

Génie logiciel: conception

Groupe 30

Mercredi, de 18h00 à 21h00 SH-2560 (cours)

Jeudi, de 18h00 à 20h00 PK-S1540 (atelier)

Responsable(s) du cours

Nom du coordonnateur : MOHA, Naouel

Nom de l'enseignant : BOUBAKER, Anis

Local : PK-4115

Téléphone : (514) 987-3000 #3699

Courriel : boubaker.anis@uqam.ca

Description du cours

Sensibiliser l'étudiant aux difficultés de la conception et lui permettre d'élaborer des solutions réutilisables, maintenables et extensibles. Problématique du processus de conception. Critères et architecture. Conception comme activité créatrice. Outils d'aide à la conception. Intégration et essais système. Conception orientée objet. Cadres d'application et patrons de conception. Documentation de conception. Rétro ingénierie. Ce cours comporte une séance obligatoire de laboratoire.

Préalables académiques :

INF5151 Génie logiciel: analyse et modélisation ; INF3135 Construction et maintenance de logiciels

Objectifs du cours

- Donner aux étudiants un aperçu théorique du cycle de vie du logiciel et en particulier de la conception.
- Sensibiliser les étudiants à l'importance des normes en génie logiciel et leur faire connaître les principales références dans le domaine concernant la conception.
- Familiariser les étudiants avec les activités antérieures à la conception.
- Initier les étudiants aux principaux modèles contemporains concernant la conception architecturale.
- Donner aux étudiants un aperçu théorique des principales propriétés d'une bonne conception.
- Initier les étudiants aux architectures 2-tiers et n-tiers.
- 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.
- Rendre les étudiants aptes à réaliser un document simple de conception.
- Sensibiliser les étudiants aux activités postérieures à la conception.
- Initier les étudiants aux nouvelles approches dans le domaine.

Formules pédagogiques

Le cours sera orienté autour d'une étude de cas réalisée tout au long de la session. À travers cette étude de cas, les étudiants seront amenés à appliquer et se sensibiliser aux difficultés liées aux différents champs de la conception orientée objet. Les concepts théoriques seront présentés à travers des exposés magistraux et les notes de cours (acétates) présentées seront

disponible sur le site de Moodle.

Durant les séances de laboratoires, les étudiants pourront appliquer les notions présentées en cours. Plusieurs séances de laboratoire seront également consacrées à assister les étudiants dans la réalisation de leurs travaux pratiques.

Modalités d'évaluation

Description sommaire	Date	Pondération
Examen commun intra	2015-03-11	30%
Examen commun final	2015-04-29	30%
TP 1	2015-02-11 @17h00	10%
TP 2	2015-03-25 @17h00	15%
TP 3	2015-04-22 @17h00	15%

- 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 TP sera précisé avec la description du TP.
- La qualité du français fait partie intégrante des critères d'évaluation des travaux et des examens jusqu'à un maximum de 25%.
- La note de passage du cours est de 60% pour l'ensemble de l'évaluation et de 50% pour les deux examens combinés.

Les règlements concernant le plagiat seront strictement appliqués. Pour plus de renseignements, consultez le site suivant : <http://www.sciences.uqam.ca/etudiants/integrite-academique.html>

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>

Calendrier détaillé du cours

Ce plan de cours est un plan préliminaire. Il présente les notions théoriques qui seront abordées en cours et qui seront complétées par l'application au cas étudié ou d'autres exemples concrets. Cependant, les contenus présentés à chaque séance peuvent varier selon l'avancement et les besoins des étudiants.

	Semaine	Cours	Démo
01	05 jan.	- Présentation du cours - Introduction	Pas de démonstration
02	12 jan.	- Rappels de modélisation UML; - Présentation du cas d'étude	Modélisation: diagrammes de classes et de séquences.
03	19 jan.	- Processus de développement logiciel (2/2) - Conception architecturale (1/2) - Énoncé TP1	- Modélisation: diagrammes de séquences.
04	26 jan.	- Conception architecturale (2/2)	- Conception architecturale - Support TP1
05	2 fév.	- Développement orienté par les modèles (introduction) - Le cadre EMF (tutoriel)	- Conception et génération de code avec EMF (1/2)
06	9 fév.	- Patrons GRASP - Remise TP1: 2015-02-11 @17h00 - Énoncé TP2	- Conception et génération de code avec EMF (2/2) - Assignment de responsabilités / GRASP
07	16 fév.	- Patrons de conception: Introduction - Patrons: Composite et Singleton	- Patron composite - Support TP2
08	23 fév.	- Patrons: Visiteur, Observateur, Décorateur.	- Révision pour l'intra (Question/Réponse - 1h) - Support TP2 (1h)
	2 mar	<i>Semaine de lecture</i>	<i>Semaine de lecture</i>
09	9 mar.	Examen Intra	- Patron Visiteur (AST)
10	16 mar.	- Patrons: Fabrique, Stratégie, Memento	- Support TP2
11	23 mar.	- Patrons: Adaptateur, Façade, Fabrique abstraite - Remise TP2: 2015-03-25 @17h00 - Énoncé TP3	- Patrons Composite et Décorateur et Stratégie
12	30 mar.	- Patrons: Itérateur, Pont, Proxy, Façade, Poids mouche	- Patron Fabrique abstraite
13	6 avr.	- Patrons d'in. logicielle: Injection de dépendance, DAO. - Révision des patrons de conception	- Patrons Memento, Décorateur. - Support TP3
14	13 avr.	- Antipatrons - Refactorisation - Notions avancées: architectures SOA et REST;	- Support TP3
15	20 avr.	- Notions avancées: Conception de processus d'affaires. - Révision pour l'examen final - Remise TP3: 2015-04-22 @17h00	- Révision pour l'examen final (Question/Réponse)
16	27 avr.	Examen Final	<i>Pas de démonstration.</i>

