

Principes et applications de la conception de logiciels

Groupe 10

Mardi, de 18h00 à 21h00 SH-3760 (cours)

Responsable(s) du cours

Nom du coordonnateur : MARTIN, Louis

Nom de l'enseignant : MARTIN, Louis

Local : PK-4330

Téléphone : (514) 987-3000 #3172

Courriel : martin.louis@uqam.ca

Site Web : www.martin.louis.uqam.ca

Description du cours

Rôle de la conception dans le cycle de vie du logiciel. Apprentissage des principales méthodes de conception. Évaluation de nouvelles méthodes de conception. Sélection et utilisation d'une méthode propre à un système logiciel donné. Évaluation de la conception: choix de la méthode, qualité de la conception, vérification formelle, respect des exigences, etc. Outils de conception.

Objectifs du cours

Comprendre la problématique de la conception dans les systèmes contemporains.

Situer les activités de conception dans le cycle de vie du logiciel.

Connaître les principales normes de la conception.

Initier les étudiants aux principaux modèles contemporains concernant la conception architecturale.

Initier les étudiants au concept d'urbanisation des systèmes d'information.

Donner aux étudiants un aperçu théorique des principales propriétés d'une bonne conception.

Initier les étudiants aux architectures 2-tiers et n-tiers.

Comprendre les tendances émergentes au niveau de la couche «présentation».

Familiariser les étudiants à l'architecture par composant.

Initier les étudiants aux principes de base de l'approche orientée objet.

Faire apprécier aux étudiants l'importance des enjeux liés à la conception.

Familiariser les étudiants aux principaux patrons utilisés pour la conception OO.

Sensibiliser les étudiants aux activités postérieures à la conception.

Donner aux étudiants un aperçu de l'impact des services Web sur l'architecture des systèmes (SOA) et sur la conception, en particulier au niveau de la granularité des services.

Initier les étudiants à la problématique de la persistance.

Sensibiliser les étudiants aux concepts de chorégraphie et d'orchestration des services Web.

Connaître certains frameworks architecturaux.

Initier les étudiants aux nouvelles approches dans le domaine.

Contenu du cours

1. Introduction

2. UML
3. Principes de base de la conception
4. Modèle en couches
5. La problématique des systèmes d'entreprises
6. La couche métier
7. Patrons classiques
8. La couche données
9. La persistance
10. La couche présentation – Un monde en mutation
11. Patrons et styles architecturaux
12. Injection de dépendance
13. SOA – SOAP – XML-RPC - REST
14. Programmation par aspect
15. Moteur de règles
16. MDA
17. Modélisation des processus
18. Urbanisation des SI
19. Frameworks architecturaux
20. Cocoon
21. Naked Objects
22. Langage de scripts
23. Qualité et métrique de la conception

Approche pédagogique

Le cours se base sur une approche par projet. L'approche par projet s'inscrit dans l'esprit de la formation par compétence. Il permet la mobilisation des ressources de l'étudiant dans la réalisation d'une tâche authentique.

Les étudiants devront travailler en équipe (taille: de 1 à 4 étudiants maximum). Il est fortement conseillé de ne pas effectuer les projets seul. Les critères de correction et les attentes ne seront pas modifiés en fonction du nombre d'étudiants dans l'équipe. Chaque équipe formera un groupe de consultants à qui seront confiés des mandats à réaliser. Chaque projet sera réalisé par l'ensemble des équipes. Cependant, chaque équipe aura une approche à promouvoir. L'attribution des approches sera effectuée par le professeur.

Chaque période de cours sera divisée en deux. La première partie abordera des thèmes choisis préalablement par les étudiants pour les aider dans la réalisation de leurs projets. Si aucun thème n'est proposé par les étudiants, un thème contemporain sera présenté par le professeur à son choix. La deuxième partie consistera à réviser avec chaque équipe le déroulement des travaux entrepris dans le but d'éviter des dérives éventuelles.

Modalités d'évaluation

Description sommaire	Date	Pondération
Projet Phase 1 (en équipe et présentation et document)	Mardi, le 4 octobre 2016	20%
Projet Phase 2 (en équipe et présentation et document)	Mardi, le 8 novembre 2016	20%
Projet Phase 3 (en équipe et présentation et document)	Mardi, le 6 décembre 2016	20%
Examen final	Mardi, le 13 décembre 2016	40%

Un travail remis en retard reçoit la note zéro à moins d'avoir fait l'objet d'une entente préalable avec le professeur.

Le détail des conditions de réalisation de chaque travail est précisé avec la description du travail.

