

# Plan de cours



Activité :	INF 755 – Méthodes d'analyse et de conception
Trimestre :	Hiver 2015
Enseignant :	Luc Lavoie

## 1. Mise en contexte

Le génie logiciel traite de la configuration d'une machine universelle (ordinateur) dans le but d'atteindre un objectif spécifique. Le logiciel de configuration peut lui-même être vu comme une machine, mais il diffère d'autres machines en ce sens qu'il est intangible. Le génie logiciel (GL) doit son nom et sa constitution comme un domaine de connaissance propre à la tenue d'un séminaire organisé par l'OTAN à Garmisch-Partenkirchen en Allemagne en 1968.

Le logiciel de configuration d'une machine universelle est désigné sous plusieurs appellations différentes, selon la caractéristique mise de l'avant : logiciel (intangibilité), programme (déterminisme), système (complexité).

Puisqu'on construit généralement un système pour atteindre un objectif donné, il est préférable de déterminer et de détailler d'abord cet objectif et le but auquel l'objectif concourt. Ce qui nous amène à l'ingénierie des exigences (IE), la partie du génie logiciel qui permet de déterminer quel système doit être développé. Plusieurs études tendent à montrer que l'échec d'un projet de développement de système est souvent imputable à des lacunes d'IE. Par ailleurs, une erreur d'IE a généralement plus d'impact qu'une erreur commise à une autre étape du développement du système et ce, que l'impact soit mesuré en terme de cout, de durée, de portée ou de qualité.

La maîtrise du processus de développement logiciel passe par celle de l'IE et intéresse toutes les organisations tributaires de systèmes comprenant des logiciels.

Note : L'expression « spécification des exigences » est parfois utilisée pour désigner l'ingénierie des exigences dans son ensemble; nous préférons réserver cette expression pour désigner une activité précise au sein du processus d'ingénierie des exigences, à savoir l'activité par laquelle on déduit (on conçoit) et met en forme, selon des critères rigoureux, les exigences du logiciel, exigences issues des activités préalables d'exploration et d'analyse.

## 2. Place de l'activité dans le programme

L'activité est destinée aux étudiants du diplôme en technologies de l'information et contribue plus particulièrement à six des neuf objectifs du programme, à savoir :

- analyser et spécifier des besoins en matière de technologies de l'information (TI);
- évaluer les qualités d'un système d'information (SI);
- contribuer au développement et à la maintenance d'un SI;
- superviser et améliorer un SI;
- déterminer des politiques, des normes et des procédures pour les SI;
- assurer le contrôle et la vérification d'un SI.

L'activité INF 755 doit suivre INF 736 « Modèle de connaissance et évolution en TI » et précéder INF 754 « Gestion de projets ». Il est éminemment souhaitable qu'elle soit concomitante ou qu'elle succède à INF 731 « Programmation orientée objet » et INF 732 « Bases de données ».

### 3. Description de l'activité

#### *Objectifs*

1. Comprendre le rôle de l'analyse des besoins dans les contextes du développement et de la maintenance de logiciels.
2. Pouvoir synthétiser une démarche d'analyse de façon cohérente et réfléchie.
3. Se familiariser avec certains des outils de modélisation les plus utilisés en entreprise.
4. Savoir reconnaître et utiliser les principaux éléments de la notation UML.

#### *Contenu*

- A. Processus d'analyse et de conception.
- B. Exploration, identification des besoins, techniques d'acquisition de l'information.
- C. Modélisation.
- D. Analyse, spécification et gestion des besoins.
- E. Stratégies et méthodes.
- F. Contexte et facteurs externes à considérer.
- G. Diagrammes de modélisation (entité-association, DFD, UML, etc.).

#### *Préalables*

INF 736 Modèle de connaissance et évolution en TI.

### 4. Objectifs spécifiques

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

- établir la structure d'un document de spécification des exigences;
- appliquer des techniques d'exploration des exigences;
- appliquer des techniques d'analyse des exigences;
- appliquer des techniques de spécification des exigences;
- vérifier les exigences;
- valider des exigences.

### 5. Planification hebdomadaire

Voir Annexe.

### 6. Approche pédagogique préconisée

L'activité est développée selon deux axes : un apprentissage des fondements de l'ingénierie des exigences et la mise en pratique de cet apprentissage dans le cadre d'un projet réaliste de petite envergure échelonné tout au long de l'activité. L'apprentissage fondamental est réalisé par le biais de lectures faites par l'étudiant et de cours assurés par l'enseignant. Une approche pragmatique, fondée sur l'étude de cas concrets documentés a été retenue. Le projet se décline en sept travaux pratiques (TP) : chacun des six premiers est associé à un atelier préparatoire réalisé en classe, le dernier est la reprise, la correction et la synthèse des premiers. Un examen permet de faire la synthèse finale de l'activité.

