Génie du logiciel

Licence professionnelle IDSE
2012-2013
http://anubis.polytech.unice.fr/iut/2012_2013/lp/idse/gl/management

Mireille Blay-Fornarino
blay@unice.fr

QUIZ

DATE 2012-2013

Question

Génie du logiciel

A. Du baratin inutile
B. Des outils
C. Une méthodologie
D. Plusieurs éléments de tout cela
E. Autre : _____

Réponse :

Génie du logiciel

WHAT IS SOFTWARE ENGINEERING?

The IEEE Computer Society defines software engineering as

"(1) The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software.

(2) The study of approaches as in (1)."

http://www.swebok.org/

Questions

Gestion de projet
Cycle de vie du logiciel
Méthodes de développement
Modélisation
Architecture logicielle
Gestionnaire de version
Programmation par objets

Question

UML ?

A. Un nouvel objet non identifié
B. Un nouveau langage
C. Unified Modelling Language

Réponse :

Objectifs du cours

- Expliciter l'organisation des enseignements autour du GL
- Comprendre les objectifs et intérêts du GL

Cours de GL en LP

Gestion de projets et GL
- Cycle de conception et Artefacts (Atos)
- Méthodes agiles
- Utilisation de Redmine
- Business Process
- Architecture
- Refonte du SI
- Place de l'ergonomie dans la réalisation d'un projet

2.Outils pour le GL
- Développement logiciel
- Gestion de configuration
- Place des tests
- Intégration continue
- Gestion du changement
- Qualité du logiciel

Projet
Fil rouge

Evaluation du module Management

Gestion de projets et GL
- Cycle de conception et Artefacts (Atos)
- Méthodes agiles
  - ➡ Evaluation des user stories
- Utilisation de Redmine
- Business Process
  - ➡BP
- Architecture
  - ➡Architecture
- Refonte du SI
- Place de l'ergonomie dans la réalisation d'un projet
  - ➡Explicitation des choix et de la méthode ergonomique suivie.

Evaluation du projet tutoré

Gestion de projets et GL
- Méthodes agiles
  - ➡ suivis des évolutions
- Utilisation de Redmine
  - ➡ usage
- Business Process
- Architecture
  - ➡ cohérence de l'ensemble de l'analyse/conception
- Refonte du SI
- Place de l'ergonomie dans la réalisation d'un projet
  - ➡ Prise en compte de l'utilisateur

2. Outils pour le GL
- Développement logiciel
- Gestion de configuration
  - ➡ usage des configurations
- Place des tests
  - ➡ gestion des tests
- Intégration continue
  - ➡ mises en place de l'évaluation continue
- Gestion du changement
  - ➡ Gestion des bugs
- Qualité
  - ➡ Analyse de la qualité des codes

Projet
Fil rouge

➡Evaluation des rendus

➡Evaluation de la démonstration

➡Evaluation des codes

Introduction au GL et nouvelles approches «agiles»

Objectifs du cours

- Positionner les différents enseignements
- Comprendre les objectifs et intérêts du GL dans le cycle de vie du logiciel

Plan

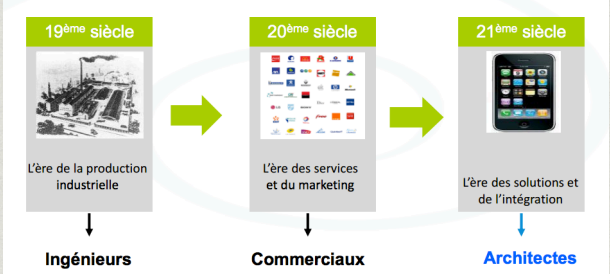
- Les entreprises dans un monde complexe
- Evolution des méthodes de développement : Bonnes pratiques
- Conclusion

