

Résumé du cours de Qualité Logiciel de 4IF de Régis Aubry

Décembre 2010

Table des matières

1	Sensibilisation à la démarche qualité	5
1.1	Historique	5
1.2	La qualité dans l'entreprise	5
1.3	Les 5 principes de la qualité	5
1.3.1	Définition/Spécification des exigences	5
1.3.2	Conformité aux exigences	5
1.3.3	Prévention plutôt que correction	5
1.3.4	Critère de réalisation : le zéro défaut	5
1.3.5	Évaluer le coût de la non-conformité	5
1.4	La qualité : qui est concerné ?	6
1.5	La qualité : comment ?	6
1.6	Quelques outils pour la résolution de problèmes et la recherche de la qualité	6
1.6.1	Résolution de problèmes	6
1.6.2	Quelques méthodes pour la démarche qualité	6
1.6.3	Quelques outils de la qualité	7
1.7	Approches utilisées pour engager une réflexion qualité	8
1.7.1	Approche socio-économique	8
1.7.2	Approche "Démarche de changement"	8
1.7.3	Approche "Élaboration d'un projet d'établissement"	8
1.8	Conclusion	8
2	La qualité totale : un objectif	9
2.1	Définition de la QUALITE TOTALE	9
2.2	Les Objectifs de la démarche QUALITE TOTALE	9
2.2.1	Objectif Réel	9
2.2.2	Objectif Manifesté	9
2.2.3	Objectifs Latents	9
2.2.4	Convergence de ces Objectifs	9
2.3	Processus de mise en oeuvre de la QUALITE TOTALE	9
2.4	Introduction des principaux concepts d'un Système Qualité	10
2.4.1	Vue synthétique des principaux concepts	10
2.4.2	La politique Qualité	10
2.4.3	Maîtrise et appréciation de la Qualité	10
2.4.4	Le cycle de la qualité (ou Processus d'amélioration)	10
2.4.5	L'assurance Qualité	11
2.4.6	La notion de procédure (disposition préventive)	11
2.4.7	Le système Qualité	11
2.4.8	Le management par la Qualité	11
2.5	Différents types de systèmes qualité	11
3	Concepts de Qualité pour le Logiciel	12
3.1	Concepts de base liés à la qualité du logiciel	12
3.1.1	Définition de la qualité	12
3.1.2	Qualité et satisfaction des besoins	12
3.1.3	Maîtriser la qualité des besoins du client avant tout !	12
3.2	Les non-conformités : perception de la non qualité du logiciel	12
3.3	Le coût de la non-qualité	13
3.3.1	Répartition des erreurs entre les phases	13
3.3.2	Relation entre le coût et sa qualité	13

3.4	Émergence de la fonction qualité dans le développement de logiciel	13
3.4.1	Position de la fonction qualité dans le logiciel	13
3.4.2	Parallèle avec l'industrie	13
3.5	Qualité attendue par les différents acteurs	13
3.6	L'approche management par la qualité pour le logiciel	14
3.7	Conclusion	14
4	Outils pour la résolution de problèmes et la recherche de la qualité	15
4.1	Introduction : Qu'est-ce qu'un problème ?	15
4.2	Quelles démarches pour la qualité ?	15
5	La norme ISO - 9000	16
5.1	Les 4 verbes de la norme	16
5.2	Approche processus	16
5.2.1	Exigences de la norme	16
5.2.2	Documents exigés par la norme	16
5.2.3	Enregistrements exigés par la norme	17
5.2.4	Plan d'assurance qualité	17
5.3	Critiques de la norme ISO-9000	17
5.4	Certification du projet qualité	17
6	Mise en Oeuvre d'un système qualité pour la production de logiciel	18
6.1	Cadre général pour la mise en oeuvre d'une démarche qualité pour la production de logiciel	18
6.2	La structure documentaire d'un système qualité	18
6.2.1	Documentation de niveau 1	19
6.2.2	Documentation de niveau 2	19
6.2.3	Documentation de niveau 3	19
6.3	1ere discipline du support : l'assurance qualité	19
6.3.1	Rôle de l'assurance qualité	19
6.3.2	AQ au niveau de l'entreprise	19
6.3.3	AQ au niveau du projet	20
6.3.4	Synthèse	20
6.4	Environnement organisationnel de la qualité dans un projet	20
6.5	Suivi de la démarche qualité	20
6.6	Conclusion	20
7	Techniques d'amélioration de la qualité	22
7.1	Avant-propos	22
7.2	2ème discipline du support : Vérification & Validation (V&V)	22
7.2.1	Introduction aux techniques de contrôle	22
7.2.2	les contrôles qualité (ou formels)	22
7.2.3	les contrôles techniques (sur le fond)	22
7.2.4	contrôle de situation (audit)	23
7.2.5	mise en oeuvre des contrôles de qualité	23
7.3	3ème discipline du support : Gestion de configuration (GdC)	23
7.3.1	intro à la GdC	23
7.3.2	apports de la GdC	23
7.3.3	champs d'application de la GdC : configurations de référence (ou référentiels)	24
7.3.4	comment organiser une GdC	24
7.3.5	interaction entre la GdC et la production de logiciel	24
7.3.6	Conclusion	25
7.4	4ème discipline du support : Test & Évaluation Indépendante (T&E)	25
7.4.1	objectif de T&E	25
7.4.2	principales livraisons	25
7.4.3	avis de livraison à l'Intégration & Validation	25
7.5	Conclusion	25
8	Gestion de la Documentation	26
8.1	Introduction	26
8.2	Plan type de la documentation	26
8.3	Résumé des Documents nécessaires en fonctions de la taille des projets	27

9	Techniques de Controle : Reviews	28
9.1	Revue Techniques	28
9.2	Inspections ou Walk-Throughs	28
9.3	Comment organiser les réunion de revues structurée	29
9.4	Règles de bonnes pratiques des Inspections	29
10	Approche Produit : « Modélisation de la qualité d'un produit logiciel »	30
10.1	Introduction	30
10.2	2e partie : 1er niveau de décomposition : Les facteurs	30
10.3	Deuxième partie : 2e niveau de décomposition : les critères	31
10.4	Troisième partie : 3e niveau de décomposition : les métriques	32
10.5	Conclusion	33
11	Les bonnes pratiques en informatique : les référentiels	34
11.1	Intro	34
11.2	Normes ISO/CEI 12220	34
11.2.1	Objectif d'un modèle	34
11.2.2	approche globale "modèle par processus"	34
11.2.3	la norme ISO 12207	34
11.2.4	Conclusion	34
11.3	CCMI (Capability Maturity Model Integrated)	35
11.3.1	définition	35
11.3.2	historique	35
11.3.3	présentation générale	35
11.3.4	méthode de mise en oeuvre	35
11.3.5	structure du modèle	35
11.3.6	organisation	35
11.3.7	représentation étagée	35
11.4	ITIL (Information Technology Infrastructure Library)	36

Avant-propos

Définition de la qualité (NF ISO 8402) : Ensemble des caractéristiques d'une entité qui lui confèrent l'aptitude à satisfaire des besoins exprimés ou implicites.

- Qualité = politique orientée client.
- Le produit doit être adapté aux besoins du client, et non pas être le meilleur produit possible.
- Formalisation des méthodes.
- Maîtriser les coûts fait partie de la qualité.
- La qualité comprend tout, y compris par exemple le marketing et le SAV.
- Démarche rigoureuse (et non rigoriste!).
- Est-ce que la qualité a un coût ?
 - Oui car investissement (formation, méthode, réorganisation)
 - Mais selon Ph. CROSBY : « QUALITY IS FREE »=D
- La qualité concerne 3 aspects de l'entreprise : Système qualité (pour mettre en oeuvre le management de la qualité), Conception/Réa (génie logiciel), Produit (sa qualité intrinsèque).
- 3 points de vue sur la qualité : utilisateurs, développeurs, décideurs
- Définitions :
 - Assurance qualité : Dispositions spécifiques et modalités d'application pour le projet de l'approche qualité globale.
 - Élaboration de la qualité : Mise en oeuvre des règles, des méthodes, des dispositions qui sont propres à l'entreprise. Buts = détecter et prévenir les non-conformités.
 - Contrôle qualité : contrôle à des points-clefs, validation de chaque étape (produits intermédiaires).

Chapitre 1

Sensibilisation à la démarche qualité

1.1 Historique

→ useless

1.2 La qualité dans l'entreprise

Une démarche qualité impacte sur la structure, l'organisation, les méthodes et la communication de l'entreprise.

