

## Maîtrise en génie logiciel

### INF 743 : Architecture logicielle

#### Plan de cours

Session : A-2023

Chargé de cours : Sylvain Picard  
(Sylvain.Picard2@Usherbrooke.ca)

#### 1. Mise en contexte

Le spécialiste en architecture logicielle a la responsabilité de créer ou de sélectionner le *plan* le plus approprié pour concevoir des solutions mettant en relation plusieurs systèmes et intervenants au sein d'une entreprise. Cet informaticien de haut niveau est une personne clé dans tout développement logiciel puisqu'il est entre autres responsable de s'assurer que le *plan* d'architecture conçu satisfait les besoins d'affaires, qu'il conforme aux exigences et contraintes des parties prenantes et qu'il soit évolutif.

Au Québec comme ailleurs, la pénurie de spécialistes en architecture logicielle est un frein au développement de systèmes. Les grandes entreprises, tout comme les gouvernements, cherchent à mettre la main sur des personnes qui possèdent les compétences nécessaires à la réalisation d'architectures logicielles. La plupart du temps, ces organisations sont obligées de s'en remettre à quelques firmes informatiques qui prétendent être capables de combler ce besoin. Devoir confier la planification et la conception des architectures logicielles à des firmes externes fragilise les organisations et les rend dépendantes de ces firmes.

Ce cours s'inscrit donc dans un contexte où il y a un besoin de formation pour permettre aux organisations de faire face aux défis des développements logiciels actuels et futurs.

#### 2. Place du cours dans le programme

Ce cours s'inscrit notamment dans la maîtrise en génie logiciel. Les spécialistes en architecture logicielle interviennent à plusieurs stades dans le développement des logiciels de l'entreprise. Le regard qu'ils portent sur le développement logiciel peut être large. Il peut par exemple s'appliquer à l'ensemble du portfolio applicatif, dans le cas de l'architecte d'entreprise. Il peut aussi être plus spécifique et détaillé à un système en se préoccupant uniquement d'une application, voire d'un composant logiciel. Il existe un type de rôle qui intervient à un niveau

intermédiaire entre ces deux pôles; on l'appelle généralement l'architecture de solutions. Ce cours vise à former un tel spécialiste.

Étant un cadre d'architecture couramment utilisé (ou inspiré) dans l'industrie, [TOGAF](#) (The Open Group Architecture Framework) ainsi que le langage de modélisation [Archimate](#) seront introduits comme référence durant le cours.

### 3. Objectifs généraux

Cette activité pédagogique vise à développer chez l'étudiante ou l'étudiant la capacité de :

- O1. Définir une nouvelle architecture logicielle en fonction de l'analyse de besoins ;
- O2. Adapter une architecture logicielle existante à l'évolution des besoins ;
- O3. Migrer d'un système ou d'une architecture à un autre

### 4. Objectifs spécifiques

À la fin de cette activité pédagogique, l'étudiante ou l'étudiant sera capable de :

- OS1. Décrire les principales familles d'architectures logicielles ;
- OS2. Identifier les principaux attributs de qualité de ces familles en lien avec les principales normes en vigueur dans le domaine (ISO/IEC 9126, ISO/IEC 25000) ;
- OS3. Sélectionner l'architecture appropriée à une situation donnée ;
- OS4. Identifier les caractéristiques et les contraintes liées à une architecture existante ;
- OS5. Modifier une architecture existante en tenant compte de ses contraintes intrinsèques et de l'évolution des besoins ;
- OS6. Planifier la migration d'une architecture à une autre ;
- OS7. Procéder à la migration d'une architecture à une autre, tout en facilitant la coexistence, puis la transition entre l'ancienne architecture et la nouvelle.

