 1.2.1 Déterminer les tâches à effectuer afin de livrer les fonctionnalités demandées.	Catalogue de produit Récits utilisateurs Schémas d'interface (maquettes et diagramme de navigation) Éléments d'authentification et d'autorisation Modèle de données Planification des activités de développement Découpage en tâches Pointage des récits utilisateurs Critères d'acceptation Outil de planification et de suivi de projet (par exemple, logiciel « Pivotal Tracker », technique des « post-it »)

00S-2	<p>2.1 Installation correcte de la plateforme de développement Web et du système de gestion de base de données de développement.</p> <p>2.2 Installation correcte des logiciels et des bibliothèques.</p> <p>2.3 Configuration appropriée du système de gestion de versions.</p> <p>2.4 Importation correcte du code source.</p>	<p>2.1.1 / 2.2.1 / 2.3.1 Installer et configurer l'environnement de travail complet pour réaliser du développement Web transactionnel, notamment un environnement de développement intégré, un cadriciel pour les tests automatisés, un système de gestion de bases de données, et un système de gestion de versions.</p> <p>2.4.1 Importer le projet de l'application.</p>	<p>Infrastructure technologique pour le développement d'une application Web :</p> <ul style="list-style-type: none"> ● Kits de développement et outils de débogage (côté client et côté serveur). ● Cadriciels (par exemple, pour les tests ou l'architecture logicielle). ● Gestionnaire de dépendances (par exemple, Nuget pour .NET). ● Serveur et logiciel d'intégration continue ● Serveur et logiciel de déploiement automatique (pré-production). ● SGBD ou cadriciel ORM système de gestion des versions.
00S-7	<p>7.1 Application rigoureuse des plans de tests.</p> <p>7.2 Revues de code et de sécurité rigoureuses.</p> <p>7.3 Pertinence des correctifs.</p> <p>7.4 Respect des procédures de suivi des problèmes et de gestion des versions.</p> <p>7.5 Respect des documents de conception.</p>	<p>7.1.1 / 7.1.3 Coder, exécuter et faire passer tous les tests afin d'assurer la qualité de l'application.</p> <p>7.2.2 Réaliser des revues de code et de sécurité fréquentes et réusiner le code en conséquence.</p> <p>7.4.1 Respecter les procédures de gestion des versions pour assurer le partage des sources et l'intégration des fonctionnalités.</p> <p>7.4.2 Respecter les procédures de suivi des problèmes et de gestion des versions pour s'assurer des corrections apportées à une version.</p> <p>7.5.1 S'assurer que les fonctionnalités réalisées sont conformes aux besoins du client.</p>	<p>Tests unitaires de fiabilité, de robustesse et d'acceptation</p> <p>Cadriciel pour les tests unitaires et d'acceptation automatisés</p> <p>Revue de code et de sécurité</p> <p>Réusinage du code</p> <p>Principes et bonnes pratiques de conception et de programmation</p> <p>Respect de l'architecture de l'application</p> <p>Bonnes pratiques de gestion des versions des programmes (par exemple, fréquence et description des « commit », création et intégration des branches, etc.)</p> <p>Documentation des bogues</p> <p>Traçabilité des problèmes ou des bogues dans le système de gestion de version</p> <p>Acceptation des récits utilisateurs</p>

O O S U - 9	9.1 Détermination correcte de l'information à rédiger.	9.1.1 / 9.2.1 S'assurer de la clarté du code et des tests, notamment le nommage.	Revues de code
	9.2 Notation claire du travail effectué.	9.1.2 / 9.2.2 Réusiner le code pour en améliorer la clarté.	Réusinage du code et des tests
		9.1.3 / 9.2.3 Commenter le code si le réusinage ne suffit pas à en assurer la lisibilité.	Emplacement et nature des commentaires
		9.1.4 / 9.2.4 Générer au besoin de la documentation pour des développeurs utilisateurs.	Outils de génération de documentation (par exemple, JavaDoc)
		9.1.5 / 9.2.5 Documenter au besoin les documents de conception.	Récits utilisateurs du catalogue