1.3 Les 5 principes de la qualité

1.3.1 Définition/Spécification des exigences

- réelle difficulté à recueillir les vrais besoins
- solution = relation client/fournisseur

1.3.2 Conformité aux exigences

- c'est le point de départ de la définition de la Q
- à chaque transmission entre différents acteurs (client-entreprise, ou différentes équipes au sein de l'entreprise, ex : marketing-cahier des charges), on perd de la qualité. $Q \text{ souhaitée} > Q \text{ exprimée} > Q \text{ interprétée} > Q \text{ définie} > Q \text{ conçue} > Q \text{ réalisée} > Q \text{ livrée} > Q \text{ perçue}$.
- solution = cahier des charges bien défini, V&V (validation&vérification), méthodologie...

1.3.3 Prévention plutôt que correction

- dans le modèle RdP, tout écart entre besoin et résultat = non-conformité
- solution = prévoir des manières de travailler pour évaluer les risques et les supprimer (prévention)

1.3.4 Critère de réalisation : le zéro défaut

- faire bien (conforme aux exigences) du 1er coup et à chaque fois
- réalisation de manière progressive, avec des objectifs intermédiaires

1.3.5 Évaluer le coût de la non-conformité

- non-conformité = 10 à 40% du prix de revient d'un produit/service
- objectif = traquer processus non conformes à l'origine des coûts
- évaluation des coûts de tous les processus
- coûts de :
 - prévenir/évaluer : COC (coût de la conformité)
 - faire : coût de conception/réalisation
 - refaire : CONC (coût de la non-conformité)

1.4 La qualité : qui est concerné ?

Tout le monde.

Il faut responsabiliser chaque membre de l'équipe dans la mesure de ses possibilités.

Il faut donner des responsabilités claires.

Il faut évaluer si le travail est bien fait.

1.5 La qualité : comment ?

- Clef 1 : Point de vue/état d'esprit qualité
- Clef 2 : Mesurer la Q avec des outils
- Clef 3 : La recherche de l'excellence (Plan -> Do -> Check -> Action : cycle universel d'amélioration de la qualité = roue de Deming)

1.6 Quelques outils pour la résolution de problèmes et la recherche de la qualité

1.6.1 Résolution de problèmes

- Définition "problème" : écart entre la situation existante et la situation idéale (l'objectif)
- 2 approches : déductive/inductive (cc @armandrossius)
- La résolution de problèmes par la démarche qualité = approche inductive mais rationnelle : exposer le problème (le comprendre, fixer un objectif, évaluer l'écart entre situation actuelle et objectif) → analyser les causes (les rechercher) → appliquer des solutions correctives (les définir, les mettre en oeuvre, les institutionnaliser).
- Différence avec méthode classique = analyse des causes au lieu de faire appel à l'intuition et/ou l'expérience.
- La résolution de pb par la démarche qualité utilise :
 - le point de vue qualité
 - démarche en 7 étapes
 - les outils de la qualité
- Prérequis pour utiliser cette méthode de résolution :
 - Les raisons qui ont guidé le choix du sujet sont connues, les objectifs sont clairs
 - Une analyse exhaustive est faite à l'aide d'outils de la qualité
 - L'analyse a révélé une réelle relation entre causes et effets
 - Les solutions sont formulées en faisant preuve d'imagination et d'esprit d'innovation (loul)
 - La résolution du problème est abordée du point de vue de la qualité
- Les 7 étapes :
 - 1 Choisir un sujet et délimiter précisément le problème
 - 2 Comprendre la situation, identifier les effets négatifs (par ex : coûts) et fixer les objectifs
 - 3 Planifier les activités
 - 4 Analyser les causes majeures du problème
 - 5 Rechercher les solutions appropriées, élaborer les plans d'action et mettre en oeuvre les solutions
 - 6 Vérifier les résultats et présenter les conclusions
 - 7 Définir les indicateurs de contrôle = critères de jugement du résultat (ex : vigueur pour la qualité d'une b*te)
- Résultats attendus : contribution de chacun à la résolution du pb, analyse de tous les aspects du pb, obtention du consensus sur les solutions retenues.

1.6.2 Quelques méthodes pour la démarche qualité

Démarche de l'Analyse Client-Fournisseur (ACF)

- Objectif : formaliser les relations Clients-Fournisseurs et déterminer ensemble des actions d'amélioration (qualité du service fourni, fonctionnement interne, meilleure communication entre équipes)
- Conditions de réussite : contribution de tout le personnel et travail en groupe, encadrement actif et motivé, volonté absolue d'aboutir, rigueur et sens du concret, organisation structurée.
- Résultats attendus : meilleure communication entre équipes, prise en compte des besoins réels des clients internes et externes, définition conjointe ds indicateurs de mesure.

- Méthode/démarche : en gros, idem que démarche générale en 7 étapes ci-dessus, mais en mettant l'accent sur la recherche des besoins réels du client et la relation client-fournisseur...

Processus itératif de la qualité

- Objectif : amélioration des processus en s'appuyant plus sur la résolution de pb que sur la hiérarchie et la responsabilité
- Démarche : Définir > Mesurer > Analyser > Améliorer > Contrôler
- Principe : on fait ça en boucle => (itération)

Démarche de l'analyse de la valeur

- Définition : L'analyse de la valeur est une méthode de compétitivité organisée et créative visant à la satisfaction du besoin de l'utilisateur par une démarche spécifique de conception à la fois fonctionnelle, économique et pluridisciplinaire.
- Ses principes : valeur, besoins/satisfaction client, méthodologie, collaboration.
- Ses buts : Concevoir au coût le plus faible, satisfaire parfaitement aux besoins, améliorer la qualité, apporter des solutions créatives.
- Démarche : orienté fonctionnalités plutôt que processus → Analyse besoins client > Identifier et hiérarchiser les fonctions du produit > Déterminer coût de chaque fonction > Choisir fonction (argumenter) > Valider les fn par rapport au besoin client > Documenter l'étude.
- Résultats attendus : Optimisation des fn et des coûts, adéquation entre solutions retenues et besoins clients (j'ai envie de dire : comme d'hab...).
- Notion de valeur : [service rendu/coût d'acquisition] pour l'utilisateur, ou [valeur de la fn/coût de concept-réa] pour le concepteur.

1.6.3 Quelques outils de la qualité

Outils de base de 1ère nécessité

Diagramme causes-effets

- Objectif : rechercher systématiquement les causes d'un effet, les classer, et visualiser les liens entre ces causes.
- Méthode : définir l'effet, rechercher les causes, les regrouper en familles, tracer et remplir le diag. de façon claire.

[insérer un diagramme ici pour exemple :) p25 du pdf chap1]

Diagramme de Pareto (80-20)

- Objectif : déterminer sous forme graphique l'importance relative des différentes causes pour choisir lesquelles traiter.
- Le principe de Pareto : 80% du problème sont dus à 20% des causes.
- Démarche : Lister les causes, quantifier leur importance, calculer les pourcentages et tracer le graphique des valeurs cumulées.

QQOQCP : Quoi Qui Où Quand Comment Pourquoi ?

- QQOQC sont les lignes du tableau, P est la colonne
- Objectif : représenter un plan d'action de façon claire et synthétique, pour ne rien oublier.
- TERRAIN!

Remue-ménages (lol) ou Brainstorming

- Démarche : générer un grand nombre d'idées (réflexion commune) puis les ordonner par famille ou critère puis les exploiter.
- Met en avant la créativité, l'imagination, toussa.

Outils de base de 2ème nécessité

OMQ : 7 outils de management de la qualité

- Mise en oeuvre en 3 phases : clarifier, rechercher et cibler les solutions les plus pertinentes, planifier leur mise en oeuvre
- Pour quantifier le qualitatif.
- 7 outils :
 - Diagramme des affinités ou KJ (=QUOI)

- Diagramme des relations (=POURQUOI)
- Diagramme en arbre (=COMMENT)
- Diagramme de décision (=SI ALORS)
- Diagramme en flèches (=QUAND)
- Diagramme matriciel (=LEQUEL)
- Analyse factorielle de données

Outils de base de 3ème nécessité

- en complément des 2 premiers si on arrive pas à trouver une solution
- Diagramme polaire (ou Kiviat ou Radar) : ensemble de paramètres sur 1 seul diag.
- Diagramme d'Euler : 3 cercles genre un diag de Venn quoi, pour montrer les combinaisons possibles de 3 éléments.

1.7 Approches utilisées pour engager une réflexion qualité

1.7.1 Approche socio-économique

- Basée sur changement d'attitudes des individus
- Associer performances économiques et performances sociales
- Intégrer ttes les catégories de personnel ds une approche structurée et volontariste
- Objectifs : Motiver les personnels, réduire les coûts anormaux, développer les potentiels de chacun, former l'encadrement au management.

