

## Programme court de deuxième cycle en génie logiciel

**Téléphone :** 514 987-0437  
**Courriel :** cyclessupgenielogiciel@uqam.ca

Code	Titre	Crédits
0822	Programme court de deuxième cycle en génie logiciel	9

<b>Contingent</b>	Programme non contingenté
<b>Régime et durée des études</b>	Temps complet : 1 trimestre Temps partiel : 3 trimestres ou moins
<b>Campus</b>	Campus de Montréal

### OBJECTIFS

Le programme court de deuxième cycle en génie logiciel a pour objectif général de former ponctuellement des personnes en génie logiciel en exercice dans l'industrie ou des candidats du 1er cycle désirant avoir une formation initiale en génie logiciel. Les étudiants pourront concevoir un cheminement personnalisé afin d'acquérir des connaissances théoriques et pratiques dans des domaines de pointe du génie logiciel. Les étudiants formés pourront mettre en pratique les connaissances acquises afin d'améliorer leur performance dans l'industrie. Les finissants du programme pourront poursuivre leur formation au sein du DESS ou de la maîtrise en génie logiciel.

Les objectifs spécifiques du programme court de deuxième cycle en génie logiciel sont :

- L'approfondissement de connaissances : l'étudiant pourra acquérir et approfondir les connaissances essentielles sur les différents aspects du processus de développement de logiciel, plus particulièrement les exigences, la conception, la réalisation, la maintenance et l'assurance qualité de logiciels.
- Le développement d'habiletés personnelles : l'étudiant développera des habiletés d'acquisition d'information, d'analyse, de réflexion et de synthèse sur des situations concrètes dans une conduite d'intégrité et d'éthique; il développera aussi des habiletés au travail d'équipe.

### PARTENARIAT

Ce programme est offert en partenariat avec l'École de technologie supérieure.

### CONDITIONS D'ADMISSION

Le candidat doit :

être titulaire d'un baccalauréat, ou l'équivalent, en informatique, en génie dans un domaine approprié, ou en sciences appliquées avec une composante en informatique, obtenu avec une moyenne cumulative d'au moins 3,0 sur 4,3 ou l'équivalent;

OU être titulaire d'un baccalauréat ou l'équivalent, dans un autre domaine, obtenu avec une moyenne d'au moins 3,0 sur 4,3 et posséder les connaissances ou l'expérience jugées suffisantes en informatique, en développement de logiciels ou en technologie de l'information;

Tout dossier de candidature avec une moyenne inférieure à 3,0 mais

supérieure à 2,7 sur 4,3 sera étudié par le sous-comité d'admission et d'évaluation du programme et pourrait, dans certains cas, faire l'objet d'une recommandation d'admission;

OU posséder les connaissances nécessaires, une formation appropriée et une expérience jugée pertinente d'au moins de cinq ans.

Les dossiers de candidats détenteurs d'un tel baccalauréat obtenu avec une moyenne inférieure à 2,7 sur 4,3, mais égale ou supérieure à 2,5 sur 4,3 (ou l'équivalent) seront étudiés par le sous-comité d'admission et d'évaluation, à la condition de posséder une formation additionnelle et appropriée d'au moins 15 crédits universitaires (ou l'équivalent) complétés avec une moyenne cumulative d'au moins 3,2 sur 4,3 (ou l'équivalent).

#### Connaissance du français

Le candidat doit posséder une connaissance suffisante de la langue française orale et écrite.

Sur étude de dossier par le sous-comité d'admission et d'évaluation, le candidat pourrait se voir imposer de passer un test de classement en français.

S'il n'a pas acquis le seuil établi par le programme, le candidat devra réussir une formation en français dès sa première inscription au programme. Un ou des cours pourront être suivis parallèlement à la scolarité régulière du programme selon les possibilités. L'étudiant devra avoir démontré une maîtrise suffisante du français avant la fin de sa première année d'inscription au programme.

#### Connaissance de l'anglais

Le candidat doit posséder la capacité de lire des textes scientifiques et techniques rédigés en anglais.

#### Méthode et critères de sélection

Évaluation du dossier académique, de la lettre de motivation du candidat et des lettres de recommandation.

Les candidats présentant une demande sur la base de l'expérience pertinente doivent détailler leurs expériences en précisant les rôles et responsabilités exacts qu'ils ont cumulés. Un curriculum vitæ complet doit accompagner la demande d'admission. Les candidats peuvent être convoqués à une entrevue.

