
QUALITÉ DU LOGICIEL

vendredi 8 février 2008

La qualité du logiciel

- * Qualité d'un logiciel ?
de manière informelle : respect des spécifications.
- * Particularités des logiciels par rapport à des produits matériels :
 - * Un logiciel a de fortes chances d'évoluer au cours de sa vie.(évolutions non connues à l'avance, impossible de les spécifier).
 - * Certaines caractéristiques peuvent être spécifiées formellement , d'autres ne le peuvent pas (facilité d'utilisation...).
 - * Une spécification exhaustive est, dans la majorité des cas, irréaliste (trop coûteuse).
- * Prise en compte de ces contraintes pour définir une politique qualité conforme à la fois au coût négocié et aux critères de qualité escomptés.

ISO 9000

- * ISO 9000: ensemble de normes pour la garantie de la qualité dans les relations clients-fournisseurs (pas spécialement logiciel)
- * ISO 9000-1 recommandations pour l'utilisation de ces normes
- * ISO 9001 : le standard à utiliser pour la fourniture de logiciels, la référence en matière de certification pour le logiciel
- * ISO 9003 : guide pour l'utilisation des standards ISO 9001 pour la fourniture de logiciels
- * Philosophie ISO 9001
Toute opération influençant la qualité doit être sous contrôle
Le contrôle doit être visible

CMMI

- * Capability Maturity Model Integration (CMMI), Software Engineering Institute de Carnegie Mellon University (<http://www.sei.cmu.edu/cmmi/>)
- * Repose sur une grille CMM permettant de classer les organisations
 1. Niveau initial : processus ad-hoc et chaotiques
 2. Niveau géré (managed) : planification et suivi de la mise-en-oeuvre
 3. Niveau défini (defined) : processus caractérisés, compris décrits rigoureusement dans des documents ...
 4. Niveau Quantitatively Managed : objectifs quantitatifs (statistiques)
 5. Niveau optimisé, en optimisation constante (optimizing)
- * Trop d'organisations encore proches du niveau 1
- * CMM donne des indications sur actions à entreprendre changer de niveau

La qualité du logiciel (Approche Mac Call)

- * Existence d'un grand nombre d'attributs permettant de caractériser la qualité d'un logiciel : facteurs ou critères de qualité
- * Distinction entre facteurs externes, perceptibles des utilisateurs, et facteurs internes, uniquement perceptibles par des informaticiens
- * Rarement économiquement possible d'optimiser tous ces critères; certains peuvent d'ailleurs être en partie contradictoires (efficacité et portabilité par exemple).
- * Plan qualité détermine l'importance de chacun d'eux pour le projet
- * Facteurs : vision externe, point de vue de l'utilisateur
- * Critères : vision interne, point de vue du concepteur
- * Les facteurs s'obtiennent par les critères
- * Les critères conditionnent les facteurs.

Facteurs de qualité

- * La capacité d'évolution (product revision)
 - * Maintenabilité: facilité de localisation et correction (erreurs)
 - * Souplesse : facilité d'adaptation à de nouveaux besoins,
 - * Testabilité : efforts requis pour le tester.
- * L'adaptabilité (product transition)
 - * Portabilité: peut-on utiliser le logiciel sur une autre machine?
 - * Réutilisabilité: certaines parties dans d'autres applications,
 - * Interopérabilité: Interfaçage avec un autre système ("ouverture").

Facteurs de qualité

- * Facteurs de qualité ont une influence les uns sur les autres.
- * Facteurs suivants diminuent l'efficacité:
 - * Intégrité (nécessité d'introduire des vérifications), Facilité d'emploi (nécessité d'introduire des interfaces sophistiqués), Maintenabilité (sacrifier l'efficacité pour la lisibilité), Portabilité (les structures portables ne sont pas nécessairement les plus efficaces), Testabilité, souplesse, réutilisabilité, interopérabilité.
 - * Facteurs suivants diminuent l'intégrité: souplesse, réutilisabilité, interopérabilité.

Facteurs de qualité (évalués par les utilisateurs)

- * Correction (correctness) : Capacité d'un logiciel à satisfaire les besoins des utilisateurs tels qu'ils sont exprimés dans les spécifications.
- * Robustesse (robustness, resilience) : Capacité d'un logiciel à réagir de manière appropriée à la présence de conditions anormales.
- * Intégrité (integrity) : Capacité d'un système à résister à des mises à jour inconsistantes.
- * Efficacité (efficiency) : Aptitude du logiciel à utiliser d'une façon optimale les ressources physiques (espace mémoire, temps d'unité centrale)
- * Facilité d'utilisation ou ergonomie (usability, friendliness) : Aptitude du système à pouvoir être utilisé avec un minimum d'efforts.
- * Facilité d'apprentissage (learnability) : Facilité avec laquelle des utilisateurs ayant des compétences différentes peuvent apprendre à utiliser un logiciel.

Facteurs de qualité (évalués par les utilisateurs)

- * **Fonctionnalité** : Étendue des possibilités offertes par un logiciel.
- * **Ponctualité** : Capacité d'un logiciel à être livré à la date prévue.
- * **Disponibilité (availability)** : Capacité d'un système à offrir ses services à un moment donné.
- * **Fiabilité (reliability)** : Capacité d'un système à fournir ses services pendant une période de temps donnée.
- * **Sûreté (safety)** : Caractéristique d'un système à ne pas causer de dommages à ses utilisateurs ou à son environnement.
- * **Sécurité (security)** : Capacité d'un système à résister à des intrusions accidentelles ou volontaires.

Critères de qualité (évalués par les informaticiens)

- * **Maintenabilité ou adaptabilité (maintainability or adaptability) :** Aptitude d'un logiciel à faciliter les opérations requises pour localiser et corriger une erreur alors que le système est en phase d'exploitation. Aptitude d'un logiciel à être adapté à une modification de ses spécifications.
- * **Portabilité (portability) :** Facilité avec laquelle un logiciel peut être transféré d'un environnement logiciel ou matériel à un autre.
- * **Réutilisabilité (reusability) :** Aptitude d'un composant logiciel à être réutilisé dans des applications différentes.
- * **Interopérabilité, compatibilité (interoperability, compatibility) :** Facilité avec laquelle un logiciel peut interagir (s'échanger des informations) avec d'autres ou être combinés avec d'autres.

Critères de qualité (évalués par les informaticiens)

- * Modularité : Facilité avec laquelle un logiciel peut être découpé en éléments pouvant être compris isolément et remplacés par d'autres éléments offrant les mêmes fonctionnalités.
- * Lisibilité, complexité ou compréhensibilité (complexity, understandability) : Facilité de compréhension de la logique d'un élément logiciel à partir de la lecture de ses spécifications ou de son code source
- * Traçabilité (tracability) : Facilité avec laquelle on peut descendre d'un élément de spécification des besoins vers le code correspondant et remonter en sens inverse.
- * Testabilité, vérifiabilité (testability, verifiability) : Aptitude d'un logiciel à se prêter à une vérification d'adéquation aux spécifications, le système étant en phase d'exploitation

Facteurs de qualité

- * Les caractéristiques opérationnelles (product operation)
 - * Conformité aux besoins: le produit fait-il ce que je souhaite?
 - * Fiabilité: le fait - il correctement dans tous les cas?
 - * Efficacité: utilise-t-il au mieux le matériel?
 - * Intégrité: protégé contre les intrusions? niveau de sécurité suffisant? Facilité d'emploi : apprentissage, mise en œuvre, interprétation