Les étudiants seront ainsi appelés à mettre leurs connaissances en pratique et à développer un réel savoir-faire au sein d'équipes composées par l'enseignant.

***La participation aux ateliers, aux travaux pratiques et à l'examen est obligatoire de même la livraison de tous les travaux pratiques.***

## 7. Évaluation des apprentissages

La répartition des évaluations est décrite au tableau 1, les dates d'évaluations sont indiquées au calendrier en annexe.

La durée de l'examen est de trois heures. La documentation personnelle (manuscrite ou imprimée) est permise, mais l'usage d'appareils informatiques, électroniques ou de communication (ordinateur, calculatrice, téléphone, etc.) est interdit.

**Tableau 1 – Sommaire des évaluations**

Évaluation	Valeur	Commentaire
TP1 à TP6	42 %	En équipe (6 x 7 %)
TP7	8 %	En équipe
Examen	50 %	Individuel et récapitulatif
<b>Total</b>	<b>100 %</b>	

**Politique d'évaluation.** L'évaluation est faite en tenant compte que chaque document remis est clair, c'est-à-dire lisible et compréhensible; précis, c'est-à-dire exact et sans erreurs; concis, c'est-à-dire sans éléments superflus; complet, c'est-à-dire comprenant tous les éléments requis. Conformément aux articles 36, 37 et 38 du règlement facultaire d'évaluation des apprentissages<sup>1</sup>, l'enseignant peut retourner à l'étudiante ou à l'étudiant tout travail non conforme aux exigences quant à la qualité de la langue et aux normes de présentation.

**Politique de report d'échéance.** Toute modification d'une date de livraison doit avoir été acceptée par le groupe et la direction du CeFTI au moins une semaine avant l'échéance de la livraison.

**Politique de gestion des retards.** Tout étudiant, toute étudiante, qui omet de remettre un travail au moment prescrit au calendrier doit convenir d'une nouvelle date de livraison avec l'enseignant. Dans tous les cas, une pénalité de 10 % par jour de retard est imposée.

**Modification à la politique d'évaluation.** En cas de circonstances extraordinaires au-delà du contrôle de l'Université de Sherbrooke et sur décision de celle-ci, l'évaluation des apprentissages de cette activité est sujette à changement.

## 8. Plagiat

### *Dispositions générales*

Toute situation de plagiat sera traitée en conformité, entre autres, avec l'article 8.1.2 du Règlement des études<sup>2</sup> de l'Université de Sherbrooke.

### *Dispositions particulières*

Un document dont le texte et la structure se rapportent 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 indument 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

<sup>1</sup> [http://www.usherbrooke.ca/accueil/documents/politiques/pol\\_2500-008/pol\\_evaluation/sciences.html](http://www.usherbrooke.ca/accueil/documents/politiques/pol_2500-008/pol_evaluation/sciences.html)

<sup>2</sup> <http://www.usherbrooke.ca/programmes/etude>

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 au Secrétaire de la Faculté des sciences.

## 9. Remise des travaux

Le contrat d'équipe, les ateliers et les travaux pratiques font l'objet d'une livraison comportant tous les éléments requis. Les éléments, leur nature et le format de fichiers associés seront fixés par les énoncés. Chaque livraison doit être faite à la date prescrite selon le calendrier présenté en annexe. Chaque livraison doit être **préparée et présentée de manière professionnelle**; les fichiers la composant doivent être regroupés sous la forme d'une archive au format ZIP. Toutes les livraisons doivent être faites par courriel à l'enseignant **et** à la chargée de travaux pratiques<sup>3</sup> en mentionnant l'objet du message décrit par la grammaire suivante :

```
<objet> ::= <activité> "_" <livraison> "_" <émetteur>;  
<activité> ::= "INF755";  
<livraison> ::= "TP1" | ... | "TP7";  
<émetteur> ::= //lettre associée à l'équipe par l'enseignant//.
```

Par exemple : « INF755\_TP1\_A ».

Le courriel de livraison doit être fait par un membre de l'équipe en mettant tous les autres membres en copie. Ceux-ci sont considérés en accord avec la livraison à moins de le signifier par courriel à l'ensemble des parties prenantes (les enseignants et les autres membres de l'équipe) sous 24 heures. Tous les courriels utilisés doivent être ceux de l'Université de Sherbrooke (domaine USherbrooke.ca). L'archive au format zip constituant la livraison doit être jointe au courriel (changer le suffixe de .zip à .piz si l'anti-virus bloque la pièce jointe) ou soumise par l'entremise du service d'envoi de fichiers<sup>4</sup> si son poids excède 2 Mo. Le non-respect des consignes de livraison entraîne la note zéro.

## 10. Bibliographie

[Audibert2009]

Laurent AUDIBERT.

*UML 2 : De l'apprentissage à la pratique.*

Ellipses Marketing, Paris, 2009; ISBN 978-2729852696.