Au plus tard le mercredi de chaque semaine, les étudiants devront avoir transmis par courriel au professeur une feuille de temps décrivant les activités accomplies lors de la semaine précédente (la date et l'heure de réception du courriel seront

utilisées pour contrôler le respect de la date de tombée). La forme et le contenu de cette feuille de temps seront précisés en classe. Pour tout retard, 1 % sera retranché à la note finale de l'étudiant.

La qualité du français fait partie intégrante des critères d'évaluation des travaux et des examens jusqu'à un maximum de 25%.

La politique de tolérance zéro du département d'informatique sera appliquée à l'égard des infractions de nature académique.

La note de passage du cours est de 60% pour l'ensemble de l'évaluation et de 50% pour l'examen final.

Politique d'absence aux examens

L'autorisation de reprendre un examen en cas d'absence est de caractère exceptionnel. Pour obtenir un tel privilège, l'étudiant-e doit avoir des motifs sérieux et bien justifiés.

Il est de la responsabilité de l'étudiant-e de ne pas s'inscrire à des cours qui sont en conflit d'horaire, tant en ce qui concerne les séances de cours ou d'exercices que les examens. **De tels conflits d'horaire ne constituent pas un motif justifiant une demande d'examen de reprise.**

Dans le cas d'une absence pour raison médicale, l'étudiant-e doit joindre un certificat médical original et signé par le médecin décrivant la raison de l'absence à l'examen. Les dates d'invalidité doivent être clairement indiquées sur le certificat. Une vérification de la validité du certificat pourrait être faite. Dans le cas d'une absence pour une raison non médicale, l'étudiant-e doit fournir les documents originaux expliquant et justifiant l'absence à l'examen – par exemple, lettre de la Cour en cas de participation à un jury, copie du certificat de décès en cas de décès d'un proche, etc. Toute demande incomplète sera refusée. Si la direction du programme d'études de l'étudiant-e constate qu'un étudiant a un comportement récurrent d'absence aux examens, l'étudiant-e peut se voir refuser une reprise d'examen.

L'étudiant-e absent-e lors d'un examen doit, dans les cinq (5) jours ouvrables suivant la date de l'examen, présenter une demande de reprise en utilisant le formulaire prévu, disponible sur le site Web du département à l'adresse suivante : <http://info.uqam.ca/politiques/>

L'étudiant-e doit déposer le formulaire dûment complété au secrétariat de la direction de son programme d'études : PK-3150 pour les programmes de premier cycle, PK-4150 pour les programmes de cycles supérieurs. Pour plus de détails sur la politique d'absence aux examens du Département d'informatique, consultez le site web suivant : <http://info.uqam.ca/politiques>

Intégrité académique

PLAGIAT Règlement no 18 sur les infractions de nature académique. (extraits)

Tout acte de plagiat, fraude, copiage, tricherie ou falsification de document commis par une étudiante, un étudiant, de même que toute participation à ces actes ou tentative de les commettre, à l'occasion d'un examen ou d'un travail faisant l'objet d'une évaluation ou dans toute autre circonstance, constituent une infraction au sens de ce règlement.

La liste non limitative des infractions est définie comme suit :

- la substitution de personnes;
- l'utilisation totale ou partielle du texte d'autrui en la faisant passer pour sien ou sans indication de référence;
- la transmission d'un travail pour fins d'évaluation alors qu'il constitue essentiellement un travail qui a déjà été transmis pour fins d'évaluation académique à l'Université ou dans une autre institution d'enseignement, sauf avec l'accord préalable de l'enseignante, l'enseignant;
- l'obtention par vol, manoeuvre ou corruption de questions ou de réponses d'examen ou de tout autre document ou matériel non autorisés, ou encore d'une évaluation non méritée;
- la possession ou l'utilisation, avant ou pendant un examen, de tout document non autorisé;
- l'utilisation pendant un examen de la copie d'examen d'une autre personne;
- l'obtention de toute aide non autorisée, qu'elle soit collective ou individuelle;
- la falsification d'un document, notamment d'un document transmis par l'Université ou d'un document de l'Université transmis ou non à une tierce personne, quelles que soient les circonstances;
- la falsification de données de recherche dans un travail, notamment une thèse, un mémoire, un mémoire-crédation, un rapport de stage ou un rapport de recherche;
- Les sanctions reliées à ces infractions sont précisées à l'article 3 du Règlement no 18.