1.7.2 Approche "Démarche de changement"

Schéma qu'on ne comprend pas =S c'est juste 'changer pour changer' visiblement

1.7.3 Approche "Élaboration d'un projet d'établissement"

Les projets des différents services de l'entreprise constituent le Projet Établissement.

1.8 Conclusion

Terrain.

Chapitre 2

La qualité totale : un objectif

2.1 Définition de la QUALITE TOTALE

Responsable : Chacun, tous

Comment améliorer : PREVENTION

Comment mesurer : Qualimetres à définir

Niveau de qualité accepté : Excellence, Zéro défaut

Définition : Ensemble de Principes et Méthodes, organisés en stratégie globale et visant à mobiliser toute l'entreprise pour obtenir une meilleure satisfaction du client au moindre coût. La **qualité totale** touche : Tous les employés, tous les services, tous les stades du projet, tous les aspects (qualité, coûts, production ...) → Exhaustive

2.2 Les Objectifs de la démarche QUALITE TOTALE

2.2.1 Objectif Réel

Compétitivité : Recherche du moindre coût

2.2.2 Objectif Manifesté

Zéro défaut

2.2.3 Objectifs Latents

- Réduire vos coûts inutiles
- Accroître votre impact commercial
- Renforcer l'efficacité des cadres
- Renforcer la motivation du personnel
- Améliorer votre organisation
- Raccourcir les délais

2.2.4 Convergence de ces Objectifs

C'est en faisant des compromis entre ces objectifs que l'on atteint, étape par étape, la **qualité totale**.

Il faut définir :

- la situation actuelle
- les priorités entre les objectifs
- les moyens (soit les objectifs intermédiaires)

2.3 Processus de mise en oeuvre de la QUALITE TOTALE

Commencer par :

- Audit → situation actuelle
- Plan d'action progressif sur 5 ans
- Ce n'est pas : Tout casser, tout chambouler

- C'est : Méthodologie amenée progressivement, en motivant le personnel

Le coût associé à la gestion de la qualité de l'entreprise dépend du DEGRÉ de MATURITÉ de l'entreprise par rapport aux principes de gestion de la QUALITÉ.

Cinq degrés de maturité :

Incertitude – pas de conscience des problèmes de qualité et non-qualité

- absence totale de perception du coût des actions de corrections

Mesure (éveil) 6 à 12 mois : conscient que le travail ne va pas dans le sens de la qualité

Sensibilisation (mise en évidence) 1 an – Actions d'évaluation de la qualité systématique

- Baisse du coût réel et meilleure évaluation de celui-ci

Formalisation (reconnaissance) 2 à 3 ans Actions de prévention

Maturité (certitude) coût perçu = coût réel = coût de prévention + coût d'évaluation

14 étapes selon Crosby

1. Engagement de l'encadrement
2. Gestion des programmes d'amélioration de la QUALITÉ
3. Mesure de la QUALITÉ
4. Coût de la QUALITÉ
5. Découverte de la QUALITÉ
6. Actions correctives n n
7. Planification du ZÉRO DEFECTS
8. Formation des cadres
9. Journées ZÉRO DEFECTS
10. Fixation des objectifs
11. Élimination des causes d'erreur
12. Reconnaissance des mérites
13. Conseil Qualité
14. Retour à la case départ (Amélioration : mise à jour du Plan d'Action qualité) → processus itératif

2.4 Introduction des principaux concepts d'un Système Qualité

2.4.1 Vue synthétique des principaux concepts

TODO : pas compris

2.4.2 La politique Qualité

- Slogan simples
- Plan d'Amélioration de la Qualité (PAQ)

2.4.3 Maîtrise et appréciation de la Qualité

Deux aspects de la qualité :

- Qualité du produit
- Qualité du processus

2.4.4 Le cycle de la qualité (ou Processus d'amélioration)

La maîtrise de la qualité se fait sur ces quatre étapes en itérations courtes :

- Définition des dispositions qualité
- Élaboration des actions Préventives
- Suivi et Évaluation des actions entreprises
- Amélioration de l'existant

2.4.5 L'assurance Qualité

L'assurance Qualité est définie comme l'ensemble des actions PRE-ETABLIES (Préventives) et SYSTEMATIQUES nécessaires pour donner la confiance appropriée (et A PRIORI) en ce qu'un produit ou service satisfera aux exigences données relatives à la Qualité (Norme NFX50 120 :1997)

Deux aspect pour l'assurance qualité

- Construction → dispositions prises
- Contrôle → mesure effectuées

2.4.6 La notion de procédure (disposition préventive)

Mis en oeuvre de l'outil **QOOQCP** (au nom imprononçable) : Qui Quand Où Quoi Comment Pourquoi
TODO : faire le tri de ce qui est important ou pas

2.4.7 Le système Qualité

Le Système Qualité est défini comme l'ensemble de la structure organisationnelle, des responsabilités, des procédures, des processus (ou procédés) et des moyens nécessaires pour mettre en oeuvre le management de la Qualité (NFX50-120)

2.4.8 Le management par la Qualité

TODO

2.5 Différents types de systèmes qualité

Conclusion TODO

Chapitre 3

Concepts de Qualité pour le Logiciel

3.1 Concepts de base liés à la qualité du logiciel

3.1.1 Définition de la qualité

Qualité des produits Répondre aux réels besoins des utilisateurs (implicites et explicites) dans un contexte économique donnée (coûts, délais)

Qualité des logiciels Répondre aux objectifs d'utilisation, de maintenance, de portage etc... Réalisation dans les délais et les budgets prévus...

Quality Of Service (QoS) Répondre aux attentes du client au niveau des perfs, des relations avec lui, réactivité en cas de pépin !

3.1.2 Qualité et satisfaction des besoins

Chaîne de traitement des besoins :

Besoins Réels → Besoins perçus → Besoins exprimés → Besoins retenus → Solutions proposées → Besoins spécifiés → Phase de conception / Réalisation → Besoin réalisé

Cycle d'affinage des besoins et de réajustement au cas où les besoins réalisés ne couvrent pas les exigences client. C'est une des plus grosses difficultés !

3.1.3 Maîtriser la qualité des besoins du client avant tout !

Définition de la satisfaction client : Interface entre la Demande Inutile / Insatisfaction client & Qualité inutile, c'est à dire des choses que le client demande, mais où il a tort de le faire (exigence à la con).

Critères de sélection :

- Besoins du Client
- Spécifications du Logiciel
- Réalisation du soft

3.2 Les non-conformités : perception de la non qualité du logiciel

Différentes expressions de la non qualité dans le logiciel :

- Indisponibilité du soft
- Correction, mise à jour de documents
- Déplacements, logistique de diffusion
- Retard de livraison
- Difficulté d'utilisation
- Abandon de l'application

3.3 Le coût de la non-qualité

3.3.1 Répartition des erreurs entre les phases

65% des erreurs sont trouvées lors de la phase de spécification, 35% lors du codage, mais ce sont celles qui sont le plus coûteuses... Et 75% des erreurs de spécifs sont trouvées avant la recette (lors du codage...) Seulement 30% des erreurs de code sont trouvées avant la recette (car on a pas fait de tests unitaires assez bon, pas assez de tests fonctionnels, ou alors ils ne sont pas assez exhaustifs, et que lors du recettage, une personne avec des yeux neufs arrive, et fait une utilisation différente du logiciel, et trouve un bug). D'où l'émergence des méthodes agiles, où le recettage est fait tout au long du produit. Citation de Linus Torvalds : « *Given enough eyeballs, all the bugs are shallow* » : « *Si on montre le logiciel/code à assez de monde, tous les bugs surgissent* ».

Plus on trouve un bug tard, plus c'est cher, dit et répété pendant 6 slides !

3.3.2 Relation entre le coût et sa qualité

Graphe intéressant (slide 13) mettant en évidence deux choses :

- Il existe un point d'équilibre entre réponse aux besoins et la surqualité, ou le coût global est minimal. C'est une courbe en x^2
- À chaque coût est associé deux points, un point en perfectionnement et un point en surqualité.

3.4 Émergence de la fonction qualité dans le développement de logiciel

3.4.1 Position de la fonction qualité dans le logiciel

« Le développement de logiciel est un système complexe soumis à un faisceau de motivations et de contraintes »

La qualité n'est pas une fin en soi, il faut trouver des compromis sur plusieurs plans (méthodes, technique...). Ces compromis sont déterminés de façon rigoureuse, de façon à obtenir l'optimum.

