

Génie Logiciel

MASTER I
AGITEL FORMATION

FORMATEUR : THIERRY MPOULI

Objectif du cours

- ▶ Permettre à l'étudiant d'approfondir ses connaissances du processus de développement de logiciel et de décomposition modulaire
- ▶ Permettre à l'étudiant d'analyser les différentes méthodes et les outils qui améliorent la qualité et diminuent le coût de développement et de gestion de systèmes logiciels

Organisation du cours

- ▶ 20 heures de cours
- ▶ 20 heures de travaux dirigés

Evaluation

- ▶ 1 projet individuel
- ▶ 2 évaluations individuelles
- ▶ 1 évaluation de groupe

Programme du cours

- ▶ Notions de Génie Logiciel
- ▶ Méthodes de développement du logiciel
- ▶ Spécification
- ▶ Conception
- ▶ Implémentation
- ▶ Maintenabilité et Evolution

Génie Logiciel Approfondi

Notions de génie logiciel

Qu'est ce que le génie logiciel ?

Le Génie logiciel (ou *software engineering*) :

- ▶ désigne « l'ensemble des méthodes, des techniques et outils concourant à la production d'un logiciel, au-delà de la seule activité de programmation ». [commentcamarche.net]
- ▶ Est un domaine des sciences de l'ingénieur dont l'objet d'étude est la *conception*, la *fabrication*, et la *maintenance des systèmes informatiques complexes*.

Qu'est ce qu'un système ?

Un système est

- un ensemble *d'éléments* interagissant entre eux suivant un certains nombres de principes et de *règles* dans le but de réaliser un *objectif*.

Qu'est ce qu'un logiciel ?

Un logiciel est

- ▶ en général un sous-système d'un système englobant.
- ▶ un ensemble d'entités nécessaires au fonctionnement d'un processus de traitement automatique de l'information.

Parmi ces entités, on trouve par exemple :

- Des programmes (e format code source ou exécutables)
- Des documentations d'utilisations
- Des informations de configuration

Qu'est ce qu'un logiciel ?

Un logiciel peut interagir avec des clients, qui peuvent être :

- ▶ Des opérateurs humains (utilisateurs, administrateurs, ...)
- ▶ D'autres logiciels (le système d'exploitation, les pilotes, ...)
- ▶ Des contrôleurs matériels (carte mère, processeur,)

Un logiciel réalise une *spécification* : son comportement vérifie un ensemble de critères qui régissent ses interactions avec son environnement.

Qu'est ce qu'un logiciel ?

Le génie logiciel vise à garantir que :

- ▶ La spécification répond aux besoins réels des clients
- ▶ Le logiciel respecte sa spécification
- ▶ Les coûts alloués pour sa réalisation sont respectés
- ▶ Les délais de réalisation sont respectés

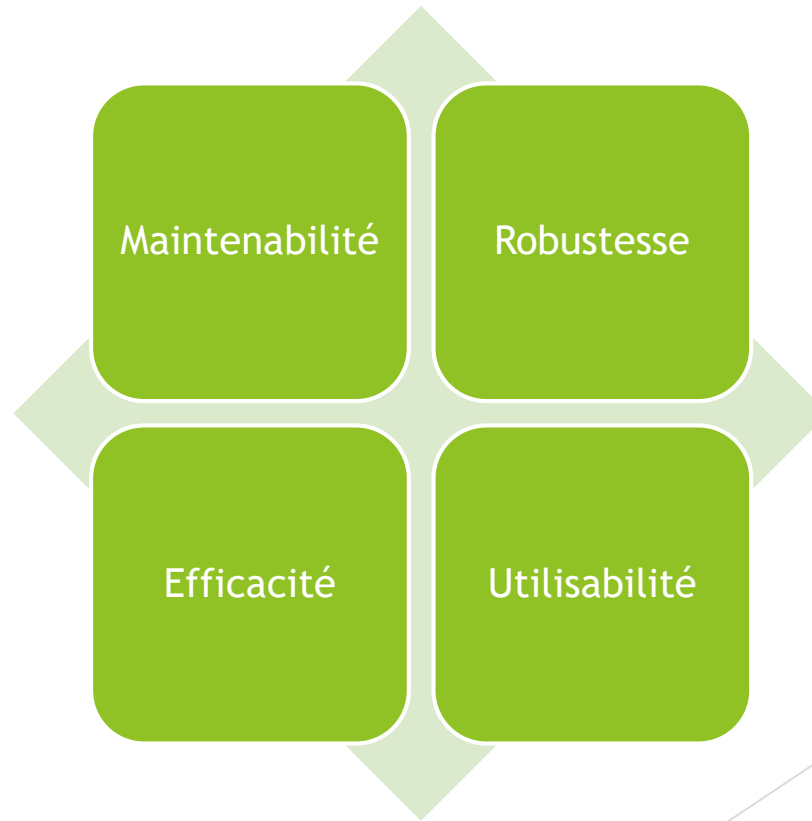
Comment spécifier un logiciel ?