Les entreprises dans un monde complexe

Les entreprises face à un monde complexe

- Un monde complexe:
  - Hyper-compétition, mondialisation, le temps se “raccourcit”
  - La puissance passe du coté du consommateur (F. Dupuy)
  - T. Friedman : « All that is easy has been done, what’s left is the hard stuff »
- Les problèmes compliqués requièrent des spécialistes, les problèmes complexes font appel à tous
  - Diversité des compétences et des points de vues …
  - … organisés en équipe
- Les problèmes complexes se traitent “sur le terrain” (gemba) un à la fois, là où ils se trouvent
  - Les abstractions cachent trop de choses, la décomposition ne marche pas!
  - “les conditions reproductibles” … ne le sont pas (isolation impossible)
  - La communication est difficile (ex: spécifier plus difficile que réaliser)

Les entreprises face à un monde complexe : changement de paradigme



L’Architecture des Systèmes est l’outil par excellence pour maîtriser la complexité

Les entreprise du 21e siècle doivent être agiles

- Court-terme (satisfaire ses clients)
  - Vitesse (lead time)
  - Zéro défauts (juste du premier coup)
  - Orienté-client
- Moyen-terme (suivre ses clients)
  - Flexibilité (s’adapter aux nouveaux besoins)
  - Réactivité (le faire rapidement)
- Long-terme (apprendre à évoluer)
  - Apprentissage (nouvelles compétences)
  - Travail d’équipe
  - Développement des collaborateurs

Innover, c’est faire

- Ce ne sont pas les idées qui comptent … c’est leur réalisation
  - Les « bonnes idées » sont partout, souvent sous des formes identiques
  - Cf. « Pretotyping Manifesto »
    - innovators beat ideas
    - doing beat talking
    - commitment beats committees
  - Cf. Facebook
    - done is better than perfect
- L’innovation digitale, c’est le code !
  - Code wins (Facebook)
  - Rough Consensus and Working Code (IETF)
- Recette « Startup »
  - Réunir développeur, marketer et designer

Outils de l’entreprise 2.0

Symptôme	Solution	Typologie	Web 2.0	exemple 0.0
• Convergence trop lente d’une édition « à plusieurs mains » (échanges multiples emails)	• Edition concurrente	• *N-to-N, synchrone ou asynchrone , immersif, participants explicits & implicites*	Google Docs	
• Toucher trop de monde avec un email, sans atteindre des collaborateurs pertinents	• Broadcasting ciblé	• *1-toN, participants implicites, immersif ou multi-tâche, sync/async.*	Blogger, YouTube, RSS	
• Surcharge de sollicitations synchrones (d’où le téléphone bloqué en messagerie)	• Comm. Synchrone élective (non-intrusive	• *1-to-1, explicite, synchrone, filtré du coté du recepteur, groupe fermé*	icq	

Outils de l'entreprise 2.0

Symptôme	Solution	Typologie	Web 2.0	exemple 0.0
• Demander une information simple prend trop de temps ou est perçu comme agressif	• Synchroni-sation émotionnelle	• *1-to-N, asynchrone implicite*		
• Pousser l'information au mauvais moment, aux mauvaises personnes	• Stigmergie dirigée	• *N-to-N, participants implicites, asynchrone.*		
• Surcharge, spam, trop de sollicitations par courriel	• Comm. Asynchrone élective	• *1-to-N, explicite, asynchrone, multi-tâche*		

Outils de l'entreprise 2.0

Symptôme	Solution	Typologie	Web 2.0	exemple 0.0
• Rigidité des structures de communication, silos, étroitesse de vue, manque de créativité	• Tissage des liens faibles	• *1-to-N, explicite, multi-tâche*		
• Toucher trop de monde avec un email, sans atteindre des collaborateurs pertinents	• Adressage par fonction	• *1-toN, participants implicites, immersif ou multi-tâche, sync/async.*		
• Processus trop lourd pour créer un projet	• Auto-organisation (emergence)	• *Organisation comme bénéfice collatéral d'un processus de communication,* • *Utilisation des cartes*		

Du plan à la Carte - Dominique Cardon

Evolutions des méthodes de développement

IBM Software Group

Simplification de :

PRJ270: Essentials of Rational Unified Process

Module 1: Best Practices of Software Engineering

Rational. software

Discussion: symptômes et causes profondes

- Quels sont les symptômes de problèmes de développement de logiciels?
- Quelles sont les causes profondes de ces symptômes ?