Le contexte à évolué, avant c'était le savoir faire qui avait la prédominance, du coup des coûts incalculables et la fiabilité tombaient au profit de l'exploit technique. Maintenant nous sommes rentrés dans un contexte de compétitivité financière ou l'on doit avoir une forte maîtrise des process de façon à rester rentable !

3.4.2 Parallèle avec l'industrie

Premier niveau Minimiser le nombre de rebut / logiciel le plus fiable possible

Second niveau Optimiser le process de fabrication / Optimiser le processus de création (introduction de méthodes USDP, agiles...)

Troisième niveau Définition de best practices, valable dans le logiciel

3.5 Qualité attendue par les différents acteurs

Compromis entre les différents acteurs, MOA, MOE et utilisateurs

MOA

- Respect des objectifs stratégiques (coûts/délais/performance)
- Maîtrise du process d'évolution du logiciel
- L'efficacité du développement et de la maintenance
- L'auditabilité des application

Dans le processus qualité, la MOA est en charge, de l'affectation des responsabilités, établir et répartir les charges nécessaires à l'obtention du niveau de qualité souhaité. Coordination, gestion process qualité et gestion projet

Utilisateurs

- Conformité fonctionnelle
- Valeur pédagogique
- Valeur technique

MOE (Maîtrise d'oeuvre)

- Choisir un process de développement approprié
- Proposer et maîtriser une démarche qualité
- Suivi du projet
- Prévoir la maintenance, l'évolutivité
- Efficacité du développement
- Respect des procédures
- Pertinence de la documentation

3.6 L'approche management par la qualité pour le logiciel

Avant toute chose, 95% du coût total d'un projet est engagé alors que 20% de ce coût total n'est dépensé.

Les grandes disciplines du management par la qualité :

- Gestion de Projet
- Production
- Assurance qualité Logicielle

La qualité totale a comme rôle d'équilibrer le système de développement, en fonction d'un optimum coût / qualité.

CS Qualité de service

CN Qualité du produit

3.7 Conclusion

Cycle de qualité lourd et long à mettre en place, on compte 1 an pour passer la phase d'éveil (prise de conscience des problèmes de qualité) à la phase de sensibilisation (prise de conscience par chacun et début d'application de best practices), puis 3 ans de formalisation (système qualité, normes, formation interne, moyens investis...), pour enfin arriver à maturité → Certification.

Chapitre 4

Outils pour la résolution de problèmes et la recherche de la qualité

4.1 Introduction : Qu'est-ce qu'un problème ?

Approche déductive : Utilisée pour résoudre des problèmes en appliquant des principes physique, chimiques etc... recherche fondamentale.

Approche inductive Résolution d'un problème par l'approche qualité :

- Approche et amélioration continue
- Approche rationnelle

4.2 Quelles démarches pour la qualité ?

- Démarche de l'Analyse Client-Fournisseurs (A.C.F)
- Formaliser les relations client-fournisseur et déterminer un ensemble d'actions pour uniformiser et péreniser cette relation
- 10 Phases...
- Maîtrise des outils et des compétences de l'équipe !
- Le cycle d'amélioration des processus ou process itératifs de la qualité
- Idem que précédemment en focalisant l'approche sur les process uniquement !
- Démarche de l'Analyse de la valeur
- AFNOR : Analyse de la méthode visant à la satisfaction du besoin utilisateur par une démarche spécifique de conception.
- Faire l'accord entre les coûts et la démarche qualité, l'importance est de bien déterminer la valeur de vente, puis la valeur de fabrication. Adapter le produit aux exigences...
- En informatique : On l'applique au niveau du cahier des charges fonctionnel
- Il faut ordonnancer par importance les fonctions de base, et investir au maximum sur les exigences importantes, par rapport à ce qui est réellement nécessaire. Quitte à mettre entre parenthèses voire supprimer certaines exigences moins prioritaires.
- Notion de coût : Traquer les fonctions inutiles, éliminer la surqualité ou l'exploit technique (on s'en fout de la technique #trollinsa),

TODO Terminer, j'ai du m'endormir à ce moment la :-D

Chapitre 5

La norme ISO - 9000

La qualité repose sur la capacité des fournisseurs à maîtriser la qualité, il est donc nécessaire pour les clients de pouvoir évaluer cette qualité. et cela se fait par une grille d'évaluation des systèmes de qualité du fournisseur. Pour que ce système de qualité soit pris au sérieux on lui attribue un nom dégueulasse qui se base sur des nombres à quatre chiffres au détriment de la poésie, de l'amour et de la nature.

Les **objectifs** de la norme sont de formaliser le système de qualité, de le documenter et de permettre des actions correctives. il s'agit d'une norme orientée **processus** et non orientée produit. On ne cherche pas à garantir la qualité, mais bien **l'application d'un système de qualité**.

La norme ISO-900 apporte une démonstration de **savoir-faire**, un gain en **image** et en **confiance** et privilégie les phases créatrices par rapport aux phases de test et de maintenance en aidant à faire un **meilleur travail**. En interne la norme aide aussi à assurer la pérennité du savoir-faire (**capitalisation**) et le management/contrôle financier. La norme recherche **l'efficience** (l'efficacité au juste coût).

5.1 Les 4 verbes de la norme

Plan Préparer planifier, prévoir

Do Réaliser, faire, produire

Check contrôler, vérifier, prouver

Act Analyser, améliorer, progresser

5.2 Approche processus

Définition C'est un ensemble d'activités corrélées ou interactives transformant des éléments d'entrée en éléments de sortie.

Description d'un processus Carte d'identité, carnet de santé, moyens de maîtrise.

→ Se reporter au schéma du poly !

indicateur de conformité service réalisé / service voulu.

indicateur de satisfaction service perçu / service attendu.

Le **pilote de processus** (responsable) a pour rôle de collecter les indicateurs, rendre compte de l'efficacité du processus et proposer à la direction des actions d'amélioration du processus.

5.2.1 Exigences de la norme

Les éléments du système de qualité doivent être documentés et démontrables d'une manière compatible avec les exigences du modèle sélectionné.

5.2.2 Documents exigés par la norme

- Politique et objectifs qualité
- Manuel qualité (périmètre d'application, description des interactions des processus, liste des procédures)
- Procédures obligatoires
- Documents nécessaires pour la planification, le fonctionnement et la maîtrise des processus
- enregistrements

5.2.3 Enregistrements exigés par la norme

On garde **TOUT** (la liste est trop longue pour la présenter ici) ce qui concerne de près ou de loin les analyses, actions, compte-rendus, incidents, évaluations, résultats, validation, identification, étalonnage, conformité, formation des employés, etc.

- Sur les processus pilotage & support
- Sur les processus métiers

5.2.4 Plan d'assurance qualité

Présentation (Fiche prestation, contrat, clauses qualité....)

Analyse des objectifs de qualité, et des risques « recherche du compromis optimum »

Référentiel documentaire (documents du client, système qualité, normes..)

Organisation (rôles et responsabilités des personnes)

Déroulement (méthodes, règles, livrables, planning...)

Configuration (gestion des matériels, logiciels, environnements, versions..)

Problèmes courants	Solution ISO 9000
Erreur de management	Responsabilités définies clairement
Manque de communication	Communication, coopération encouragées, points (réunions) planifiées
Incohérence	Standards, gestion de la cohérence
Produit "Buggé" (non qualité)	Revue, Vérification, Validation
Erreurs qui se reproduisent	Actions correctives (en principes diminution)
Peu ou pas de réflexes préventifs	Actions Préventives
CLIENTS MECONTENTES	QUALITE et SERVICE
Faible maîtrise	Planification et Gestion
Charge de travail	Café

5.3 Critiques de la norme ISO-9000

Critique	Contre-Critique
La norme ne donne pas de détail sur le niveau de maîtrise des activités requis	heu, ouais.
"Trop de rigorisme" "manque de réactivité"	Faute de l'expert qualité : rigueur != rigorisme
Culte de l'écrit	Ok pour la formalisation, non pour la pape-rasserie
Normes pour l'industrie manufacturière	Les adaptations ont été faite (Extension aux services)
Aucune prescription (ni méthodes, ni outils ni pratiques de base)	Permet de faire rentrer vos pratiques

5.4 Certification du projet qualité

- Déposer une demande à un organisme habilité
- Audit par l'organisme habilité sur le site
 - Remise d'un rapport précisant ce qui va, ce qui ne va pas
- Maintenir le système de qualité
 - Audit suivi une fois par an
 - Audit de renouvellement une fois tous les trois ans

La certification n'est qu'un point d'ancrage et ne doit pas être une fin en soi, **le but c'est la qualité.**

Chapitre 6

Mise en Oeuvre d'un système qualité pour la production de logiciel

6.1 Cadre général pour la mise en oeuvre d'une démarche qualité pour la production de logiciel

L'assurance qualité est chargée d'appliquer à l'échelle d'un produit les directives / standards pré-établis d'un système qualité. Ceci avant (préparation), pendant (réalisation) et après (maintenance). Il est évident que l'AQ peut définir au fil des projets de nouveaux standards visant à améliorer le SQ, mais attention aux faux best practices !