Les règlements concernant le plagiat seront strictement appliqués. Pour plus de renseignements, veuillez consulter les sites suivants : <http://www.sciences.uqam.ca/etudiants/integrite-academique.html> et <http://www.bibliotheques.uqam.ca/recherche/plagiat/index.html>

Médiagraphie

VC BASS, Len; CLEMENTS, Paul & KAZMAN, Rick -- *Software Architecture in Practice (2ième édition)* -- 2003, Addison Wesley – ISBN: 0321154959

VC BAUER, Christian & KING, Gavin -- *Hibernate in Action* -- 2005, Manning – ISBN: 193239415X

VC BUDGEN, David -- *Software Design (2ième édition)* -- 2003, Addison Wesley – ISBN: 0201722194

VC BUSCHMANN, Frank & al. -- *Pattern-Oriented Software Architecture – A System of Patterns* -- 1996, Wiley – ISBN: 0471958697

VC CASEAU, Yves -- *Urbanisation et BPM – Le point de vue d'un DSI* -- 2005, Dunod – ISBN: 2100487248

VC CHAPPELL, David A. -- *Enterprise Service Bus* -- 2004, O'Reilly – ISBN: 0596006756

VC CLEMENTS, Paul & al. -- *Documenting Software Architecture – Views and Beyond* -- 2003, Addison Wesley – ISBN: 0201703726

VC Ed. BOURQUE, Pierre; DUPUIS, Robert -- *Guide to the Software Engineering Body of Knowledge* -- 2005, IEEE Computer Society – ISBN: 0769523307 Ce livre est disponible gratuitement en format PDF sur le site de SWEBOK.

VC ERL, Thomas -- *Service-Oriented Architecture – Concepts, Technology, and Design* -- 2005, Prentice Hall – ISBN: 0131858580

VC FOWLER, Martin -- *Patterns of Enterprise Application Architecture* -- 2003, Addison Wesley – ISBN: 0321127420

VC FREEMAN, Eric & FREEMAN, Elisabeth -- *Head First – Design Patterns* -- 2004, O'Reilly – ISBN: 0596007124

VC FRIEDMANN-HILL, Ernest -- *Jess in Action – Java Rule-based Systems* -- 2003, Manning – ISBN: 1930110898

VC GAMMA, Erich & others -- *Design Patterns – Elements of Reusable Object-Oriented Software* -- 1995, Addison Wesley – ISBN: 0201633612

VC GRAND, Mark -- *Java Enterprise Design Patterns – Patterns in Java Volume 3* -- 2002, Wiley – ISBN: 0471333158

VC HARROP, Rob & MACHACEK, Jan -- *Pro Spring* -- 2005, Apress – ISBN: 1590594614

VC HOHPE, Gregor & WOOLF, Bobby -- *Enterprise Integration Patterns* -- 2004, Addison Wesley – ISBN: 0321200683

VC JOHNSON, Rod & al. -- *Java Development with the Spring Framework* -- 2005, Wrox – ISBN: 0764574833

VO KAISLER, Stephen H. -- *Software Paradigms* -- 2005, Wiley - ISBN: 0471483478 Ce livre est obligatoire pour le cours.

VC LADDAD, Ramnivas -- *AspectJ in Action – Practical Aspect-Oriented Programming* -- 2003, Manning – ISBN: 1930110936

VC LAPASSAT, Guy -- *Urbanisme informatique et architectures applicatives* -- 2003, Hermes Science – ISBN: 2746206447

VC LONGÉPÉ, Christophe -- *Le projet d'urbanisation du S.I. – Démarche pratique avec cas concret* -- 2001, Dunod – ISBN: 2100073761

VC MAESANO, Libero; BERNARD, Christian; LE GALLES, Xavier -- *Services Web en J2EE et .NET* -- 2003, Eyrolles – ISBN: 2212110677

VC O'ROURKE, Carol; FISHMAN, Neal & SELKOW, Warren -- *Enterprise Architecture ‐ Using the Zachman Framework* -- **2003, Thomson ‐ ISBN: 0619064463**

VC PAWSON, Richard & MATTEWS, Robert -- *Naked Objects* -- **2002, Wiley ‐ ISBN: 0470844205**

VC ROZANSKI, Nick et WOODS, Eoin -- *Software Systems Architecture ‐ Working With Stakeholders Using Viewpoints and perspectives* -- **2005, Addison Wesley ‐ ISBN: 0321112296**

VC SCHMIDT, Douglas & al. -- *Pattern-Oriented Software Architecture ‐ Patterns for Concurrent and Networked Objects* -- **2000, Wiley ‐ ISBN: 0471606952**

VC SINGH, Inderjeet & al. -- *Designing Enterprise Applications (2ième édition)* -- **2002, Addison Wesley ‐ ISBN: 0201787903** Ce livre disponible gratuitement sur le site de Sun est une lecture incontournable pour connaître la plate-forme J2EE. Une nouvelle version de certains chapitres couvrant la version J2EE 1.4 est également disponible sur le site de Sun.