Renseignements utiles

Le cours INF5151 est un préalable à ce cours. Particulièrement, il est important que les étudiants maîtrisent correctement la norme UML (modèle de classes, modèle de séquences, modèle de packages, modèle de composants). Bien qu'un bref rappel de modélisation sera présenté, il est fortement recommandé aux étudiants de rafraîchir leurs connaissances en modélisation en vue de suivre ce cours.

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, manœuvre 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 liées 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

VO [LAR05] LARMAN, Craig -- *UML 2 et les design patterns* -- Troisième édition, Pearson Education, 2005, ISBN 2-7440-7090-4.

OU

VO [LAR05] LARMAN, Craig -- *Applying UML and Patterns - An Introduction to Object-Oriented Analysis and Design* -- Third Edition, Prentice Hall, 2005, ISBN 0-13-148906-2.

VR [ARL05] ARLOW, Jim et NEUSTADT, Ila -- *UML2 and the Unified Process: Practical Object-Oriented Analysis and Design, second edition* -- Addison-Wesley Object Technology Series, 2005

VR [BAS03] BASS, Len, CLEMENTS, Paul et KAZMAN, Rick -- *Software Architecture in Practice, second edition* -- SEI Series in Software Engineering, Addison-Wesley, 2003

UC [BAN10] BANAS, Derek - Design Patterns Video Tutorials, <http://goo.gl/znWcZ9>

VC [BEN97] BENNETT, Douglas -- *Designing Hard Software - The Essential Tasks* -- Manning, 1997 ISBN 0-13-304619-2.

VC [BER99] BERNADAC, Jean-Christophe & KNAB, François -- *Construire une application XML* -- Eyrolles, 1999 ISBN

2-212-09081-1.

VC **[COA99]** COAD, Peter, LEFEBVRE, Eric & De LUCA, Jeff -- *Java Modeling Color with UML - Enterprise Components and Process* -- **Prentice Hall, 1999 ISBN 0-13-011510-X .**

VC **[FOW04]** FOWLER, Martin -- *UML Distilled -- Third Edition*, Addison-Wesley, 2004 ISBN 0-321-19368-7.

VC **[FOW99]** FOWLER, Martin & others -- *Refactoring - Improving the Design of Existing Code* -- Addison-Wesley, 1999 ISBN 0-201-48567-2.

VC **[GAM95]** GAMMA, Erich & others *Design Patterns - Elements of Reusable Object-Oriented Software -- Design Patterns - Elements of Reusable Object-Oriented Software* -- Addison-Wesley, 1995 ISBN 0-201-63361-2

VC **[GRA02]** GRAND, Mark -- *Patterns in Java - Volume 1* -- Second Edition, Wiley, 2002 ISBN 0-471-22729-3.

VC **[GRA99]** GRAND, Mark -- *Patterns in Java - Volume 2* -- Wiley, 1999 ISBN 0-471-25841-5.

VC **[HOF00]** HOFMEISTER, Christine & others -- *Applied Software Architecture* -- Addison-Wesley, 2000 ISBN 0-201-32571-3.

VC **[MEY97]** MEYER, Bertrand -- *Object-Oriented Software Construction* -- Second Edition Prentice Hall, 1997 ISBN 0-13-629155-4.

VC **[MILI02]** MILI, Hafedh, MILI, Ali, YACOUB, Sherif et ADDY, Edward -- *Reuse-Based Software Engineering: Techniques, Organization and Controls* -- John Wiley & Sons, 2002

VC **[MUL00]** MULLER, Pierre-Alain -- *Modélisation objet avec UML* -- Deuxième édition, Éditions Eyrolles, 2000 ISBN 2-212-09122-2.

VC **[ROZ05]** ROZANSKI, Nick & WOODS, EOIN -- *Software Systems Architecture* -- Addison-Wesley, 2005 ISBN 0-321-11229-6

VC **[RUM05]** RUMBAUGH, James & others -- *Object-Oriented Modeling and Design with UML* -- Second Edition, Prentice Hall, 2005 ISBN 0-13-015920-4.

VC **[SHAW96]** SHAW, Mary et GARLAN, David -- *Software Architecture: Perspectives on an Emerging Discipline* -- Prentice-Hall, 1996

VC **[SZY98]** SZYPERSKI, Clemens -- *Component Software - Beyond Object-Oriented Programming* -- Addison-Wesley, 1998 ISBN 0-201-17888-5.

VC **[WIL95]** WILKINSON, Nancy M. -- *Using CRC Cards - An Informal Approach to Object-Oriented Development SIGS* -- 1995 ISBN 1-884842-07-0.

<p>A : article - C : comptes rendus - L : logiciel S: Standard - U : uri - V : volume</p>	<p>C : complémentaire - O : Obligatoire - R : recommandé</p>
---	--