Trois grandes entités du système qualité logiciel :

- **La Gestion de configuration** : Assure la visibilité permanente d'une version du soft ainsi que des composants satellites (documentation, modules)
- **Vérification & Validation** : S'assure de la conformité par rapport aux besoins à tout niveau du dev.
- **Tests & Evaluation** : S'assure de la recette au niveau client, et de la maintenance derrière pour y apporter quelque chose.

Le système qualité modélise et évalue le système de production logiciel en continu. Ses objectifs sont de mettre en oeuvre une démarche qualité au niveau Entreprise et au niveau Projet, en gardant toujours un soucis d'optimisation constant. Chaque projet est unique, et nécessite son propre PAQL !

Zone d'action d'un Système qualité : – PdV Organisationnel : Mise sous contrôle de l'organisation de l'entreprise (ensemble de processus),

- **PdV Démarche** : Approche progressive et évolutive de la démarche, allant de l'expression de la politique qualité, sa mise en place, son entrée dans les moeurs et enfin son évaluation.

Objectifs d'un SQ : – Prise en compte du contexte multi-partenaires et les préoccupations qualité des différents métiers

- Minimiser le risque de dérive des projets Facilier le retour d'expérience
- Maîtriser les couts / délais, harmoniser le corps et les normes des procédures
- Une visibilité totale du process de production

6.2 La structure documentaire d'un système qualité

Comment se décompose un SQ : – Structure organisationnelle

- Responsabilités
- Processus de développement
- Procédures
- Ressources

Avec bien entendu la direction au dessus !

Architecture documentaire : – Politique Qualité de l'Entreprise : Manuel Qualité / Manuel Assurance Qualité

- Procédures : Instructions / Modes opératoires

- Projet : Documents (contrats, PAQL, PDL...)

6.2.1 Documentation de niveau 1

Manuel Qualité : énonce et détaille la Politique qualité applicable au niveau de l'entreprise ou d'un sous ensemble sectoriel important. Pose les standards de l'entreprise, définition des Best Practices.

“Avec mon MAQ (MAnuel Qualité) je peux frimer à l'extérieur, et ça se voit à l'intérieur !”

6.2.2 Documentation de niveau 2

Manuel de Mise en Oeuvre Opérationnelle : énonce les Règles d'ingénierie, outils utilisés, procédures, documentations liées. Cela cimente le respect des standards et la réutilisation des bests practices, c'est la qualité par construction.

6.2.3 Documentation de niveau 3

Documentation des projets informatiques :

5 grands types :

1. Documents de relations Contractuels (CdC, Proposition...)
2. Document de gestion projet (Initialisation, suivi, bilan...)
3. Document d'assurance qualité (PAQL, Audit interne, evaluation en fin de phase, plans de tests) **Disciplines du Support**
4. Document d'études et de développement (Étude préalable, détaillée, dev..)
5. Documents d'utilisation (manuel),
6. Enregistrements qualité (preuves de la mise en oeuvre des règles qualité pendant le projet) → comptes-rendus de tests, etc

Quelques dossiers importants :

Dossier d'init Description de la mise en oeuvre des ressources pour développer et produire le produit logiciel en respectant les contraintes. Évalue les risques, l'organisation du projet, le planing prévisionnel, les moyens mis en oeuvre, documentation de suivi...

PAQL Modes opératoires, des ressources, des séquences d'activités liés a la qualité. Permet de s'assurer de la mise en oeuvre et de l'efficacité des activités prévues pour obtenir la qualité requise.

IL FAUT DONNER CONFIANCE AU CLIENT !

6.3 1ere discipline du support : l'assurance qualité

6.3.1 Rôle de l'assurance qualité

“Ensemble de dispositions préétablies et systématiques nécessaires pour donner la confiance appropriée en ce qu'un produit ou service satisfera aux exigences données, relatives à la qualité”

6.3.2 AQ au niveau de l'entreprise

Rédaction du Système qualité :

Expérience et culture maison + Contexte + Politique de l'entreprises + Etat de l'art → CHOIX → MAQ, MAQ.service, spécifications et conceptions
(Recherche des Best Practices)

Chaque service d'une entreprise doit rédiger son propre MAQ (MAQL si service informatique) héritant du MAQ de la direction générale.

(Exemple sur un cycle en Y et sur un cycle en V de la diapo 25 à 27)

Exemple sur la production du code : MAQL :

- Rédaction et mise en place des procédures

- Définition de Normes de conception et programmation
- Rédaction de plan type, règles de productions
- Choix d’une organisation, des méthodes et des outils
(Exemple niveau guide de style diapo 29)

6.3.3 AQ au niveau du projet

Objectif du génie logiciel Faire correctement le bon produit dans des conditions satisfaisantes

Objectif de l’Assurance Qualité Logiciel (AQL) Bien faire le produit (bonnes règles de prod et bon procédé)

Contrôle Qualité Logiciel (CQL) vérifier le produit

Objectif de la Gestion de projet Logiciel (GdP) maîtriser le procédé

RQ doit s’assurer que les logiciels et processus utilisés sont conformes aux exigences du CdC et respectent les plans établis.

Le PAQL doit répondre à :

Qui est responsable de quoi ?

Quels sont les engagements des différents partenaires envers la MOA ?

Quels sont les Objectifs Qualité du projet ? du produit ?

Quels sont les circuits d’information, règles de production et de gestion de la documentation ?

Quels sont les méthodes et outils ?

Portée d’un PAQL : totale.

Paramètres influençant le PAQL :

les exigences du client (cf CdC, éventuellement référentiel d’exigences)

les difficultés à résoudre le problème

équipe de développement

environnement de développement

stabilité du système

Procédure d’élaboration :

Besoins → Classification des exigences Qualité → Moyen pour assurer les qualités requises → PAQL

6.3.4 Synthèse

6.4 Environnement organisationnel de la qualité dans un projet

Equivalent entre le déroulement du projet et la mise en place du PAQL :

Projet	Avant vente	Lancement du projet	Processus de projet	Cloture du Projet
PAQL	Définition, Rédaction initiale	Rédaction initiale, validation	MAJ Rédaction, validation, mise en oeuvre	Bilan qualité

6.5 Suivi de la démarche qualité

Définir des vérifications pour contrôler la conformité, aussi bien sur la forme (structuration des documents, programmes) que sur le fond (conception, réalisation), de l’évolution des processus et des produits par rapport au PAQL → rédaction d’un chapitre sur le contrôle de la qualité (cf Chap 12 du PAQL).

6.6 Conclusion

- Démarche qualité à tous les niveaux (Entreprise, processus, projet/produit)
- Suivi de la qualité constant
- Indépendance des documents Qualité

Prévoir pour mieux Agir, Contrôler pour mieux Réagir.

Système Qualité :

- **MQ** : Manuel Qualité de l'entreprise
- **MAQ** : Manuel d'Assurance Qualité
- **MAQL** : Manuel d'Assurance Qualité Logiciel
- **PAQL** : Plan d'Assurance Qualité Logiciel
- **PDL** : Plan de Développement de Logiciel
- phase de Construction de la Qualité
- Assurance Qualité
- Évaluation (Vérification ou Contrôle)
- Évaluation de la Qualité (META-Qualité)

La qualité de construit progressivement

Règles d'or de l'AQL :

- Découper le processus de développement
- Concentrer ses efforts sur les points critiques
- Définir les responsabilités entre partenaires
- Structurer la documentation
- Vérifier la qualité en cours de développement
- Mesurer l'avancement et le comparer avec les prévisions
- Définir des standards, normes, conventions et best-practices
- Mettre en oeuvre des outils de fabrication
- Disposer de règles pour maintenir la cohérence du logiciel

Chapitre 7

Techniques d'amélioration de la qualité

7.1 Avant-propos

3 domaines de Management par la Qualité : – GdC (Gestion de Configuration),

- Support
- Doing (Réalisation)

4 disciplines du support : – AQ (chap 6),

- Vérification et Validation (V&V),
- Gestion de Configuration (GdC),
- Test et Évaluation Indépendante (T&E)

V&V : assure la conformité par rapport aux exigences (tous niveaux du développement)

GdC : assure la visibilité permanente de la production du logiciel (suivi des documents, modules..)

