Université du Québec en Outaouais **Département d'informatique et d'ingénierie**

Sigle : INF1163 Gr. 01
Titre : Modélisation et conception orientée objet
Session : Automne 2019 Horaire et local
Professeur : El Guemhioui, Karim

1. Description du cours paraissant à l'annuaire :

Objectifs

Introduire l'étudiant à la modélisation et à la conception logicielle et lui permettre d'élaborer des solutions réutilisables et extensibles; le familiariser avec un langage de modélisation.

Contenu

Processus de conception orientée objet. Cas d'utilisation. Modèle conceptuel. Architectures logicielles. Conception par contrat et comportement. Patrons de conception. UML. Outils d'aide à la conception. Génération automatique du code. Tests orientés objets. Ce cours comporte des séances obligatoires de travaux dirigés (TD) de deux heures par semaine.

Descriptif – Annuaire

2. Objectifs spécifiques du cours :

Ce cours couvre 4 des 12 qualités requises des diplômé(e)s telles que définies dans les normes d'agrément des programmes de génie au Canada (http://www.engineerscanada.ca/fr/ressources-en-matiere-dagrement) :

a. Qualité 1 : Connaissance en génie

b. Qualité 4 : Conception

c. Qualité 5 : Utilisation d'outils d'ingénierie

d. Qualité 6 : Travail individuel et en équipe

Les qualités 4 et 6 sont mesurées dans ce cours pour fins de rétroaction.

Objectifs spécifiques	Qualité	Indicateurs	Introduit	Développé	Appliqué
Concevoir et développer une solution logicielle à un problème complexe en répondant aux besoins spécifiés et aux contraintes existantes.	4	1- Identifier les besoins des clients et les contraintes économiques, réglementaires et législatives, environnementales, culturelles, sociales, et de santé et sécurité.		x	
	4	2- Produire et comparer différentes solutions possibles afin de sélectionner le meilleur concept.		x	
	4	3- Créer des modèles, simulations, prototypes, et faire des tests.		x	
Travailler en équipe et faire preuve d'initiative.	6	2- Contribuer équitablement au travail d'équipe		x	

	6	3- Contribuer à l'efficacité de l'équipe : participation, initiative, résolution de conflit, etc.		x	

3. Stratégies pédagogiques :

Les formules pédagogiques suivantes seront utilisées :

- Cours magistraux intégrant des exemples
- Séances de travaux dirigés
- Devoirs
- Projet de conception
- Examen de mi-session
- Examen final

Les travaux à terme devront être remis aux dates indiquées et en début de séance; aucun retard ne sera toléré.

4. Heures de disponibilité ou modalités pour rendez-vous :

Communications par courriel et via le forum du cours sur Moodle.

Rencontres personnelles sur rendez-vous (local B-0123).

5. Plan détaillé du cours sur 15 semaines :

Semaine	Thèmes	Dates
1	**Architectures logicielles** • Principes et problématique de la conception • Stratégies et styles architecturaux • Décomposition et modularité • Cohésion et couplage	05 sept. 2019
2	**Introduction au langage UML** • Classes et objets • Associations et liens • Multiplicités et noms de rôles • Associations qualifiées • Classes associatives • Contraintes • Agrégations et compositions • Diagrammes d'interaction	12 sept. 2019
3	**Les cas d'utilisation** • Rappels : Notations et formats • Scénario principal et scénarios alternatifs • Test du patron, test PME, test de la taille • Relations entre cas d'utilisation • Inclusion, extension, généralisation/spécialisation	19 sept. 2019

	Travail dirigé 1 : Introduction à la modélisation avec UML (25 ou 27 septembre 2019)	
4	**Introduction au processus unifié** • Processus unifié et méthodes agiles • Technique du *timeboxing* • Les phases du processus unifié • Les disciplines du processus unifié • Itération et incrémentation • Introduction à une étude de cas	26 sept. 2019
5	**Examen de mi-session**	03 oct. 2019
6	**Le modèle du domaine** • Différence entre classe conceptuelle et classe de conception • Identification des concepts • Identification des associations • Identification des attributs • Pièges et erreurs à éviter • Glossaire **Travail dirigé 2 : Modèle du domaine (23 ou 25 octobre 2019)**	10 oct. 2019
7	**Semaine d'études**	17 oct. 2019
8	**DSS et contrats** • Diagramme de séquence du système • Éléments d'un contrat • Transition vers la conception • Responsabilités et méthodes • Cas d'utilisation au style concret **Travail dirigé 3 : DSS et contrats (30 octobre ou 01 novembre 2019)**	24 oct. 2019
9	**Les patrons de conception** • Définitions • Le patron Expert • Le patron Créateur • Le patron Contrôleur • Les patrons Faible couplage et Forte cohésion • Principe de la séparation des problèmes **Travail dirigé 4 : Design Patterns (06 ou 08 novembre 2019)**	31 oct. 2019
10	**Le modèle de conception** • Catégories de visibilité • Relation de dépendance • Patrons de conception Fabrication pure et Singleton • Finalisation du diagramme de classes • Conception en couches	07 nov. 2019

