

**Sigle : INF1163 Gr. 01**

**Titre : Modélisation et conception orientée objet**

**Session : Été 2026, mardi et jeudi de 8h30 à 11h30 au local A0112**

**Professeur : Trabelsi, Abdel**

**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-dagrément>) :

- 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 approches pédagogiques suivantes seront utilisées :

#### Séances de cours :

- Les séances de cours auront lieu en mode présentiel.
- Les connaissances seront présentées sous forme de cours magistraux.
- Le matériel pédagogique sera mis à la disposition des étudiant(e)s sur Moodle.
- Un forum de discussion sera aussi mis à la disposition des étudiant(e)s afin de leur permettre de poser leurs questions et, le cas échéant, de contribuer à l'élaboration des réponses.

#### Séances de TD :

- Les séances de TD auront lieu en mode présentiel sous la supervision de l'assistant d'enseignement.
- Les travaux à terme devront être remis aux dates indiquées; aucun retard ne sera toléré.

#### Projet de session :

- Travail individuel ou en binôme (maximum 2 étudiants) afin d'approfondir les notions théoriques vues en cours.

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

Communications par courriel et via le forum du cours sur Moodle. [trabab01@uqo.ca](mailto:trabab01@uqo.ca)  
Rencontres personnelles sur demande.

### 5. Plan détaillé du cours sur 7 semaines :

Séances	Thèmes	Dates
1	<b>Architectures logicielles</b> <ul style="list-style-type: none"><li>• Principes et enjeux de la conception logicielle</li><li>• Stratégies et styles architecturaux</li><li>• Décomposition et modularité</li><li>• Facteurs de qualité logicielle</li></ul>	30 juin 2026
2	<b>Introduction au langage UML</b> <ul style="list-style-type: none"><li>• Les éléments UML</li><li>• Les relations UML</li><li>• Les diagrammes UML</li></ul>	2 juillet 2026
3	<b>Les cas d'utilisation</b> <ul style="list-style-type: none"><li>• Introduction aux cas d'utilisation</li><li>• Description des cas d'utilisation</li><li>• Modélisation des cas d'utilisation</li><li>• Tests des cas d'utilisation</li><li>• Relations entre cas d'utilisation</li></ul>	7 juillet 2026
4	<b>Introduction au processus unifié</b> <ul style="list-style-type: none"><li>• Méthodologies de développement logiciel</li><li>• Processus Unifié</li><li>• Timeboxing dans le Processus Unifié</li><li>• Disciplines du Processus Unifié</li><li>• Phases du Processus Unifié</li></ul> <b>Travail dirigé 1 (10 juillet 2026)</b>	9 juillet 2026
5	<b>Modèle du domaine</b> <ul style="list-style-type: none"><li>• Identification des classes conceptuelles</li><li>• Identification des associations</li><li>• Identification des attributs</li><li>• Modèle du domaine dans le Processus Unifié</li></ul>	14 juillet 2026

6	<p><b>DSS et contrats d'opération</b></p> <ul style="list-style-type: none"> <li>• Diagramme de séquence système (DSS)</li> <li>• Contrats d'opération</li> <li>• Transition vers la conception</li> <li>• Responsabilités et méthodes</li> <li>• Cas d'utilisation concret</li> </ul> <p><b>Travail dirigé 2 (17 juillet 2026)</b></p>	16 juillet 2026
7	<p><b>Examen intra</b></p>	21 juillet 2026
8	<p><b>Patrons de conception</b></p> <ul style="list-style-type: none"> <li>• Les patrons GRASP <ul style="list-style-type: none"> <li>○ Créateur</li> <li>○ Expert</li> <li>○ Faible Couplage</li> <li>○ Contrôleur</li> <li>○ Forte Cohésion</li> </ul> </li> <li>• Principe de séparation des problèmes</li> </ul>	23 juillet 2026
9	<p><b>Modèle de conception</b></p> <ul style="list-style-type: none"> <li>• Visibilité entre objets</li> <li>• Patrons de conception <ul style="list-style-type: none"> <li>○ Fabrication Pure</li> <li>○ Singleton</li> </ul> </li> <li>• Diagramme de classes</li> </ul>	28 juillet 2026
10	<p><b>Modèle de conception (suite)</b></p> <ul style="list-style-type: none"> <li>• Relations de dépendance</li> <li>• Conception en couches</li> </ul> <p><b>Travail dirigé 3 (31 juillet 2026)</b></p>	30 juillet 2026
11	<p><b>Modélisation des états et des activités</b></p> <ul style="list-style-type: none"> <li>• Diagramme d'activités UML</li> <li>• Diagramme d'états UML</li> <li>• Événements et transitions</li> <li>• États imbriqués, états concurrents</li> <li>• Autres diagrammes UML</li> </ul>	4 août 2026
12	<p><b>Développement piloté par les tests</b></p> <ul style="list-style-type: none"> <li>• Vérification et validation</li> <li>• Revues statiques</li> <li>• Stratégies de tests logiciels</li> </ul> <p><b>Travail dirigé 4 (7 août 2026)</b></p>	6 août 2026
13	<p><b>Développement piloté par les tests (suite)</b></p> <ul style="list-style-type: none"> <li>• Stratégies de tests logiciels</li> <li>• Outils TDD</li> <li>• Refactorisation</li> <li>• Critères de cessation des activités de test</li> </ul>	11 août 2026
14	<p><b>Examen final</b></p>	13 août 2026

## 6. Évaluation du cours :

Outils d'évaluation	Pondération	Indicateurs mesurés
Devoirs	15 %	4.1; 4.2; 4.3;
Examen intra	25 %	4.1; 4.2
Examen final	40 %	4.1; 4.2; 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 acceptable	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 insuffisantes	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](#)
- [Règlement concernant le plagiat et la fraude](#)
- [Politique linguistique](#)
- [Procédure en cas d'absence aux évaluations](#) : [Cadre de gestion](#) | [Formulaire d'absence](#)

Tolérance **ZÉRO** en matière de violence à caractère sexuel.

Le Bureau d'intervention et de prévention en matière de harcèlement (BIPH) a pour mission d'accueillir, soutenir et guider toute personne vivant une situation de harcèlement, de discrimination ou de violence à caractère sexuel. Le BIPH oriente ses actions afin de prévenir les violences à caractère sexuel pour que nous puissions étudier, travailler et s'épanouir dans un milieu sain et sécuritaire.

Vous vivez ou êtes une personne témoin d'une situation de violence à caractère sexuel ? Vous êtes une personne membre de la communauté étudiante ou une personne membre du personnel, autant à Gatineau qu'à Ripon et St-Jérôme, l'équipe du BIPH est là pour vous, sans jugement et en toute confidentialité.

Ensemble, participons à une culture de respect.

Pour de plus amples renseignements consultez [UQO.ca/biph](http://UQO.ca/biph) ou écrivez-nous au [Biph@uqo.ca](mailto:Biph@uqo.ca)

## 8. Principales références :

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

## 9. Page Web du cours :

<https://moodle.uqo.ca>