#### Procédure

Le formulaire de demande d'admission complété doit être transmis au registrariat, accompagné d'une lettre de motivation précisant les objectifs du candidat face au programme, de ses relevés de notes officiels pour toutes les études de niveau universitaire (sauf celles

réalisées à l'UQAM), d'un curriculum vitae et d'attestations d'expérience s'il y a lieu.

### Régime et durée des études

Temps complet : un trimestre

Temps partiel : trois trimestres ou moins (1 an ou moins)

## COURS À SUIVRE

(Sauf indication contraire, les cours comportent 3 crédits. Certains cours ont des préalables. Consultez la description des cours pour les connaître.)

L'étudiant pourra choisir trois cours de la liste ci-dessous, constituée d'activités offertes à la maîtrise en génie logiciel. Le choix de cours et le cheminement devront faire l'objet d'un accord préalable avec le directeur du programme.

### Trois cours choisis dans la liste suivante (9 crédits) :

- MGL7130 Développement d'applications mobiles
- MGL7230 Tests logiciels
- MGL7240 Mesures et développement de logiciel
- MGL7250 Processus de développement Agile
- MGL7260 Exigences et spécifications de systèmes logiciels
- MGL7315 Gestion de projet en génie logiciel
- MGL7320 Ingénierie logicielle des systèmes d'intelligence artificielle
- MGL7361 Principes et applications de la conception de logiciels
- MGL7460 Réalisation et maintenance de logiciels
- MGL7560 Vérification et assurance qualité de logiciels
- MGL7760 Qualité et productivité des outils logiciels
- MGL7810 Sujets spéciaux en génie logiciel I
- MGL7811 Sujets spéciaux en génie logiciel II
- MGL7815 Lecture dirigée I
- INF7210 Nouvelles perspectives en bases de données
- INF7235 Programmation parallèle haute performance
- INF7270 Écosystème du logiciel libre
- INF8750 Sécurité des systèmes informatiques

Remarque : Le candidat vaudra bien prendre note que les activités au choix dans ce programme et énumérées ci-dessus ne peuvent pas être offertes à chacun des trimestres (automne, hiver ou été). Par conséquent, elles sont réparties sur plusieurs trimestres et sont donc offertes en alternance d'un trimestre, voire d'une année à l'autre. De plus, la majorité des cours sont offerts en soirée.

## PASSERELLES

Un étudiant désirant continuer sa formation au DESS ou à la maîtrise en génie logiciel se verra reconnaître tous les cours réussis dans le cadre du programme court de 2e cycle en génie logiciel.

## DESCRIPTION DES COURS

### INF7210 Nouvelles perspectives en bases de données

Concepts avancés des bases de données. Gestion de transactions. Contrôle et optimisation des performances. Bases de données parallèles et réparties. Développement d'applications de bases de données Web et multitières. Bases de données objet et objet-relationnel. Gestion de données semi-structurées et multimédia. Entrepôts de données et analyse de données (OLAP). Fouille de données (data mining). Bases de données déductives. Repérage de l'information.

### INF7235 Programmation parallèle haute performance

Modèles d'architectures à haute performance. Paradigmes de programmation parallèle et stratégies de conception de programmes parallèles. Métriques de performances et principales sources des surcoûts. Langages et bibliothèques de programmation parallèle. Problèmes typiques en programmation scientifique haute performance: calculs de grilles, de particules, de matrices.

### INF7270 Écosystème du logiciel libre

### Objectifs

Ce cours vise à présenter aux étudiants la problématique du logiciel libre dans les organisations. Au terme du cours, les étudiants devraient être capables d'évaluer le potentiel des logiciels libres dans la stratégie des TI ; d'en évaluer les avantages et les inconvénients ; d'identifier les applications où le logiciel libre peut représenter des bénéfices importants ; de comprendre la dynamique des communautés de développeurs de logiciel libre ; de concevoir des approches pour effectuer une veille technologique du secteur ; de s'initier à la mise sur pied d'un projet de logiciel libre ; de saisir les principaux enjeux relatifs aux licences libres ; d'établir une stratégie adaptée à chaque organisation relativement au logiciel libre y compris la sensibilisation de la haute direction à ce mouvement.

