

Une démarche d'amélioration dans un très petit organisme

Étude de cas

Canada - Numéro 002 (Français)

Mai 2010

Revues et inspections de code



Abrégé

Acme est une société franco-canadienne qui développe des logiciels commerciaux. Son domaine d'activité est la conception de logiciels de gestion immobilière: gestion de la maintenance technique des immeubles, portail de service dédié à la gestion immobilière et de gestion de la maintenance des équipements biomédicaux.

Le projet d'intervention visait à introduire les revues et inspections de code dans le cycle de développement de la société. Ce projet pilote a permis de démontrer que les revues et inspections sont efficaces dans la détection des erreurs et l'amélioration de la qualité des produits logiciels.

L'organisation et son environnement

La société Acme est une société qui œuvre dans le domaine de conception de logiciels. Elle a été fondée en 1990 en France et en 2005. Elle a ouvert une filiale au Canada.

Le nombre d'employés dans la société est de 35 personnes. L'équipe de développement est composée de 8 personnes au Canada et de 3 personnes en France. La structure de l'entreprise ne comporte pas une équipe de d'assurance-qualité, mais certaines activités d'assurance-qualité sont effectuées par l'équipe de test qui est composée de 2 personnes.

La méthode de développement de logiciels utilisée par l'entreprise est l'eXtreme Programming. Les principaux langages de programmation utilisés sont PHP et .NET.

Point de départ

Vu le nombre d'erreurs détectées par les testeurs, ainsi que l'absence des méthodologies de revue et des inspections du processus du développement de la société, nous avons proposé la mise en place d'une méthode de revue et d'inspection du code. Ceci nous permettra de détecter les anomalies dans le code avant la phase de test. En effet, on procéderait ainsi, nous pensons qu'une grande quantité d'erreurs sera éliminée avant la phase de test.

L'objectif de la mise en place des revues est la diminution du nombre d'erreurs détectées pour diminuer les coûts relatifs à la détection tardive des erreurs et aussi pour offrir une meilleure qualité du produit livré.

Le projet d'amélioration

Pour effectuer quelques inspections au niveau de l'entreprise une discussion a été menée avec le directeur et le responsable de développement de la société. Ils ont confirmé leur engagement afin d'allouer les ressources pour les inspections. Après avoir obtenu l'accord de la direction, la méthode a été présentée aux ressources impliquées dans le projet lors d'une réunion. Cette réunion a aussi tenu le rôle d'une session de formation.

Les objectifs de la réunion étaient :

- La présentation des avantages et de l'efficacité des inspections de code lors du cycle de développement;
- L'argumentation du choix de cette méthode pour le projet pilote;
- L'initialisation de l'équipe de développement dans la méthodologie de l'inspection;
- La présentation des étapes de l'inspection, des rôles et des gabarits utilisés.

Ensuite, nous avons préparé les documents et les gabarits nécessaires. On a utilisé la trousse de déploiement des inspections disponible sur le site public du groupe de travail 24 de l'ISO/IEC JTC1/SC7. Les documents suivants ont été utilisés :

- Trousse pour revue du type inspection, version 1.
- Formulaire d'inspection.
- Formulaire d'enregistrement des anomalies.
- Liste de vérification. Cette liste de vérification est spécifique au projet et a été conçue pour le code PHP.

Après la préparation de ces documents, ils ont été distribués aux ressources concernées par les inspections.

Trois inspections ont été réalisées sur trois projets pilotes tout en respectant les étapes et les activités de ces étapes comme décrites dans la trousse d'inspection (voir référence 2).

Les Résultats

Après avoir réalisé 3 inspections, nous avons obtenu des résultats intéressants. Le nombre d'anomalies détectées était:

- 1^{ère} inspection : 20 anomalies ont été détectées et 4 améliorations ont été proposées.
- 2^{ème} inspection : 3 anomalies ont été détectées et une amélioration a été proposée.
- 3^{ème} inspection : 16 anomalies ont été détectées.

En utilisant ces nombres, des mesures ont été calculées comme le taux d'inspection, l'effort d'inspection et le taux de détection des erreurs.

Leçons apprises

Nous avons appris que le processus d'inspection n'est pas un processus facile à mettre en place au sein d'un petit organisme. En plus, si on fait une comparaison avec le modèle de maturité CMMI du Software Engineering Institute, l'organisation du processus de développement de la société est au niveau de maturité initial. Donc, il faut mettre plus d'effort pour l'intégration du processus d'inspection par rapport à un organisme d'un niveau de maturité plus élevé.

Les inspections dépendent des gens et de leur mentalité : certains développeurs étaient réticents face aux inspections parce qu'ils n'étaient pas à l'aise avec le fait que quelqu'un allait inspecter leur code. Mais, après la présentation de la méthode et le déroulement de quelques inspections, nous avons constaté que les développeurs s'habituaient à les pratiquer et ne les percevaient plus comme une menace.

En conclusion, l'implémentation des inspections peut prendre plus de temps dans le contexte où il faut implémenter la méthode graduellement pour ne pas nuire à la productivité à long terme. Nous allons continuer à faire des inspections afin de les intégrer dans le processus de développement.

Références

1. Norme ISO/CEI 29110 - Lifecycle Profiles for Very Small Entities (VSEs) – Part 1: Overview. Organisation internationale de normalisation, Genève, Suisse.
2. Site public du groupe de travail 24 de l'ISO/IEC JTC1/SC7 :
<http://profs.logti.etsmtl.ca/claporte/VSE/index.html>
3. Norme IEEE 1028:2008, IEEE Standard for Software Reviews and Audits.

Remerciements

Nous remercions le personnel de la société et tout spécialement Madame Olesea Nemtanu.

D'autres informations sont disponibles sur le site Web suivant:

<http://profs.logti.etsmtl.ca/claporte/VSE/index.html>

Partenaires du réseau international de support aux très petits organismes (TPOs)

(Very Small Entities)

- Centre d'Excellence en Technologies de l'Information et de la Communication (CETIC) (Belgique):
www.cetic.be
- Centre de recherche public Tudor (Luxembourg): www.tudor.lu
- Irish Software Engineering Research Center (LERO) (Irlande):
 - <http://www.lero.ie/research/internationalprojects/softwareprocessesforsmallenterprises/>
- Parquesoft (Colombie)
www.parquesoft.com
- Tampere University of Technology (TUT, Pori, Finlande): www.tut.fi/pori
- Thailand Software Center (Thaïlande)
www.center4vse.net