### 5. Planification hebdomadaire

#### Séance #1 - Présentation du cours et des concepts clés (28 août)

- Présentation du chargé de cours et des étudiants
- Présentation du plan de cours
- Définitions et mise en contexte de l'architecture applicative dans l'architecture d'entreprise

#### Séance #2 – Introduction à l'architecture logicielle (11 septembre)

- Fondements et objectifs globaux
- Outils de l'architecte de solution
- Préparation des équipes et explication sommaire du TP

#### Séance #3 – Construction d'une architecture logicielle (18 septembre)

- Règles du jeu
- Principes d'architecture
- Quelques patrons d'architecture

#### **Séance #4 – Les vues et perspectives d’une architecture de solutions (25 septembre)**

- Modélisation, formalisme et représentation
- Les principales vues d’architecture (4+1) pour documenter architecture de solution
- Perspectives (viewpoints) d’une architecture de solution
- Exigences fonctionnelles, non fonctionnelles et opérationnelles
- Quiz #1 (séances 1, 2 & 3 - 60 min.)

#### **Séance #5 – Principes d’architecture logicielle appliqués et construction d’une architecture de solution (2 octobre)**

- Contraintes dans l’élaboration de l’architecture
- Développement d’une architecture de solution
- Fondations d’une architecture
- Décisions d’architecture
- Évaluation des impacts d’une architecture (technologie, personnel et processus)

#### **Séance #6 – Approches d’architecture pour l’évolution du portfolio applicatif (16 octobre)**

- Les logiciels commerciaux (COTS et « buy or build »)
- Stratégie d’évolution d’applications en place
- TIME de Gartner (Tolerate, Invest, Migrate, Eliminate) et simplification portfolio applicatif
- Infonuagique, LA solution pour évolution des applications?
- Questions / réponses avant la 1<sup>re</sup> présentation du TP

#### **Séance #7 – 1<sup>re</sup> du TP (23 octobre)**

- Rencontre avec le comité d’architecture et le client (représentés par le chargé de cours) pour validation de la 1<sup>re</sup> itération de l’architecture cible proposée (15 à 20 minutes selon le nombre d’équipe). Le client sera disponible pour répondre aux questions de clarification
- Commentaires et précisions pour la suite du TP

#### **Séance #8 – Architecture d’intégration (30 octobre)**

- Différentes approches d’intégration du portail aux APIs en passant par RPA...
- Patrons d’intégration interapplication
- Architecture orientée service et microservices
- Architecture d’un API (Application Programming Interfaces)
- Quiz #2 (séances 4, 5, 6 & 7 - 60 min.)

#### **Séance #9 – Architecture de données et protection de l’information (6 novembre)**

- Gestion des données opérationnelles / transactionnelles
- Gestion des données pour des fins analytiques (tableau de bord et indicateur)
- Gestion des données pour des fins d’analytiques prédictives via l’intelligence artificielle
- Protection de l’information sensible

#### **Séance #10 – Processus de développement logiciel (Software Development LifeCycle) (13 novembre)**

- Principes des processus de tous les processus de développement logiciel

- Pratiques d'un processus de développement logiciel et interaction de l'architecture
- Agile, CMMI et Macroscopie
- \*Ops (dev, data, etc.) les principes et critères de succès
- Quelques mots sur la gestion de projets
- Planification de l'évolution d'une architecture (feuille de route)

#### **Séance #11 – Gouvernance de l'architecture (20 novembre)**

- Processus de gouvernance de l'architecture
- Architecture d'entreprise, un allié
- Comité de révision d'architecture (Architecture Review Board)
- Architecture Agile (just in time / just enough architecture)
- Architecture de solution, une approche de gestion de risques ?

#### **Séance #12 – 2e présentation TP et quiz #3 (27 novembre)**

- Présentation du plan de migration du TP (15 à 20 min. max selon le nombre d'équipes) au client avant la présentation finale pour approbation
- Quiz #3 (séances 8, 9, 10 & 11 - 60 min.)

#### **Séance #13 – Présentation finale du TP (4 décembre)**

- Présentations de 30 minutes de la solution finale au client et comité d'architecture
- Révision pour l'examen final – période de questions

#### **Séance #14 – Examen final récapitulatif (11 décembre)**

## **6. Approche pédagogique préconisée**

Cours magistraux, discussions. Un travail d'équipe pratique sur des scénarios. Présentation au groupe et documentation.

Il y aura 2 conférenciers invités pour partager leur parcours, leurs défis et les opportunités de l'architecture de solution qu'ils vivent dans leurs organisations respectives. Le choix des conférenciers sera fait en fonction du niveau de connaissances et le vécu des participants au cours.