### Sommaire du contenu

Définition du logiciel libre selon la Free Software Foundation (FSF) et l'Open Software Initiative (OSI) ; identification des principales licences libres ; problématiques particulières des logiciels applicatifs ; principaux domaines où le logiciel libre fait des percées importantes ; études de cas (Apache, Mozilla, Eclipse) ; les forges et leur fonctionnement ; le logiciel libre comme modèle d'affaires : Alfresco, MongoDB, etc.; la dynamique des projets libres ; l'étude de faisabilité d'un projet libre ; problématique particulière des logiciels applicatifs ; motivation des participants à des projets libres : revue des principales études sur ce sujet (ex. FLOSS de la Commission européenne) ; le logiciel libre comme outil de développement économique (ex. rapport de l'OCDE) ; élaboration d'un plan stratégique incluant le libre et plan de communication destiné à la direction ; enfin principaux inconvénients du libre.

### Préalables académiques

INF7115 Bases de données ou INF7215 Analyse et conception des systèmes d'information de l'entreprise ou l'équivalent

### INF8750 Sécurité des systèmes informatiques

Principes et concepts fondamentaux de la sécurité des systèmes informatiques. Principaux services: confidentialité, intégrité, disponibilité, authentification, non répudiation, contrôle d'accès. Typologie des attaques: fuites, modifications d'information, privations de service. Mécanismes sécuritaires modernes: systèmes de chiffrement symétriques et asymétriques; fonctions de hachage; génération pseudo-aléatoire. Protocoles sécuritaires: authentification, signature, échange et gestion de clés. Sécurité des systèmes centralisés et des systèmes répartis: politiques et modèles de sécurité; contrôle d'accès; rôles et privilèges. Sécurité des programmes: virus, chevaux de Troie. Contre-mesures: journalisation, audits; détection d'intrusion; filtrage; mécanismes de recouvrement. Analyse de risque. Éducation des usagers. Considérations légales, politiques et éthiques.

### MGL7130 Développement d'applications mobiles

#### Objectifs

À l'issue de ce cours, l'étudiant sera en mesure de développer des applications mobiles dans des environnements modernes.

#### Sommaire du contenu

Architecture et fonctionnalités de base des plate-formes mobiles; environnement de développement des plate-formes mobiles; structure et composants fondamentaux des applications mobiles ; construction de l'interface utilisateur ; utilisation des ressources : XML, images, fichiers, etc. ; persistance des données ; intégration ; signature d'une application ; déploiement et contraintes particulières des applications mobiles. API pour le développement des applications mobiles dépendantes de leur contexte. Interactions avec les outils de captage de données à travers les appareils et les réseaux mobile. Développement d'applications de commerce électronique mobile. Tendances et perspectives futures.

#### Conditions d'accès

Connaissance d'un langage de programmation orienté-objet des technologies du Web.

### MGL7230 Tests logiciels

**Objectifs**

À la fin du cours, l'étudiant sera en mesure de réaliser la planification, la conception et le suivi de toutes les activités de tests d'un projet; de développer et accomplir efficacement les différentes phases de tests;

**Sommaire du contenu**

Tests logiciels dans le cycle de vie du logiciel. Normes relatives aux tests. Processus des tests logiciels. Les différents niveaux de tests. Conception des cas de tests. Efficacité des tests. Stratégies de tests pour les environnements modernes : infonuagique, applications Web, applications mobiles, applications multi-couches. Tests et méthodologies agiles. Principaux cadres pour les tests. Automatisation des tests.

**MGL7240 Mesures et développement de logiciel****Objectifs**

Ce cours vise à développer les connaissances et habiletés de l'étudiant à définir, à déployer et à faire évoluer un plan de mesure afin de soutenir les processus de décision et de gestion du logiciel (estimation, planification, suivi, déploiement, exploitation et mise au rancart). À la fin du cours, l'étudiant sera en mesure de décrire les concepts de base de la mesure et d'élaborer et de déployer les divers éléments d'un plan de mesure adapté au contexte d'une entreprise afin de répondre aux besoins en information du personnel technique et des gestionnaires. Il connaîtra aussi les principales normes applicables.