- ▶ La spécification d'un logiciel peut prendre de nombreuses formes
- ▶ La complexité et les dimensions de la spécification peuvent varier énormément en fonction de l'environnement d'utilisation du logiciel et des objectifs auxquels il répond.



Comment fabriquer un bon logiciel ?

En plus du respect de sa spécification, la qualité d'un logiciel dépend de 4 critères.



Comment fabriquer un bon logiciel ?

Maintenabilité

► Peut-on faire évoluer le logiciel ?

Un logiciel ne s'use pas. Il peut subir des évolutions, des mises à jour, des corrections.

Exemple :

Une application qui fonctionne sous Windows 7 et qui ne fonctionne plus sous Windows 10 à cause d'une librairie manquante.

Cette application ne peut pas être considérée comme une application de qualité, à moins que le concepteur fasse une mise à jour pour que le critère de Maintenabilité soit respecté.

Comment fabriquer un bon logiciel ?

Robustesse

- **Le logiciel est-il sujet à des dysfonctionnements**

Exemple :

Photoshop se plante lorsque vous essayez d'utiliser l'outil rogner. Le logiciel lance un message d'erreur signalant un problème de mémoire.

Pour palier à cette erreur, Adobe a donné la possibilité de modifier la gestion de la mémoire via les paramètres de l'application Photoshop.

Ce plus permet de qualifier Adobe Photoshop comme un logiciel respectant le critère de Robustesse.

Comment fabriquer un bon logiciel ?

Efficacité

- **Le logiciel fait-il bon usage de ses ressources ?**

Exemple :

L'application Candy Crush peut fonctionner sous plusieurs versions d'Android. Néanmoins, sur certains téléphones (d'entrée de gamme), la seule utilisation de Candy Crush entraîne parfois surchauffe du téléphone, décharge rapide de la batterie, plantage, ...

Il est évident que les ressources de ces téléphones ne sont pas suffisantes pour le fonctionnement de Candy Crush. Néanmoins l'application peut quand même être installée sur ces téléphones.

Candy Crush ne respecte pas toujours le critère d'Efficacité

Comment fabriquer un bon logiciel ?

Utilisabilité

► Est-il facile à utiliser ?

Exemple :

Est-il compliqué de faire une addition sur l'application Calculatrice incluse dans Windows ?

L'application est facile à utiliser avec une interface qui est intuitive et donc facile à utiliser.

L'application Calculatrice respecte le critère d'utilisabilité.

Comment fabriquer un bon logiciel ?

Dans les années 70, le coût de construction du logiciel est devenu plus important que celui de la construction du matériel; cette prise de conscience fut appelée **la crise du logiciel**.

Pour répondre à cette crise, le génie logiciel a essayé d'appliquer les méthodes connues de l'ingénieur au domaine du logiciel, pour établir des méthodes fiables sur lesquelles construire une industrie du logiciel.

Il s'agit de se doter d'un cadre rigoureux pour :

- ▶ Guider le développement du logiciel, de sa conception à sa livraison
- ▶ Contrôler les coûts, évaluer les risques et respecter les délais
- ▶ Etablir des critères d'évaluation de la qualité d'un logiciel

Les spécificités du logiciel

Un logiciel a des caractères spécifiques dont il faut tenir compte. En effet, construire un logiciel selon les méthodes d'ingénieur diffère de la construction d'un pont:

- ▶ Un changement minime peut avoir des conséquences critiques
- ▶ Les progrès technologiques très rapides peuvent rendre un logiciel caduque
- ▶ Il est difficile de raisonner sur des programmes
- ▶ Les domaines des entrées des logiciels sont trop grands pour le test exhaustif
- ▶ Les défaillances des programmes sont en général dues à des erreurs humaines
- ▶ Chaque logiciel a son organisation et sa logique propre
- ▶ ...

Les grands principes du génie logiciel

Dans son ouvrage consacré aux fondamentaux du génie logiciel, **Carlo Ghezzi** souligne que la discipline ne peut se baser sur les méthodes, les techniques et les outils pour établir ses fondements. **La discipline doit plutôt s'appuyer sur un ensemble de principes fondamentaux qui seraient plus durables et moins sensibles au court cycle de vie des technologies.**

Carlo Ghezzi
Fundamentals of Software Engineering
Prentice Hall, 2nd edition, 2003.

Les grands principes du génie logiciel

La rigueur

- Les principales sources de défaillances d'un logiciel sont d'origine humaine
- Des outils de vérification accompagnant le développement peuvent aider à réduire les erreurs (CASE - Computer Aided Software Engineering)

La décomposition en sous-problèmes

- Décorréliser les problèmes pour n'en traiter qu'un seul à la fois
- Simplifier les problèmes (temporairement) pour aborder leur complexité progressivement

La modularité

- Couplée à la décomposition des problèmes, il s'agit de partitionner le logiciel en modules qui ont une cohérence interne et possèdent une interface ne divulguant que ce qui est strictement nécessaire aux modules clients.