La participation active des élèves aux discussions sur les thèmes abordés est attendue et sera encouragée.

## **7. Évaluation de l'apprentissage**

<b>Modalités d'évaluation</b>	<b>%</b>
Trois quiz de 10% chacun	30 %
Travail pratique en équipe (15% présentation, 25% documentation)	40 %
Examen final	30 %

## **8. Travail pratique - contexte**

Vous êtes l'équipe d'architecture assignée à l'évolution d'une solution commerciale de dossier médical électronique. Votre équipe devra développer les vues d'architecture et artefacts pertinents. Un ensemble de documents permettra à l'équipe d'architecture d'avoir

l'information sur la solution commerciale, les ajouts souhaités (fonctionnels et qualité) pour en définir la cible et un programme d'implantation.

Les premiers documents pertinents seront disponibles dès la séance #2 (11 septembre) durant laquelle les équipes de 2 à 3 personnes seront formées (selon la taille de la classe). Comme indiqué dans le plan de cours, il y aura quelques présentations intermédiaires au client (représenté par le chargé de cours).

Lors de la séance #13 (4 décembre), vous devrez présenter en équipe, l'architecture au client et au comité d'architecture (représenté par le chargé de cours). Comme dans la vraie vie, la solution proposée sera importante, mais aussi la façon et la clarté de la présentation au client. Tous les membres de l'équipe devront participer activement à la présentation – une évaluation commune sera octroyée pour la portion documentation, tandis qu'une évaluation individuelle pour la présentation. Le comité d'architecture et le client auront des questions lors de la présentation. Un gabarit de présentation avec les principales sections sera proposé.

Les présentations du TP seront faites équipe par équipe qu'au chargé de cours. Une séquence et un horaire seront convenus d'avance et communiqués la semaine avant la présentation. Lors des séances intermédiaires, le chargé de cours donnera, au besoin, quelques conseils ou orientations pour la suite du TO.

## **9. Plagiat**

Un document dont le texte ou la structure se rapporte à des textes intégraux tirés d'un livre, d'une publication scientifique ou même d'un site Internet doit être référencé adéquatement. Lors de la correction de tout travail individuel ou de groupe, une attention spéciale sera portée au plagiat, défini dans le Règlement des études comme « le fait, dans une activité pédagogique évaluée, de faire passer indûment pour siens des passages ou des idées tirés de l'œuvre d'autrui. » Le cas échéant, le plagiat est un délit qui contrevient à l'article 8.1.2 du Règlement des études : « tout acte ou manœuvre visant à tromper quant au rendement scolaire ou quant à la réussite d'une exigence relative à une activité pédagogique. ». À titre de sanction disciplinaire, les mesures suivantes peuvent être imposées : a) l'obligation de reprendre un travail, un examen ou une activité pédagogique et b) l'attribution de la note E ou de la note 0 pour un travail, un examen ou une activité évaluée. Tout travail suspecté de plagiat sera transmis à la vice-doyenne à l'enseignement de la Faculté des sciences.

## **10. Remise des travaux**

Tous les travaux doivent être remis dans un format PDF, clairement identifiés tant au niveau de l'auteur ou des auteurs que de la nature du travail, ainsi qu'être remis de façon électronique le sera par Moodle.

## **11. Communication**

Pour les questions ou discussions communes d'intérêt pour tout le groupe (matière présentée, référence, TP, etc.), le Forum Moodle sera utilisé afin de facilement partager l'information.

Cependant, pour des sujets liés à votre parcours individuel ou difficulté particulière, vous pouvez utiliser mon courriel de l'Université (sylvain.picard2@usherbrooke.ca).

## 12. Bibliographie

The Open Group, TOGAF 9, <http://www.opengroup.org/togaf/>

International Organization for Standardization, ISO/IEC 9126, [www.iso.org](http://www.iso.org)

International Organization for Standardization, ISO/IEC 25000, [www.iso.org](http://www.iso.org)

Software Engineering Institute du Carnegie Mellon University <http://www.sei.cmu.edu/>

[D'autres sources de références pertinentes seront proposées durant le cours](#)