CERTIFICATION
Professional Scrum Developer (.NET)

Durée
5 jours

Description
Le cours « Professional Scrum Developer » de Pyxis offre une expérience intensive unique aux développeurs de logiciels. Ce cours guide les équipes sur la façon de transformer les exigences relatives au produit en incréments de logiciel fonctionnel à l'aide du processus Scrum, de Visual Studio 2010 et de pratiques modernes d'ingénierie logicielle. Les participants travailleront en équipes auto-organisées et utiliseront une instance commune de Team Foundation Server 2010.

L'apprentissage de Scrum se fera par une combinaison d'enseignement magistral, de démonstrations, de discussions et d'exercices pratiques. Les participants apprendront à appliquer Scrum correctement tout en étant coachés et examinés de façon critique par le formateur.

Objectifs
À la fin du cours les participants seront en mesure de réaliser les activités suivantes :

- Former des équipes qui sont efficaces;
- Explorer et comprendre l'architecture Brownfield traditionnelle;
- Définir les attributs de qualité, les critères d'acceptation et « terminé »;
- Créer des *builds* automatisés;
- Gérer les correctifs logiciels;
- Assurer la découverte et l'élimination de bogues;
- Planifier les livraisons et les sprints;
- Estimer les éléments du carnet de produit;
- Créer et gérer un carnet de tâches du sprint (*sprint backlog*);
- Tenir des revues de sprint efficaces;
- Améliorer le processus en place à l'aide des rétrospectives;
- Utiliser une architecture émergente pour éviter la dette technique;
- Utiliser le développement piloté par les tests (*Test-Driven Development* ou TDD) comme outil de conception;
- Paramétrer et optimiser l'intégration continue;
- Utiliser l'analyse d'impact des tests afin de réduire les temps d'essai;
- Gérer les développements à l'aide de SQL Server de manière Agile;
- Utiliser efficacement le remaniement de code (*refactoring*) .NET et T-SQL;
- Créer, déployer et tester des bases de données SQL Server;
- Créer et gérer des protocoles de test et des cas d'essai;
- Créer, exécuter, enregistrer et repasser des tests manuels;
- Définir une stratégie et un code de branchement;
- Écrire du code plus facile à maintenir;
- Déceler et éliminer les dysfonctionnements humains ou relatifs au processus;
- Inspecter et améliorer le processus de développement logiciel de votre équipe.

Public visé
Ce cours est destiné à tout membre d'une équipe de développement logiciel, soit programmeurs, testeurs, développeurs de base de données, architectes, etc. Le cours est ouvert aux particuliers mais aussi aux équipes complètes souhaitant participer au cours ensemble.

CERTIFICATION
Professional Scrum Developer (.NET)

Les participants s'auto-organiseront afin de former des équipes Scrum multifonctionnelles. Les équipes doivent avoir un ensemble de compétences propres à l'étude de cas choisie.

Les Product Owners, les Scrum Masters et autre partie prenante sont les bienvenus. Toutefois, il faut savoir que les participants doivent participer activement aux activités de développement au sein d'une équipe Scrum.

Préalable

Avant de suivre ce cours, il est important d'avoir lu le **guide Scrum** de Ken Schwaber.

Contenu

MODULE 1 – INTRODUCTION
Ce module permet à chaque participant d'apprendre à connaître le formateur et les autres participants. On y présente le programme ainsi que le déroulement du cours. Puis, il y aura formation des équipes Scrum afin que les activités de développement des équipes puissent débuter, soit : la création (*forming*), la turbulence (*storming*), l'harmonisation (*norming*) et la performance (*performing*).

- Présentation du formateur et des participants;
- Programme du cours « Professional Scrum Developer »;
- Calendrier du cours;
- Logistique;
- Formation des équipes;
- Rétrospective.

MODULE 2 – PRINCIPES FONDAMENTAUX
Ce module permet de comprendre Scrum, c'est-à-dire les différents rôles, les artefacts, etc. Les équipes mettent Scrum en pratique en simulant un sprint de plusieurs jours pour le développement d'un produit, y compris les rencontres de planification, de revue et de rétrospective de sprint.