T&E : assure la recette client d'un système indépendamment de la production

7.2 2ème discipline du support : Vérification & Validation (V&V)

7.2.1 Introduction aux techniques de contrôle

contrôle : à partir d'informations on prend une décision, qui débouche sur des actions correctrices. à la fois validation (bon produit) et vérification (le produit est fait correctement), évaluation du produit (validation) et du processus (vérification)

3 règles d'or :

- **Définir les vérifications :** fixer des référentiels + liste des vérifications
- Mettre en face de chaque vérification les procédures associées : on fixe le contrôle (quoi, qui, comment, quand...)
- Spécifier les outils pour :
 - aider à réaliser les contrôles (aide à la saisie, recherche automatisée...)
 - gérer les contrôles (confidentialité, enregistrement, traçabilité...)
 - exploiter les résultats

7.2.2 les contrôles qualité (ou formels)

- outils d'évaluation de la qualité
- contrôler "ce qui est nécessaire" par rapport aux clauses qualité et la nature du produit
- pourquoi contrôler : vérifier les écarts entre les dispositions d'assurance qualité définies dans le PAQL et la réalité

7.2.3 les contrôles techniques (sur le fond)

objectifs : s'assurer tout au long du processus de fabrication :

- de la cohérence entre livrables de l'étape n et (actuelle) et résultats de n-1
- description des interfaces
- conformité vis à vis exigences fonctionnelles et non fonctionnelles

On fait à la fois la validation (bon produit) et la vérification (faire le produit correctement).

Justification des contrôles techniques : plus on voit une erreur tôt moins ça coûte!

On met en place des contrôles tout au long du cycle de développement (définition, conception, réalisation et exploitation).

7.2.4 contrôle de situation (audit)

objectif : faire point sur situation technique

quand : à la demande d'un client

qui : personnes indépendantes du projet

quoi : organisation de l'entreprise, état d'avancement du logiciel, processus de conduite de projet

comment : identifier écarts entre situation réelle et prévue (détail des différents types d'audits suivant type de référentiel slide 15)

7.2.5 mise en oeuvre des contrôles de qualité

(3 slides, 3 slides barrés 16 à 18)

Présentation d'un exemple : cycle de vie d'un système, positionnement des revues et modèle d'organisation des équipes.

7.3 3ème discipline du support : Gestion de configuration (GdC)

(configuration = version)

7.3.1 intro à la GdC

configuration : définie par une liste cohérente de composants dans un certain état. Associée à un certain type d'acteurs (utilisateur, développeur,...)

changement de version : évolution majeure du logiciel

révision : évolution mineure

que doit-on gérer : schéma (illisible) slide 21, interprété ainsi : on gère le lien entre les différentes versions temporelles et les fichiers (description statique)

moyens : PAQL, chaîne de production, Dossier de suivi de configuration (DSC), plan de test et environnement de test

gestion de différents espaces : espace de référence contenant la dernière version cohérente

espace de développement : pour le développement!

espace d'archivage : contient l'ensemble des configurations de référence

article de configuration : plus petites entités gérées dans la GdC, plus petites entités pouvant passer d'un espace à un autre

7.3.2 apports de la GdC

Chacun (chef de projet, resp qualité, équipe de dvpt, client) y trouve son compte (slide 26 pose 4 questions que chacun se pose).

GdC consiste à :

- gérer l'ensemble des composants (documents, logiciels d'application, logiciels de base, fiches produites pendant dvpt, matériel)
- prendre une assurance contre le désordre en assurant la cohérence de ce qui lui est confié
- présenter de manière claire et complète la configuration instantanée du produit et l'état d'accomplissement des spécifs fonctionnelles et techniques (doc, source, exécutables, ...)

7.3.3 champs d'application de la GdC : configurations de référence (ou référentiels)

Tous les articles sont en cohérence fonctionnelle et technique.

Un article de configuration est représenté par :

- ses composants,
- les liens entre ces composants,
- ses interfaces dans le cadre d'ensembles plus larges.

Chaque intervenant doit pouvoir manipuler un ensemble cohérent sans perturber les autres : on définit donc plusieurs types de configuration.

Configuration :

- de spécification,
- de tests,
- de développement
- de référence : liste des composants approuvés qui servent pour des phases ultérieures du projet (les autres configurations).

7.3.4 comment organiser une GdC

Démarche :

- Analyse du besoin des différents intervenants (espaces nécessaires, activités de gestion de configuration et leur fréquence, confidentialité, ..)
- Enumération des types d'articles de configuration et de leurs liens
- Mise en place de l'organisation
- Mise en place des moyens
- Identification des articles de configuration pour chaque type retenu
- mise en oeuvre

Attributs d'un article de configuration :

- identification,
- auteur,
- date de création et de dernière modification,
- état (encours, gelé, ..),
- droits d'accès,
- support
- ...

Articles groupés en configurations (ensembles complets et cohérents d'articles logiquement reliés) : ces liens servent à analyser impact d'une modification et assurer la cohérence des liens après modification.

À une configuration on associe un arbre de configuration : le **noeud** est un objet de type agrégat, la **feuille** représente un fichier avec des liens de composition, génération, référence, applicabilité, instanciation, traçabilité (exemples slide 34 à 36).

4 axes :

Identifier : chaque configuration → description technique des articles à chaque instant de leur cycle de vie

Maitriser : évolution de chaque configuration

Enregistrer : prise en compte de l'état de chaque configuration

Contrôler : audits de configuration fonctionnels (ACF) et physiques (ACP)

Deux actions peuvent déclencher une gestion des modifications :

- détection d'une anomalie (DA) qui génère un rapport d'anomalie (RA)
- demande d'évolution demandée par le client (DE)

7.3.5 interaction entre la GdC et la production de logiciel

GdC : états stables et référencés du logiciel, gestion de la configuration de référence d'un logiciel et ses évolutions.

Production de logiciel : structuration et maitrise de la construction du logiciel, phase de construction du logiciel.

On voit 2 approches différentes, mais un objectif commun : faciliter la vie et obtenir une meilleure qualité.

7.3.6 Conclusion

C'est une activité de soutien de projet, qui doit être adaptée au cas de chaque projet.

coûts : définition, mise en place et suivi,

gains : satisfaction du client, homogénéité des développements, organisation de la production, qualité des produits, performance

7.4 4ème discipline du support : Test & Évaluation Indépendante (T&E)

7.4.1 objectif de T&E

objectif : exprimer en phase d'intégration système un point de vue différent et indépendant des équipes système et développement de logiciel,

règle : la prise en compte des évolutions implique une reprise du processus de développement qui doit se situer le plus en amont possible

7.4.2 principales livraisons

→ slide 51

7.4.3 avis de livraison à l'Intégration & Validation

plan type pour un avis de livraison :

identification du logiciel : – éléments associés

- RA corrigés et non corrigés
- DM prises en compte

contenu de la livraison : – les PROM

- sous produits logiciel
- sous systèmes

contenu du logiciel livré : – fonctions testées

- fonctions non testées
- restrictions
- infos sur le logiciel livré

procédure d'installation du logiciel

procédure de lancement du logiciel

annexes : – référentiel prévisionnel

- fichiers de démarrage
- utilisation particulière

7.5 Conclusion

Toutes ces activités doivent être mises en oeuvre pour obtenir un produit de qualité. Elles ne sont pas indépendantes. Il faut les réduire au maximum → et donc réduire les coûts.

Chapitre 8

Gestion de la Documentation

8.1 Introduction

Objectif : Comment gérer la documentation générée par une approche qualité.

Besoins de base :

- Connaissance de ISO 9000
- Exigences de CMMI en matière de documentation (Capability Maturity Model Integration)

Dans toute la suite, on verra défiler les plans des principaux documents nécessaires à la gestion documentaire. Il est donc conseillé d'imprimer cette partie plus que de l'apprendre.