VC WALLS, Craig & BREIDENBACH, Ryan -- *Spring in Action* -- **2005, Manning ‐ ISBN: 1932394354**

SC -- *ESA PSS-05-04* <http://martin.louis.uqam.ca/domaines/normes/esa/pss0504.pdf>
Guide to the software architectural design phase

SC -- *ESA PSS-05-05* <http://martin.louis.uqam.ca/domaines/normes/esa/pss0505.pdf>
Guide to the software detailed design and production phase

SC -- *IEEE Std 1016.1-1993* Guide to Software Design Descriptions

SC -- *IEEE Std 1016-1998* Recommended Practice for Software Design Descriptions

SC -- *IEEE Std 1074-1997* Standard for Developing Software Life Cycle Processes

SC -- *IEEE Std 1471-2000* Recommended Practice for Architectural Description of Software-Intensive Systems

SC -- *IEEE/EIA 12207.0-1996* Guide for Information Technology ‐ Software life cycle processes

SC -- *IEEE/EIA 12207.1-1997* Guide for Information Technology ‐ Software life cycle processes ‐ Life cycle data

SC -- *IEEE/EIA 12207.2-1997* Guide for Information Technology ‐ Software life cycle processes ‐ Implementation considerations

SC -- *ISO/IEC 12207:1995* Information technology ‐ Software life cycle processe

UC -- *Apache* <http://cocoon.apache.org/>
Page d'accueil du projet Cocoon

UC -- *Apache* <http://struts.apache.org/>
Page d'accueil du projet Struts

UC -- *Extensible Interface Markup Language* <http://www.xml.org/>
Page d'accueil

UC -- *FEAPMO* <http://www.egov.gov/>
Federal Enterprise Architecture Program Management Offic

UC -- *Groovy* <http://groovy.codehaus.org/>
Page d'accueil

UC -- *Hibernate* <http://www.hibernate.org/>
Framework de persistance

UC -- *Hillside.net* <http://hillside.net/patterns/>
Bibliothèque de patterns

UC -- *IBM developerWorks* <http://www-128.ibm.com/developerworks/java/library/j-cb03216/index.html>
Continuations, Web development, and Java programming

UC -- *IFEAD* <http://www.enterprise-architecture.info/>
Institute For Enterprise Architecture Developments

UC Louis Martin <http://martin.louis.uqam.ca/>
Portail

UC Martin Fowler <http://www.dotnetguru.org/articles/dossiers/ioc/Fowler/loCFowler.pdf>
Article sur l'injection de dépendance

UC -- *Naked Objects* <http://www.nakedobjects.org/>
Page d'accueil

UC -- *OASIS* <http://docs.oasis-open.org/ubl/cd-UBL-1.0/>
Universal Business Language 1.0

UC -- *Object Management Group* <http://www.omg.org/mda/>
Page d'accueil pour le projet MDA (Model-Driven Architecture)

UC -- *OMG* <http://www.uml.org/>
Unified Modeling Language

UC -- *Open Group* <http://www.opengroup.org/architecture/togaf/>
The Open Group Architecture Framework

UC -- *OpenLaszlo* <http://www.openlaszlo.org/>
Plateforme pour la couche présentation

UC -- *Projet RIFE pour gérer les continuations WEB en Java* <http://rifers.org/>
Page d'accueil

UC -- *Spring Framework* <http://www.springframework.org/>
Framework d'injection de dépendance

UC -- *Spring Web Flow* <http://opensource.atlassian.com/confluence/spring/display/WEBFLOW/Home>
Abstraction de la continuation WEB

UC -- *Sun* http://java.sun.com/blueprints/guidelines/designing_enterprise_applications_2e/index.html
J2EE Architecture Blue Book 2

UC -- *SWEBOK* <http://www.swebok.org/>
Guide to the Software Engineering Body of Knowledge

UC -- *The Server Side* <http://www.theserverside.com/>
Le site de Ed Roman sur J2EE

UC -- *User Interface Markup Language* <http://www.uiml.org/>
Page d'accueil

UC -- *Zachman Institute for Framework Advancement (ZIFA)* <http://www.zifa.com/>
Page d'accueil