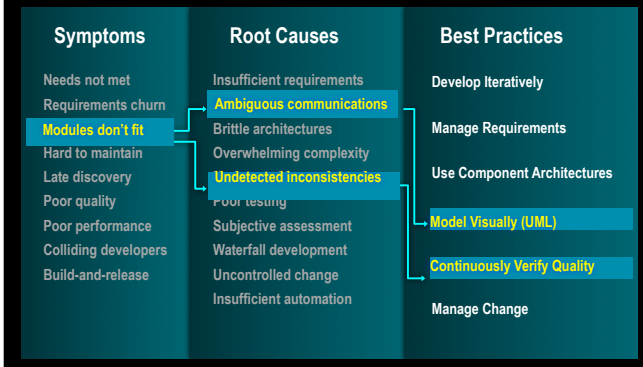
Symptômes de problèmes de développement logiciel

- Non réponse aux besoins
- Non couverture des exigences
- Mauvaise intégration
- Difficile à maintenir
- Découverte tardive de défauts
- Mauvaise qualité en terme de service rendu ou d'expérience utilisateur
- Mauvaises performances lors de la charge
- Aucun effort de coordination de l'équipe
- Problème de construction des versions finales

Causes des symptômes des problèmes de développement logiciel

- Prise en compte des exigences insuffisantes
- Communications ambiguës
- Architectures fragiles
- Complexité écrasante
- Incohérences non détectées
- Faiblesse des tests
- Evaluation subjective
- Développement en cascade
- Absence de contrôle du changement
- Pas d'automatisation

Trace Symptoms to Root Causes



Bonnes pratiques

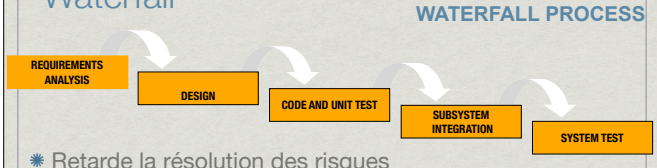
- Développer itérativement
- Gérer les exigences
- Travailler l'architecture
- Modéliser visuellement
- Vérifier la qualité en permanence
- Gérer les changements

Practice 1: Develop Iteratively

Best Practices
Process Made Practical

Develop Iteratively
Manage Requirements
Use Component Architectures
Model Visually
Continuously Verify Quality
Manage Change

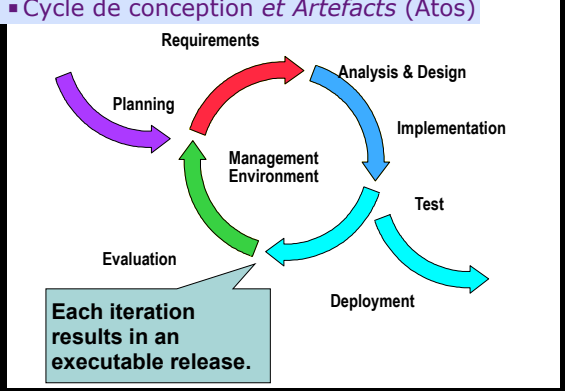
Caracteristiques du développement Waterfall