- Aperçu de Scrum;
- Rôles de Scrum;
- Cérémonies de Scrum;
- Artefacts de Scrum;
- Simulation;
- Rétrospective.

MODULE 3 – MISE EN ŒUVRE DE SCRUM DANS VISUAL STUDIO 2010
Ce module démontre comment mettre en œuvre Scrum dans Visual Studio 2010 à l'aide de Scrum Process Template*. Les équipes apprennent le mappage des différents concepts de Scrum et la façon dont ils sont insérés dans l'outil. Une fois connectés à une version partagée de Team Foundation Server, les membres des équipes poursuivent l'exercice de simulation. Cette fois-ci, ils utilisent Visual Studio pour gérer le développement du produit.

- Mappage de Scrum avec Visual Studio 2010;
- Éléments de travail relatifs aux scénarios utilisateurs;
- Éléments de travail relatifs aux tâches;
- Éléments de travail relatifs aux bogues;
- Démonstration;
- Simulation;

- Rétrospective.

*Ce cours peut être donné à partir d'une version bêta de Visual Studio 2010 ou du modèle de processus. C'est pourquoi, il se peut que certaines fonctionnalités annoncées ne soient pas disponibles ou démontrées. Si vous avez des questions, veuillez communiquer avec nous à l'avance.

MODULE 4 – ÉTUDE DE CAS
Dans ce module, on présente aux équipes leur espace de problème (*problem domain*) pour la semaine. Une réunion de démarrage tenue par le Product Owner (le formateur) donne le ton aux pourquoi et aux quoi qui surviendront au cours des sprints. Les équipes définiront par la suite les attributs de qualité du projet ainsi que la notion de « terminé ». Le code d'application traditionnel sera téléchargé, créé et exploré. Ainsi, il sera possible de découvrir et de signaler tous bogues.

- Présentation de l'étude de cas;
- Téléchargement du code source, construction et exploration de l'application;
- Définition des attributs de qualité du projet;
- Définition de « terminé »;
- Comment classer les bogues dans Visual Studio 2010?;
- Rétrospective.

MODULE 5 – CORRECTIFS (*HOTFIXES*)
Dans ce module, les équipes vivent une expérience *brownfield* traditionnelle. Elles sont forcées à analyser l'architecture et le code de l'application existante afin de déceler et de corriger les bogues ayant la priorité la plus élevée selon le Product Owner. Les équipes apprennent les meilleures pratiques entourant la découverte d'un bogue, les tests validant son existence ainsi que la correction, la validation et la clôture de ce bogue.

- Utilisation d'Architecture Explorer pour la visualisation et l'exploration;
- Création d'un test unitaire pour valider l'existence d'un bogue;
- Découverte et correction du bogue;
- Validation et fermeture du bogue;
- Rétrospective.

MODULE 6 – PLANIFICATION
Dans ce module particulièrement court, les équipes apprennent à planifier les sprints et les livraisons dans Visual Studio 2010. Elles définissent les objectifs et toute autre information importante relative à la planification.

- Planification des sprints par rapport à la planification des livraisons (*releases*);
- Planification des livraisons et carnet de produit (*product backlog*);
- Ordonnancement du carnet de produit;
- Critères d'acceptation et tests;
- Planification de sprint et carnet de tâches du sprint;
- Création des tâches du sprint et des liens entre celles-ci;
- Rétrospective.

À ce stade, les équipes ont acquis des connaissances de Scrum, de Visual Studio 2010 et de l'étude de cas leur permettant de commencer à développer des incréments de logiciel potentiellement livrables qui répondent à la définition de « terminé ».

CERTIFICATION
Professional Scrum Developer (.NET)

MODULE 7 – ARCHITECTURE ÉMERGENTE

Ce module permet d'apprendre les pratiques et outils architecturaux que les équipes peuvent utilisés pour avoir une conception juste sur laquelle asseoir le développement de nouvelles fonctionnalités. Les équipes apprennent comment Scrum soutient les bonnes pratiques architecturales et de conception. Après discussion, on présente aux équipes le carnet de produit ordonné par le Product Owner. Ainsi, elles peuvent choisir les fonctionnalités qu'elles prévoient livrer au cours du sprint en cours.