Les grands principes du génie logiciel

L'abstraction

- Exhiber des concepts généraux regroupant un certain nombre de cas particuliers et de raisonner sur ces concepts plutôt que sur des cas particuliers.
- Fixer le bon niveau de détails permet de raisonner plus efficacement et de factoriser le travail en instanciant le raisonnement général sur chaque cas particulier

L'anticipation des évolutions

- La maintenance est la gestion des évolutions du logiciel. Il est primordial de prévoir les évolutions possibles d'un logiciel pour une maintenance efficace.

La généricité

- Un logiciel réutilisable a beaucoup plus de valeur qu'un composant dédié
- Un composant est générique lorsqu'il est adaptable

Les grands principes du génie logiciel

La construction incrémentale

- Un développement logiciel a plus de chances d'aboutir si il suit un cheminement étape par étape (incrémental)

La gestion de projets

Le rôle d'un chef de projet

Les tâches clés d'un chef de projet informatique

- ▶ Ecriture de proposition de projet (cahier de charge)
- ▶ Evaluation des coûts
- ▶ Planification du projet
- ▶ Surveillance du projet et écriture de rapport d'étapes
- ▶ Sélection du personnel et évaluation
- ▶ Ecriture de rapport de fin et de présentation du projet

La gestion de projets

Ecrire une proposition de projet

- ▶ A partir d'un appel d'offre, un chef de projet doit écrire une proposition de projet décrivant les objectifs du projet et les grandes lignes de sa réalisation
- ▶ Une proposition doit aussi contenir une évaluation des risques et des coûts
- ▶ Cette proposition requiert une importante expérience et compréhension du domaine d'activité. Le chef de projet engage sa responsabilité car la proposition sert aussi d'argumentaire pour justifier la mise en route du projet.

La gestion de projets

Planifier un projet

- ▶ Le chef de projet doit répartir les activités dans le temps en fonction de leurs dépendances, des ressources disponibles et d'une évaluation des risques liés à leur réalisation
- ▶ Cette tâche nécessite une connaissance très précise du domaine, des équipes de développement, ...
- ▶ Pour réaliser cette tâche, il existe plusieurs méthodes et outils parmi lesquels : les diagrammes de GANT, le réseau PERT, ...

La gestion de projets

Veiller sur un projet

- ▶ Le chef de projet contrôle le progrès des tâches et le respect des délais
- ▶ Il doit réévaluer la planification et même renégocier les ressources, les contraintes du projet en cas de retard ou dépassement de délai.
- ▶ En tout temps, un chef de projet doit s'armer d'indicateurs révélateurs sur l'état du développement du projet.

La gestion de projets

Sélectionner le personnel

- ▶ Le chef de projet doit affecter des activités et des rôles aux différentes personnes impliquées dans le projet
- ▶ Cette sélection doit être en concordance avec la politique de gestion du personnel (sous-traitance, formation continue, ...)
- ▶ Il est important de pouvoir évaluer le personnel sélectionné afin de déterminer si les qualités professionnelles correspondent au besoin du projet mais aussi que les qualités relationnelles soient présentes pour une bonne cohésion de l'équipe.

La gestion de projets

Ecrire un rapport

- ▶ Le chef de projet doit pouvoir communiquer une vue synthétique du projet à différents publics
- ▶ Il est important de disposer de rapports à chaque étape du projet qui sont validés par les autres parties du projet (équipe, clients, supérieurs, ...)

Quelques conséquences d'un mauvais logiciel

- ▶ **12/10/2006** : la « mauvaise utilisation » d'un logiciel à l'origine d'accidents de radiothérapie à Epinal - un décès et des complications chez treize patients traités pour des cancers...
Cause : mauvaise ergonomie d'un logiciel obsolète.
- ▶ **11/10/2006** : bug informatique, il faudra peut-être recompter à Liège ! Après une désastreuse soirée électorale avec des résultats annoncés vers 18H00 qui se sont rapidement révélés mathématiquement impossibles et des résultats définitifs seulement vers 23h45, certains candidats pensent à demander un recomptage et des explications.
- ▶ **29/12/2005** : cotisation retraite - Quelque 113 personnes ont eu la désagréable surprise de recevoir pour Noël un avis d'échéance de leurs cotisations retraite pour l'année 2006 de... deux milliards d'euros, un montant erroné dû à une erreur informatique, a-t-on appris mercredi de sources concordantes.
- ▶ **01/11/2005** : gros bug à la bourse de Tokyo, la plus importante d'Asie, et ce sont toutes les cotations qui sont bloquées toute la journée.