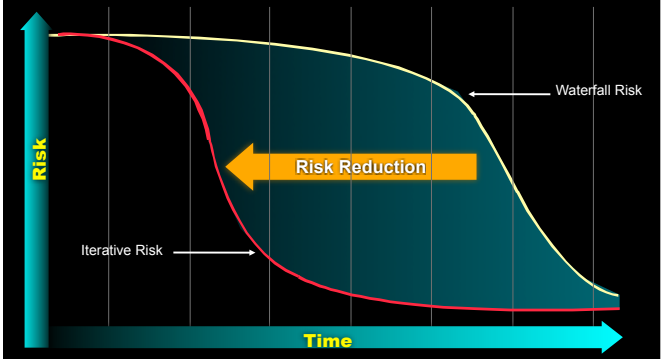
- Retarde la résolution des risques
- Mesure les progrès par des «produits» qui supportent difficilement la prédiction des temps avant complétion
- Retarde l'intégration et les tests
- Empêche le déploiement rapide
- Résulte souvent en de nombreuses itérations non planifiées

Iterative Development Produces Executables

- Cycle de conception *et Artefacts* (Atos)



Risk Profiles



IBM

Practice 2: Manage Requirements

Best Practices
Process Made Practical

Develop Iteratively
Manage Requirements
Use Component Architectures
Model Visually
Continuously Verify Quality
Manage Change

IBM

Gestion des exigences

- S'assurer que
  - Vous résolvez le bon problème
  - Vous construisez le système attendu
- en adoptant une approche systématique pour
  - susciter
  - organiser
  - documenter
  - gérer
- les besoins changeant d'une application logicielle.

▪ Développement logiciel ▪ Méthodes agiles

Practice 3: Use Component Architectures

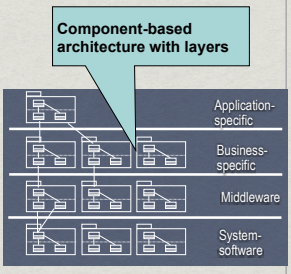
Best Practices
Process Made Practical

Develop Iteratively
Manage Requirements
Use Component Architectures
Model Visually
Continuously Verify Quality
Manage Change

IBM

Place de l'architecture

- L'architecture informatique définit la structuration d'un système informatique (i.e. matériel et logiciel) en termes de composants et d'organisation de ses fonctions.
- Base pour la réutilisation
- Base pour la gestion de projet
  - planification
  - attribution des ressources
  - livraison
- Base pour contrôler
  - gérer la complexité
  - maintenir l'intégrité



▪ Architectures

Practice 4: Model Visually (UML)

Best Practices
Process Made Practical

Develop Iteratively
Manage Requirements
Use Component Architectures
Model Visually
Continuously Verify Quality
Manage Change

IBM

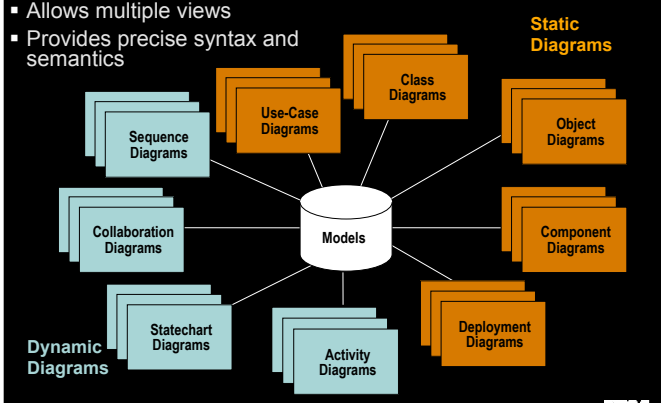
Modéliser visuellement

- Pour
  - Capturer la structure et le comportement
  - Expliciter les relations entre les éléments
  - Gardez la conception et la mise en œuvre cohérente
  - Cacher ou révéler les détails en fonction des besoins
  - Promouvoir la communication sans ambiguïté

- UML par ailleurs
- Business Process

Visual Modeling with Unified Modeling Language

- Allows multiple views
- Provides precise syntax and semantics



Pourquoi modéliser des processus ?

ON NE PEUT PAS CONTRÔLER CE QUE L'ON NE COMPREND PAS.

- Comprendre et contrôler les processus existants
  - temps, circuit, ressource, ...
- Améliorer les processus existants
  - rationaliser, étapes oubliés, sécuriser, ...
- Construire de nouveaux processus
- Communiquer sur les processus
- Automatiser les processus
  - Utilisation de moteur d'exécution.