- Architecture et Scrum;
- Architecture émergente;
- Principes, modèles et pratiques;
- Outils de modélisation de Visual Studio 2010;
- Langage UML et diagrammes de couche;
- Sprint 1;
- Rétrospective.

MODULE 8 – DÉVELOPPEMENT PILOTÉ PAR LES TESTS

Ce module présente le développement piloté par les tests (TDD) comme outil de conception. Il couvre aussi comment le mettre en œuvre à l'aide de Visual Studio 2010. Afin de maximiser la productivité et la qualité, les équipes Scrum devraient favoriser l'intégration continue pour introduire régulièrement les changements de code de chaque membre de l'équipe et exécuter les tests de régression. Le remaniement de code est également défini et démontré de concert avec l'analyse d'impact des tests de Visual Studio pour réexécuter de manière efficace uniquement les tests touchés par le remaniement de code.

- Intégration continue;
- Team Foundation Build;
- Développement piloté par les tests;
- Remaniement de code;
- Analyse d'impact des tests;
- Sprint 2;
- Rétrospective.

MODULE 9 – DÉVELOPPEMENT D'UNE BASE DE DONNÉES AGILE

Ce module révèle un petit secret aux développeurs de base de données SQL Server – ils peuvent être Agiles eux aussi. En utilisant les projets de base de données de Visual Studio 2010, les développeurs de base de données peuvent faire partie intégrante de l'équipe. Les participants apprennent à appliquer des techniques Agiles relatives aux bases de données dans Visual Studio afin de soutenir le cycle de développement de SQL Server 2005/2008/2008R2.

- Développement d'une base de données Agile;
- Projets de base de données de Visual Studio;
- Importation du schéma et des scripts;
- Construction et déploiement;
- Génération de données;
- Tests unitaires;
- Sprint 3;
- Rétrospective.

CERTIFICATION
Professional Scrum Developer (.NET)

MODULE 10 – LIVRAISON

Il est important que les équipes sachent que ce n'est pas parce qu'elles aiment une fonctionnalité que le Product Owner l'aimera aussi. Ce module réexamine les critères d'acceptation puisqu'il couvre les tests d'acceptation. En affinant les critères d'acceptation en étapes de contrôle manuel, les équipes peuvent exécuter les tests, enregistrer les résultats et établir les relevés de bogues de plusieurs façons. Les contrôles manuels seront établis et effectués à l'aide du gestionnaire de test de Microsoft. À la fin du sprint, alors qu'il y a livraison d'un incrément fonctionnel du logiciel, les équipes apprennent pourquoi elles devraient créer un branchement de ligne de code et quand elles devraient le faire.

- Critères d'acceptation;
- Essais dans Visual Studio 2010;
- Microsoft Test Manager;
- Écriture et exécution de tests manuels;
- Branchement;
- Sprint 4;
- Rétrospective.

MODULE 11 – DYSFONCTIONNEMENT SURMONTÉ

Ce module présente les différents dysfonctionnements devant lesquels les équipes se trouvent en situation réelle : dysfonctionnements humains ou relatifs aux processus et outils. Les équipes signalent des dysfonctionnements et établissent des scénarios pour les atténuer en prenant part à des discussions et en lançant des idées. De fait, à la fin du cours, les équipes ont acquis une autonomie et perfectionné leurs connaissances de Scrum.

- ScrumBut et Scrum flasque;
- Meilleures pratiques pour travailler en équipe;
- Défis de l'équipe;
- Défis du Scrum Master;
- Défis du Product Owner;
- Défis des parties prenantes;
- Rétrospective du cours.

Évaluation

Après le cours, les participants auront la possibilité de passer un examen pour évaluer leurs connaissances. L'examen est optionnel, mais il est fortement recommandé pour continuer l'apprentissage de Scrum après le cours. Les participants qui réussissent l'examen reçoivent un document de certification. Pour en savoir davantage sur les examens de Scrum.org, visitez : http://www.scrum.org/assessments.

Pyxis est heureuse de s'associer à Scrum.org et Ken Schwaber pour offrir ce cours de formation Agile.