	• Transition vers le code	
11	**Modélisation des états et des activités** • Diagramme d'états • Événements et transitions • Effets et gardes • États imbriqués, états concurrents • Autres diagrammes de l'UML	14 nov. 2019
12	**Évolution du logiciel** • Introduction aux tests • La maintenance • Rétro-ingénierie et réingénierie • Réflexion sur le processus	21 nov. 2019
13	**Tests du logiciel** • Techniques de tests : cas de l'orienté objet • Développement piloté par les tests	28 nov. 2019
14	**Projet de session** • Présentations et démos	05 déc. 2019
15	**Examen final**	12 déc. 2019

6. Évaluation du cours :

Il faut obtenir au moins 50 % de la note aux examens (mi-session et final) pour que les notes des autres travaux (devoirs et projets) soient prises en compte dans le calcul de la moyenne finale.

Outils d'évaluation	Pondération	Indicateurs mesurés
Devoirs	10 %	
Examen intra	35 %	4.1 ; 4.2
Examen final	35 %	4.3
Projet	20 %	4.1; 4.2; 4.3; 6.2; 6.3

Par **indicateur mesuré**, on entend qu'à la fin du cours, un niveau de performance (0, 1, 2, 3) est donné pour chaque indicateur et pour chaque étudiant(e) selon la grille ci-dessous :

Indicateurs	Niveau 0	Niveau 1	Niveau 2	Niveau 3

4.1 - Identifier les besoins des clients et les contraintes économiques, réglementaires et législatives, environnementales, culturelles, sociales, et de santé et sécurité.	Identification inadéquate des besoins et des contraintes	Identification des besoins acceptable, mais détermination des contraintes insuffisante	Identification acceptable des besoins et des contraintes	Identification exhaustive des besoins et des contraintes
4.2 - Produire et comparer différentes solutions possibles afin de sélectionner le meilleur concept.	Production et comparaison de solutions possibles inadéquates ou inexistantes	Production et comparaison de solutions possibles acceptables, mais sélection du meilleur concept inadéquate	Production et comparaison de solutions possibles et sélection du meilleur concept acceptables	Production, comparaison et sélection remarquables
4.3 - Créer des modèles, simulations, prototypes, et faire des tests.	Création de modèles, simulations, prototypes et/ou exécution des tests inadéquates ou inexistantes.	Création acceptable de modèles, simulations, prototypes, mais exécution de tests insuffisante.	Création de modèles, simulations, prototypes et exécution de tests adéquates.	Création de modèles, simulations, prototypes et exécution de tests remarquables.
6.2 - Contribuer équitablement au travail d'équipe.	Contribution inexistante ou controversée	Contribution minimale	Contribution équitable	Contribution exceptionnelle
6.3 - Contribuer à l'efficacité de l'équipe : participation, initiative, résolution de conflit, etc.	Contribution à l'efficacité de l'équipe inacceptable ou inexistante	Contribution minimale à l'efficacité de l'équipe	Contribution acceptable à l'efficacité de l'équipe	Contribution remarquable à l'efficacité de l'équipe

7. Politiques départementales et institutionnelles :

- Politique du département d'informatique et d'ingénierie relative à la tenue des examens
- Note sur le plagiat et sur la fraude
- Politique relative à la qualité de l'expression française écrite chez les étudiants et les étudiantes de premier cycle à l'UQO
- Absence aux examens : cadre de gestion, demande de reprise d'examen (formulaire)
- *SANS OUI C'EST NON!*

Travaillons ensemble pour développer une culture du respect ! La communauté universitaire de l'UQO se mobilise et lance un message haut et fort de **tolérance zéro en matière de violence à caractère sexuel** (pour de plus amples renseignements, veuillez consulter la page Web : uqo.ca/sansouicestnon).

8. Principales références :

- Craig Larman, UML 2 et les design-patterns, Pearson Education France, 3e édition, 2005
- Eric Gamma et al., Design Patterns - Elements of Reusable Object-Oriented Software, Addison-Wesley, 1995 (version française)
- Ian Sommerville, Software Engineering, 9e ed., Addison-Wesley, 2011
- Audibert, Laurent, UML 2 de l'apprentissage à la pratique, ellipses, 2e édition, 2014

9. Page Web du cours :

https://moodle.uqo.ca