**Sommaire du contenu**

Concepts de base de la mesure. Normes, modèles et méthodes liés aux principes, aux processus et aux pratiques de la mesure en génie logiciel. Processus type de la mesure. Conception et application d'un plan de mesure. Présentation des résultats pour faciliter la prise de décision. Mesures liées aux projets, aux processus, aux applications et à la gouvernance des TI. Mesure de la taille fonctionnelle des logiciels et ses usages: productivité des processus de développement et de maintenance, estimation de projet, étalonnage (benchmarking), gouvernance du portfolio applicatif et amélioration de la qualité et de la productivité du processus. Aspects économiques de la mesure. Rôle de la mesure dans les différents modèles d'affaires de sous-traitance du développement logiciel. Défis organisationnels liés à la mesure. Outils de mesure.

**MGL7250 Processus de développement Agile****Objectifs**

Ce cours vise à développer les connaissances et habiletés de l'étudiant à comprendre et appliquer les différents principes et méthodes dites Agile. A la fin du cours, l'étudiant sera apte à mettre en place un processus de développement agile au sein d'un groupe de développement logiciel

**Sommaire du contenu**

Valeurs et principes de l'Agilité. Méthodes : Scrum, TDD, XP, Crystal, DDD, DSDM. Démarrage d'un projet Agile. Architecture et livraison incrémentale. Gestion de projet; déploiement. Impact de l'Agilité sur les équipes d'infrastructure technologique. Entretien et évolution de logiciel. Culture organisationnelle et gouvernance. Évolution du processus logiciel. Soutien au développement. Agilité et documentation. Mesures de performance organisationnelle liées à l'adoption de l'Agilité. Modèles de bonnes pratiques du développement logiciel. Impact de l'adoption de l'Agilité sur les individus.

**MGL7260 Exigences et spécifications de systèmes logiciels**

Introduction à l'ingénierie des systèmes. - Modèles de processus des exigences logicielles. - Intervenants dans le processus des exigences logicielles. - Support et gestion du processus des exigences logicielles. - Qualité et amélioration du processus des exigences logicielles. - Sources des exigences logicielles. - Techniques d'explicitation des exigences logicielles. - Classification des exigences logicielles. - Modélisation conceptuelle. - Conception architecturale et allocation des exigences logicielles. - Négociation des exigences logicielles. - Document de définition des exigences logicielles. - Document de spécification des exigences logicielles. - Structure et normes de documentation des exigences logicielles. - Qualité de la documentation

des exigences logicielles. - Revue des exigences logicielles. - Prototypage. - Validation des modèles. - Tests d'acceptation. - Gestion des changements des exigences logicielles. - Attributs des exigences logicielles. - Trace des exigences logicielles. - Sujets avancés en exigences logicielles.

**MGL7315 Gestion de projet en génie logiciel**

Principes et gestion de projet de génie logiciel. Gestion de l'ingénierie des exigences, gestion de l'ingénierie du design, de l'ingénierie de la construction du code, des stratégies d'essais, de la maintenance et de l'évolution des logiciels. Principes et techniques de gestion spécifiques au développement de projets de génie logiciel, incluant la mesure et l'estimation, l'amélioration des processus, l'ingénierie de la qualité, les outils de soutien au développement et la gestion de configuration. Application des normes d'ingénierie du logiciel (incluant les normes ISO, IEEE et les normes industrielles) pour la planification, l'encadrement et la réalisation de projets de génie logiciel.

**MGL7320 Ingénierie logicielle des systèmes d'intelligence artificielle**

Ce cours vise à fournir les concepts clés liés à l'ingénierie des systèmes d'intelligence artificielle.

**Objectifs**

L'objectif du cours est de fournir aux étudiants gradués les connaissances techniques de base et fondamentales tout en appliquant les concepts et processus de génie logiciel (GL) dans le contexte d'un système d'intelligence artificielle (IA). À la fin de ce cours, les étudiants seront capables d'identifier les enjeux liés aux systèmes d'intelligence artificielle et de proposer un ensemble de solutions pour spécifier, développer et maintenir les systèmes d'intelligence artificielle. Le cours se concentre sur l'interconnexion du GL et de l'IA, et sur la manière dont les sujets couverts s'appliquent aux systèmes basés sur l'IA.

**Sommaire du contenu**

Les sujets inclus sont la spécification et l'architecture des systèmes d'intelligence artificielle, la validation et la gestion des données, le processus de sélection des modèles, les tests ainsi que le déploiement des systèmes d'intelligence artificielle. Les sujets spéciaux sur l'interprétation, l'équité et les opérations des systèmes d'intelligence artificielle peuvent aussi être inclus.