- Business Process

Practice 5: Continuously Verify Quality

Best Practices
Process Made Practical

Develop Iteratively
Manage Requirements
Use Component Architectures
Model Visually (UML)
Continuously Verify Quality
Manage Change

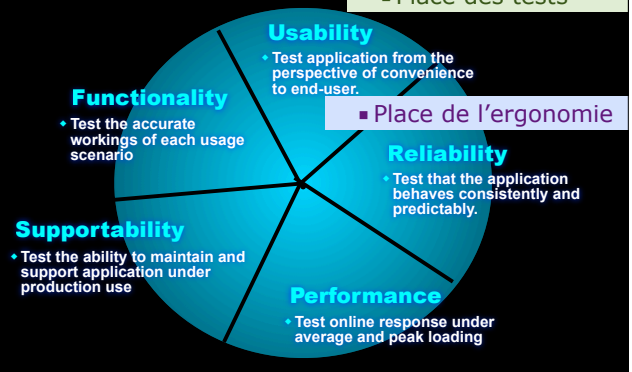
Continuously Verify Your Software's Quality

Software problems are 100 to 1000 times more costly to find and repair after deployment

- Cost to Repair Software
- Cost of Lost Opportunities
- Cost of Lost Customers

Testing Dimensions of Quality

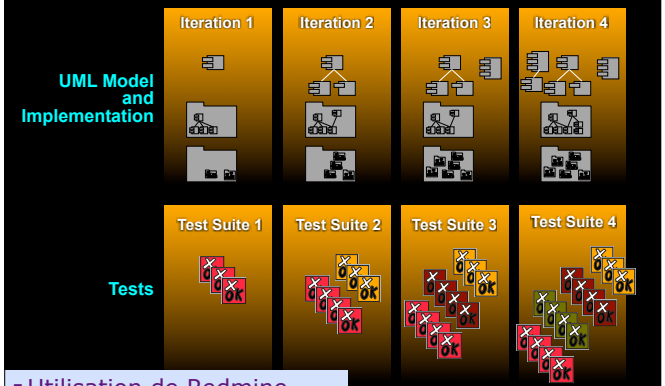
- Qualité du logiciel
- Place des tests



- Place de l'ergonomie

Test Each Iteration

- Gestion des configurations



- Utilisation de Redmine

IBM

Practice 6: Manage Change

Best Practices
Process Made Practical

Develop Iteratively
Manage Requirements
Use Component Architectures
Model Visually (UML)
Continuously Verify Quality
Manage Change

IBM

Que contrôler?

- Coordination des équipes
  - contributions de chacun
  - Développements parallèles
- Automatisation de l'intégration et gestion des assemblages
  - Gestion du changement
  - Intégration continue
  - Gestion des configurations
  - Refonte du SI

IBM, PRJ270: Essentials of Rational Unified Process

En résumé : Couverture du module

- Cycle de conception et Artefacts (Atos)
- Méthodes agiles
- Utilisation de Redmine
- Business Process
- Architecture
- Refonte du SI
- Place de l'ergonomie dans la réalisation d'un projet

- Développement logiciel
- Gestion de configuration
- Place des tests
- Intégration continue
- Gestion du changement
- Qualité du logiciel

Biblio pour GL

- Processus et Entreprise 2.0 Innover par la Collaboration et le Lean Management Club des Pilotes de Processus 5 Avril 2012 (v0.2) Yves CASEAU Bouygues Télécom – Académie des Technologies
- http://www.copilotes.eu/files/Livrable_WG3_BeerGame.pdf
- Lean Entreprise 2.0, Optimiser la communication, le pilotage et l'innovation, Conférence Lean & Services, 7 Décembre 2010, Yves CASEAU
- Cours IBM : PRJ270: Essentials of Rational Unified Process, Module 1: Best Practices of Software Engineering