8.2 Plan type de la documentation

Sommaire

1. Contexte
2. Objet
3. Documents applicables et de référence
4. Principaux vocabulaires
5. Formalisation des documents produits
 - (a) Processus de création d'un nouveau document
 - (b) Présentation des documents
 - i. " Draft "
 - ii. Livrables
6. Structuration des documents
 - (a) Informations de la page de garde
 - (b) Informations de pied de page
 - (c) Informations au coeur du document
7. Gestion des documents produits
 - (a) Etat des documents et gestion des modifications.
 - (b) Vérification/Validation.
 - (c) Gestion des versions
 - (d) Gestion des sauvegardes
8. Classement des documents produits
 - (a) Journal de la documentation
 - (b) Nomenclatures et références
 - (c) Gestion des répertoires
9. Gestion de la documentation fournie
 - (a) Documentation informatique
 - (b) Documentation papier
10. Gestion du glossaire

- (a) Définition du glossaire
 - (b) Acteurs
 - (c) Processus de remplissage
11. Plans Type → Points importants à définir tôt
 12. Annexes

8.3 Résumé des Documents nécessaires en fonctions de la taille des projets

x	Petits projets	Moyens Projets	Fat Projets
Dossier d'init	OUI	OUI	OUI
PAQL	NON	OUI	OUI
Plan de vérification	NON	OUI	OUI
Plan de gestion de configuration	NON	OUI (Si exigence de maintenabilité)	OUI

NB : se référer au chap5 pour le détails de ces documents

« on n'est pas sûr que l'on fera de la qualité mais, on fera au moins de l'assurance »

Chapitre 9

Techniques de Controle : Reviews

9.1 Revues Techniques

Objectifs : – détecter des défauts

- améliorer la communication
- améliorer l’homogénéité du projet
- avoir un rôle d’éducation

Types de revues – revue technique (aka revue structurée)

- Informelle
 - Cycle Auteur/lecteur (aka Lecture croisée ou peer review)
- Formelle
 - Revue structurée
 - Inspection (détails dans la suite)
- revue de projet
 - Complète
 - Simplifiée
- audit

9.2 Inspections ou Walk-Throughs

Idée principale : Etape de validation de ce qui a été fait. Impossible de passer cette étape si non respect des exigences posées.

Le schéma ci dessous représente comment doit se faire la validation d’une inspection. Si inspection échoue, on itère tant qu’on ne la valide pas.

Dans ce schéma, on se positionne à l’étape de conception fonctionnelle. Cela dit, la méthode reste valable pour toutes les autres étape de développement d’un logiciel.

Définitions des objets utilisés dans le schéma :

Liste de contrôle : liste de tous les points où il pourrait y avoir accrochage : Toutes les constantes sont elles bien définies ?

Critères de Sortie : Liste de critères pouvant faire que l’inspection soit considérée comme un échec

- non prise en compte d’une spécification
- non prise en compte d’une contrainte

Critères d’entrée (non représentés) Ces critères permettent de déterminer si le projet est dans un état tel qu’une inspection est envisageable.

9.3 Comment organiser les réunion de revues structurée

x	Préparation	Réunion	Suivi
Durée approximative	5 jours	2 heures	5 jours
Activité de l'auteur	Fournir le document	Présenter le document, Détailler les points posant problème	Corriger le document
Activités du président	Designer les participants à la reunion	Diriger les débats	S'assurer que les corrections sont effectuées
Activité des participants (sauf auteur)	Lire de Document, Corriger les erreur mineures, Définir les points posant problème	Se mettre d'accord sur les points à corriger	

9.4 Règles de bonnes pratiques des Inspections

- distribuer, à l'avance l'information nécessaire (documents, programmes).
- 3 personnes minimum, 7 personnes maximum
- Dans un contexte de grands projets ou projets à risque, au moins 3 personnes ne doivent pas prendre part au développement de cette partie/module/composants etc. . .
- Pas plus de 2 heures de réunion
- Inspection != conception, on cherche les erreurs et comment les corriger, on ne reconçoit pas l'intégralité de l'application
- Effectuer un suivi
- Ne pas juger l'auteur du document.

Chapitre 10

Approche Produit : « Modélisation de la qualité d'un produit logiciel »

10.1 Introduction

La qualité peut être vue de deux manières :

- Par une **approche produit** : la qualité est une propriété intrinsèque du produit logiciel
- Par une **approche processus** : la qualité a pour objectif de contrôler la manière et le cadre dans lequel le logiciel est développé : cela a pour conséquence de produire un logiciel de meilleure qualité.

À la base on peut agir sur **deux facteurs** : les **exigences non fonctionnelles** (contraintes sur le développement), qui vont déterminer la fiabilité, la maintenabilité, l'efficacité, mais aussi sur les **exigences fonctionnelles** : qualité de l'analyse, choix de conception... Ce dernier facteur va déterminer la complétude, la clarté, et la cohérence du produit logiciel.

La modélisation de la qualité se fait de deux points de vue : (Approche de McCall)

Qualité externe : du point de vue du client – Le produit est conforme à ses attentes.

Qualité interne : du point de vue du concepteur – La conception et la réalisation s'est fait dans le respect des exigences fonctionnelles et non fonctionnelles et développées dans des conditions économiques acceptables.

Souvent, une bonne qualité interne amène une bonne qualité externe : si le produit est bien conçu et respecte les exigences du client, alors le produit va être conforme aux attentes du client. Des mesures et des contrôles sont utilisées pour mesurer l'écart entre les exigences client (fonc. et non fonc.). Une négociation entre les différentes parties (MOA, MOE) est envisagée si les contrôles et les mesures ne sont pas satisfaisantes, donc si la qualité interne est trop basse, donc que la qualité externe fait défaut.

Cette approche permet :

- En début de projet, de définir une politique de qualité
- Par la suite, d'évaluer le niveau de qualité atteint

Concept d'arbre de qualité. Il doit souvent être personnalisé pour être efficace. Cela se fait en fonction du type de produit, et du métier de l'entreprise.

10.2 2e partie : 1er niveau de décomposition : Les facteurs

Pyramide de décomposition des niveaux de qualité :

Les usages du logiciels doivent être rassemblés selon 3 groupes :

Le fonctionnement : vue de l'utilisateur – Conformité – besoin clients

- Fiabilité – rendu du service souhaité avec la précision requise
- Efficacité
- Intégrité – Protection des données contre les attaques
- Facilité d'emploi

- La transposition (portabilité, réutilisation, couplabilité...)** – Portabilité : pouvoir changer le logiciel d’environnement matériel ou logiciel facilement
- Réutilisation : pouvoir réutiliser du code pour un autre logiciel
 - Couplabilité : pouvoir mettre ce logiciel en communication avec d’autres logiciels.
- L’évolution** – Maintenabilité : dépannage facile.
- Flexibilité : capacité à évoluer facilement
 - Testabilité : capacité à pouvoir facilement tester le logiciel.

Certains facteurs sont en opposition (par exemple, si l’on veut un logiciel plus efficace, bien souvent, le code est plus compliqué, et donc la maintenabilité en prend un coup).

En général, on classe le niveau d’exigence de A à D, avec D le plus faible et A le plus critique, et selon des domaines d’application (sûreté, économie, sécurité, environnement).

Exemple : Niveau d’exigence B

Sûreté	Attente à la vie des personnes
Economie	Perte économiques
Sécurité	Service et informations critiques
Environnement	Nuisance de l’environnement récupérable

Pour déterminer le niveau d’exigence, on utilise un processus en trois étapes :

On formule les besoins : première liste de facteurs. Puis on prend les plus importants, et on leur associe un niveau d’exigence (de priorité). Enfin, on choisit les principales dispositions d’assurance qualité et de contrôle qualité, selon les bonnes pratiques.

Cela débouche sur le Plan d’assurance qualité logiciel (PAQL).

Cela se fait avec le client, et devient en fait les clauses qualité dans le contrat. On a des facteurs, donc on se réfère au cours d’Aide à la Décision, pour faire le choix (en général, Aubry dit d’utiliser une moyenne pondérée, ou de regarder les contradictions entre les facteurs).

10.3 Deuxième partie : 2e niveau de décomposition : les critères

Il y a 23 propriétés d’un logiciel, pour lesquelles le génie logiciel et l’assurance qualité offrent un moyen de contrôle.

- Fonctionnement
- Facilité d’emploi
- Facilité d’apprentissage** : le logiciel est simple à prendre en main pour un nouvel utilisateur (aide en ligne)
- Facilité d’utilisation** : le logiciel précise clairement comment il doit être utilisé (et a de la doc).
- Communicabilité** : le logiciel a une IHM pratique, aisée à comprendre (norme d’interface, étude d’ergonomie).
- Intégrité
- Contrôle des accès** : On peut restreindre certaines fonctionnalités à certains types d’utilisateurs.
- Vérification des accès** : on peut suivre l’accès aux données (date de dernière modif, etc.).
- Efficacité
- Efficacité de stockage** : le logiciel utilise le minimum de mémoire pour son exécution
- Efficacité d’exécution** : Le logiciel utilise un minimum de temps machine pour son exécution.
- Conformité
- Traçabilité** : On peut tracer les modifications dans les données dans tout le logiciel.
- Complétude** : Le logiciel remplit la totalité des fonctions pour lequel il a été conçu.
- Fiabilité
- Précision** : précision exigée dans le calcul (demande de vérification par un expert, analyse numérique).
- Tolérance aux erreurs** : Le logiciel peut continuer à fonctionner après une erreur (traitement des exceptions, etc.)