**Compétences professionnelles en enseignement**

Diplôme de 3e cycle en génie logiciel spécialisé en IA; Diplôme de 2e cycle en génie logiciel et minimum de 3 ans d'expérience en développement de système d'IA.

**Modalité d'enseignement**

Cours magistral. Les étudiants seront évalués sur un projet de recherche en IA, des critiques d'articles et activités en IA, un examen et leur participation en classe.

**Conditions d'accès**

Connaitre le langage de programmation Python. Être capable de lire et comprendre des articles scientifiques en anglais. Avoir des connaissances de base en conception de logiciels.

**MGL7361 Principes et applications de la conception de logiciels**

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.

**MGL7460 Réalisation et maintenance de logiciels**

Rôle de la réalisation et de la maintenance dans le cycle de vie du logiciel. Évolution et maintenance du logiciel. Méthodes propres à étendre la durée de vie. Sélection de la méthode appropriée de réalisation. Prototypage. Mise au point. Gestion de la maintenance. Réutilisation et rétro-ingénierie des logiciels. L'interaction entre réalisation et maintenance sera traitée tout au long du cours.

**MGL7560 Vérification et assurance qualité de logiciels**

But et concepts de qualité des logiciels. Facteurs qualité (efficacité, exactitude, performance, facilité d'entretien). Normes d'assurance qualité et de vérification et validation (ISO, IEEE). Plans d'assurance qualité et de vérification et validation (coût, activités, ressources). Méthodes d'assurance qualité et de vérification et validation (revues, inspections, audits). Les tests: principes, méthodes, processus et plan de tests. Outils logiciels facilitant la mise en oeuvre de l'assurance qualité, de la vérification et validation de logiciels et des tests.

**MGL7760 Qualité et productivité des outils logiciels**

Aperçu des outils pour assister au développement et à l'entretien des logiciels. Plates-formes d'intégration des outils. Environnements de développement. Outils pour la rétro-ingénierie des logiciels. Critères et stratégies d'évaluation des outils.

**MGL7810 Sujets spéciaux en génie logiciel I**

## Objectifs

Présenter et diffuser aux étudiants des sujets de pointe ou en émergence dans le domaine du génie logiciel.

## Sommaire du contenu

Présentation de sujets d'intérêt majeur dans le domaine du génie logiciel et familiarisation avec les derniers développements technologiques dans un ou plusieurs domaines de pointe ou en émergence. Les sujets spéciaux seront déterminés avant les inscriptions de chaque trimestre.

**MGL7811 Sujets spéciaux en génie logiciel II**

## Objectifs

Présenter et diffuser aux étudiants des sujets de pointe ou en émergence dans le domaine du génie logiciel.

## Sommaire du contenu

Présentation de sujets d'intérêt majeur dans le domaine du génie logiciel et familiarisation avec les derniers développements technologiques dans un ou plusieurs domaines de pointe ou en émergence. Les sujets spéciaux seront déterminés avant les inscriptions de chaque trimestre.

**MGL7815 Lecture dirigée I**

## Objectifs

Cette activité a pour objectif de permettre à l'étudiant d'approfondir un thème relatif au génie logiciel. Elle s'inscrit dans une démarche personnalisée de formation à l'intérieur du programme de maîtrise en génie logiciel.

## Sommaire du contenu

Lectures approfondies sous la supervision d'un professeur dans un sujet relié à son domaine de spécialisation. Compléter la formation en approfondissant ou en diversifiant ses connaissances tout en développant son sens critique et son esprit de synthèse. Cette activité requiert une grande autonomie de la part de l'étudiant. Le choix des lectures et le suivi de l'étudiant sont sous la responsabilité du professeur. Des rencontres périodiques ont lieu tout au long de cette activité. Un rapport dont l'objectif, la nature et l'ampleur sont définis par le professeur doit être rédigé à la fin de l'activité. Le rapport doit être remis à la fin du trimestre de l'inscription.

## Conditions d'accès

L'inscription à ce cours requiert l'autorisation préalable du directeur du programme.

N.B. : Le masculin désigne à la fois les hommes et les femmes sans aucune discrimination et dans le seul but d'alléger le texte.  
Cet imprimé est publié par le Registrariat. Basé sur les renseignements disponibles le 21/03/23, son contenu est sujet à changement sans préavis.  
Version Hiver 2023