OOSY – Collaborer à la conception d'applications			
	Critères de performance	Habiletés	Contenus
00SY-3	<p>3.1 Modélisation d'une base de données conforme au besoin.</p> <p>3.2 Détermination claire des données initiales d'une base de données.</p> <p>3.3 Description claire de la logique applicative et de l'interface à générer ou à programmer.</p> <p>3.4 Modélisation orientée objet conforme aux principes d'encapsulation, d'héritage, de composition ou de polymorphisme.</p> <p>3.5 Choix ou production judicieux des algorithmes.</p> <p>3.6 Respect des règles de nomenclature.</p>	<p>3.1.1 Modéliser le modèle de la base de données pour réaliser les récits utilisateurs du catalogue produit.</p> <p>3.2.1 Répertorier les informations de base « fixes » du domaine d'affaire qui devront être insérées dans la base de données et livrées avec l'application.</p> <p>3.3.1 / 3.4.1 / 3.5.1 Décrire dans les documents de conception la logique applicative pour réaliser les récits utilisateurs.</p> <p>3.3.2 / 3.4.2 / 3.5.2 Modéliser les classes de l'application en respectant et en exploitant les principes d'encapsulation, d'héritage, de composition ou de polymorphisme.</p> <p>3.6.1 Respecter la nomenclature technique de l'entreprise pour faciliter la compréhension commune.</p>	<p>Modèle de base de données</p> <p>Bonnes pratiques de modélisation des bases de données, notamment relationnelles (par exemple, élimination de la redondance et normalisation, clés artificielles, etc.)</p> <p>Données de base d'un domaine d'affaire (par exemple, liste de villes, énumération de catégories, taxes, etc.)</p> <p>Description détaillée des récits utilisateur et des critères d'acceptation</p> <p>Description et documentation des tâches pour un récit utilisateur, par exemple:</p> <ul style="list-style-type: none"> • Algorithmes • Maquettes d'interface • Modèle de classes (associations et héritage, abstraction et polymorphisme) • Patrons de conception • Considérations de sécurité <p>Exemples de nomenclature technique (convention de nommage)</p>

O O S Y - 4	<p>4.1 Représentation graphique correcte des différents modèles.</p> <p>4.2 Rédaction correcte des plans de tests unitaires, d'intégration, fonctionnels ou d'acceptation.</p> <p>4.3 Participation active à la revue de conception.</p> <p>4.4 Utilisation du vocabulaire approprié.</p> <p>4.5 Respect des normes, des méthodes et des bonnes pratiques en matière de développement d'applications.</p>	<p>4.1.1 Représenter les différents modèles de conception en respectant les formalismes de modélisations utilisés.</p> <p>4.2.1 Planifier les approches de tests à réaliser dans le projet.</p> <p>4.3.1 Participer à la revue de conception et évaluer les choix de conception au regard des exigences du client et des contraintes techniques.</p> <p>4.4.1 Utiliser le vocabulaire du domaine d'affaire de l'entreprise dans les documents de conception.</p> <p>4.5.1 Respecter des normes, méthodes et bonnes pratiques de développement d'applications lors de la production des documents de conception.</p>	<p>Représentations d'une architecture logicielle (par exemple, diagrammes de composants, diagrammes de classes, modèles de données, diagrammes d'états)</p> <p>Approches et types de tests</p> <p>Plan de tests</p> <p>Patrons de conception de tests (par exemple: Page Object Pattern)</p> <p>Bases de données de test</p> <p>Plateformes de tests</p> <p>Pratiques agiles d'inspection (par exemple: planification de l'itération, mêlée quotidienne, revue d'itération, rétrospective)</p> <p>Mise à jour du catalogue utilisateur et priorisation des récits utilisateurs</p> <p>Utilisation du vocabulaire du domaine d'affaire de l'entreprise dans les documents de conception</p> <p>Glossaire d'affaire</p> <p>Respect des normes, méthodes et bonnes pratiques de développement d'applications lors de la production des documents de conception</p>
----------------------------	---	---	---