Homogénéité : Vocabulaire uniforme, etc., dans le but d'augmenter la prédictabilité (au niveau documentation, interface, etc.).

Simplicité : les choix de réalisation sont faciles à comprendre.

Concision : Il n'y a pas trop de code inutilement présent, trop de blabla dans les docs, etc.

– Évolution

– Maintenabilité

Auto-description : le logiciel décrit son propre fonctionnement.

Modularité : on peut interchanger des blocs logiciels, ça fonctionne toujours.

Extensibilité : on peut ajouter de fonctionnalités facilement, redimensionner la capacité de stockage, etc.

– Testabilité

Instrumentation : On peut mesurer les temps d'exécution, les erreurs survenues, etc.

– Flexibilité

Indépendance matérielle : on peut faire tourner le logiciel sur un grand nombre de matériel différents.

– Transposition

– Réutilisabilité

Généralité : Capacité à conserver le code en ajoutant plein de fonctionnalité, à rester ouvert au changement.

– Portabilité

Indépendance matérielle, logicielle : pas trop de dépendances à des bibliothèques externes, à un système d'exploitation.

– Couplabilité

Standardisation des communications : Utilisation de protocoles standards (HTTP, DBUS, CORBA, etc.).

Standardisation des données : Utilisation de standards normalisés pour le stockage des données (SQL, XML, YML, STL, etc.).

10.4 Troisième partie : 3e niveau de décomposition : les métriques

Les métriques et indicateurs servent à mettre des chiffres derrière les critères, car « *Sans mesure, pas de contrôle, sans contrôle, pas de qualité* ».

On utilise donc souvent une approche dite quantitative pour ce contrôler l'état du projet par rapport à ces objectifs.

On a donc quatre objectifs principaux pour la qualité :

1. Évaluer le niveau de qualité atteint
2. Répartir l'effort de test
3. Estimer l'effort de maintenance et de difficulté d'un transfert
4. Détecter les séquences à haut risque

Trois grandes approches de l'approche quantitative :

1. L'analyse statique : sans lancer le programme (détection de fautes de styles, comptage des commentaires, graphes d'appel, complexité cyclomatique, en gros ce que fait Kalistik). Exemple de métrique : nombre d'arc du graphe d'appel, nombre de noeuds, nombre de niveau, complexité hiérarchique (nombre de composants / nombre de niveaux).
2. L'analyse dynamique : pendant que le programme tourne (profiling du code, détection de fuites de mémoire, en gros, ce que fait Valgrind). On prend aussi en compte dans cette approche l'analyse de modèle de fiabilité (mean time to failure, par exemple).
3. L'analyse formelle : beaucoup plus de maths, calcul manuel de deadlines pour le temps réel, par exemple (beaucoup plus cher, et à réserver aux systèmes critiques).

On peut faire trois types de mesures :

1. Des mesures absolues
2. Des mesures relatives
3. Des mesures booléennes

On définit des bornes inférieures et supérieures à ne pas dépasser pour les mesures absolues et relatives.

Quatre sources de mesures pour l'approche quantitative :

1. Dossier de spécif, de conception préliminaire, etc.
2. Jeux de tests
3. Mesure de performances, d'exploitation
4. Règles, normes de production

On a exactement 11 facteurs, 23 critères, et 176 métriques au total, normalisées dans l'intervalle $[0; 1]$, la plupart se calculant lors du codage et de la conception.

Un autre élément à considérer : la couverture des tests.

On définit les objets à tester, et on calcule le ratio nombre d'objets testés (et avec les tests qui passent) / nombre d'objets à tester. Quand on est à 1, on a atteint notre objectif.

On teste tout en trois passes :

1. Niveau mesure : comparaison à des valeurs seuils
2. Niveau critères : classification du composant selon la valeur chiffrée, et selon une association mesure-critère qui a été définie et qui est propre à chaque critère et mesure.
3. Bilan : Classification du composant selon l'ensemble des critères associés.

On fait ensuite en graphe de Kiviat (spider chart : comme ci-dessous) :

avec des seuil : valeur minimum acceptable, valeur maximum acceptable. Quand un composant respecte toutes les normes, alors le polygone qui lui est associé est à l'intérieur du polygone de référence.

Il ne faut pas oublier qu'on est en démarche produit. On pourrait donc résumer l'approche retenue dans le schéma ci-dessous :

10.5 Conclusion

Les méthodes présentées sont assez vieilles, mais elle marchent. On doit cependant les remettre au goût du jour avec les nouvelles technologies (comme l'objet, qui a ses propres métriques). On peut avoir des nouveaux facteurs aussi, par exemple la disponibilité d'un serveur critique, la robustesse (on essaie de faire crasher le système, et on évalue sa capacité à tenir face aux attaques, voulues ou non).

La qualité logicielle est modélisable, mesurable.

Chapitre 11

Les bonnes pratiques en informatique : les référentiels

11.1 Intro

Utilité du référentiel : guide méthodologique efficace et label à mettre en avant.

11.2 Normes ISO/CEI 12220

11.2.1 Objectif d'un modèle

constituer cadre de référence

structurer organisation du cycle de vie : chronologie temporelle : points de rendez-vous entre les différents acteurs

caractériser points d'avancement du projet

permettre la gestion de la qualité

11.2.2 approche globale "modèle par processus"

Les modèles par processus ne préconisent pas de méthodes, techniques ou outils particuliers. Ils disent ce qu'il faut faire mais pas comment il faut le faire ni l'organisation à mettre en place, ni de cycle de vie de dvpt. Ils s'orientent progressivement vers l'approche processus généralisée.

11.2.3 la norme ISO 12207

La norme décrit des exigences pour l'acquisition, la fourniture, le dvpt, la maintenance, l'exploitation des produits logiciels.

Elle définit : activités, responsabilités des intervenants et politique de qualité.

Elle propose séparation nette entre tâches de l'acquéreur et celles du fournisseur, met en place un catalogue des procédures utiles.

Elle insiste sur :

- vérification et validation des exigences
- traçabilité des exigences
- anticipation de la maintenance dans phases amont
- interaction système/logiciel en amont du dvpt

Trois types de processus : métier, de support, organisationnels. Un processus est découpé en activités. A chaque activité on associe une personnes compétente, une mission, des responsabilités, des liens avec les autres acteurs.

11.2.4 Conclusion

La norme nécessite un guide de mise en pratique, c'est un référentiel de base.

ISO 12207 sert à évaluer le niveau méthodologique d'une entreprise (catalogue de ce qui est maitrisé et des

choses à améliorer.

11.3 CCMI (Capability Maturity Model Integrated)

11.3.1 définition

- ISO9001 → cahier des charges
- ISO15504 → spécifs générales
- CMMI → spécifs détaillées
- UP (ou USDP) processus unifié

11.3.2 historique

oui mais non

11.3.3 présentation générale

modèle de bonnes pratiques

objectif : améliorer qualité du produit et productivité ;

augmenter satisfaction du client ; réduire couts et respecter les délais

évaluation de la maturité d'une organisation (de 1 à 5)

11.3.4 méthode de mise en oeuvre

oui mais non

11.3.5 structure du modèle

le niveau de maturité est évalué pour les différents domaines de processus. Chaque domaine de processus est constitué d'objectifs spécifiques et d'objectifs génériques, eux-même composés de pratiques spécifiques et génériques.

11.3.6 organisation

CMMI : "souple sur la forme, dur sur le fond"

11.3.7 représentation étagée

5 niveaux de maturité :

1. initial : processus imprévisibles, mal controlés et réactifs
2. processus géré : un processus par projet, parfois réactif
3. processus défini : processus standardisés, gestion proactive
4. processus maîtrisé : processus maîtrisés et controlés
5. optimisation : amélioration continue des processus

OU

1. initial : processus peu prédictible, peu controlé et réactif
2. reproductible : stucturation et discipline
3. défini : standardisation et institutionnalisation
4. géré quantitativement : mesure prévision
5. optimisé : changement innovation

11.4 ITIL (Information Technology Infrastructure Library)

définition :

- ensemble cohérent des meilleures pratiques en matière gestion de services info
- modèle pour la production informatique
- approche par la définition de processus
- indépendant des technologies et de l'organisation

objectif :

- aligner SI avec les objectifs métier
- améliorer qualité des services fournis
- maîtriser coûts
- mettre en oeuvre une logique de service
- faire adhérer à ces principes et pas les imposer
- renforcer la notion de "gestion de projets"