De larges extraits sont disponibles sous <http://laurent-audibert.developpez.com/Cours-UML/> (consulté le 2012-01-08).

[Braude2011]

Eric J. BRAUDE, Michael E. BERNSTEIN.

*Software engineering : modern approaches.*

2<sup>nd</sup> édition, John Wiley & sons, 2011, ISBN 978-0-471-69208-9; [UdeS QA 76.758 B743, 2011].

[Bray2002]

Ian K. BRAY.

*An Introduction to requirements engineering.*

Addison-Wesley, 2002; ISBN 0-201-76792-9; [UdeS QA 76.758 B744, 2002].

<sup>3</sup> Luc.Lavoie@USherbrooke.ca, Christina.Khnaissier@USherbrooke.ca

<sup>4</sup> <https://www.usherbrooke.ca/envoi-de-fichiers/>

- [Contantinidis2011]  
Yves CONTANTINIDIS.  
*Expression des besoins pour le système d'information.*  
Eyrolles, 2011. ISBN 978-2-212-12783-6.
- [Davis2007]  
M. DAVIS.  
*Requirements Bibliography.*  
<http://web.uccs.edu/adavis/UCCS/reqbib.htm> (consulté le 2007-03-15).
- [De Lucia2008]  
Andrea DE LUCIA, Filomena FERRUCI, Genny TORTORA, Maurizio TUCCI (ed).  
*Emerging Methods, Technologies, and Process Manafement in Software Engineering.*  
John Wiley and sons, 2008. ISBN 978-0-470-08571-4.
- [Elmasri2007]  
Ramez ELMASRI, Shamkant B. NAVATHE.  
*Fundamentals of database systems.*  
5<sup>th</sup> edition, Pearson Addison Wesley, 2007, ISBN 0-321-36957-2.
- [Frappier2001]  
Marc Frappier, Henri Habrias.  
*Software specification methods: an overview using a case study.*  
Springer, 2001. ISBN 978-1-8523-3353-9.
- [GDT]  
*Grand dictionnaire terminologique.*  
Office québécois de la langue française.  
<http://www.grandictionnaire.com>  
(consulté le 2011-12-15).
- [GLOGUS]  
GROUPE Μήτις  
*GLOGUS – Recueil de modèles de documents pour le développement logiciel.*  
<http://info.usherbrooke.ca/lavoie/glogus.php>  
Département d'informatique, Faculté des sciences, Université de Sherbrooke,  
Sherbrooke, Canada, janvier 2012.
- [Gottesdiener2002]  
Stephen GOTTESDIENER.  
*Requirements by Collaboration.*  
Eighth Printing (2011), Addison Wesley, Pearson Education ;2002; ISBN 0-201-78606-0.
- [Hull2011]  
Elizabeth HULL, Ken JACKSON, Jeremy DICK.  
*Requirements Engineering.*  
3<sup>rd</sup> edition, Springer, 2011; ISBN 978-1-84996-405-0.
- [IEEE12207]  
*Industry Implementation of International Standard ISO/IEC 12207-1995.*  
IEEE Std 12207.0-1996, IEEE, New York, 1998.
- [IEEE1233]  
*IEEE Guide for Developing System Requirements Specifications.*  
IEEE Std 1233-1998, IEEE, New York, 1998.
- [IEEE830]  
*IEEE Recommended Practice for Software Requirements Specifications;*  
IEEE Std 830-1998, IEEE, New York, 1998.
- [INF 755]  
Luc LAVOIE.  
*INF 755 – Méthodes d'analyse et de conception (notes complémentaires et synthétiques).*  
<http://info.usherbrooke.ca/lavoie/enseignement/INF755>  
Département d'informatique, Faculté des sciences, Université de Sherbrooke,  
Sherbrooke, Canada, janvier 2012.

- [Jackson1995]  
Michael JACKSON.  
*Software Requirements & Specifications*.  
Addison Wesley, 1995; ISBN 0-201-87712-0; [UdeS QA 76.76 D47J33, 1995].
- [Jackson2001]  
Michael JACKSON.  
*Problem frames*.  
ACM Press Book, Addison Wesley, 2001; ISBN 0-20159 627— X.; [UdeS QA 76.76 D47J32, 2001].
- [Jacobson1999] (traduit en français, voir [Jacobson2000])  
Ivar JACOBSON, Grady BOOCH, James RUMBAUGH.  
*The unified software development process*.  
Addison-Wesley, 1999; ISBN 0-201-57169-2.
- [Jacobson2000] (traduction de [Jacobson1999])  
Ivar JACOBSON, Grady BOOCH, James RUMBAUGH.  
*Le processus unifié de développement logiciel*.  
Eyrolles, 2000; ISBN 2-212-09142-7; [UdeS 76.76 D47J3514 2000].
- [Knight2012]  
John KNIGHT.  
*Fundamentals of Dependable Computing for Software Engineers*.  
CRC Press, Taylor and Francis Group, 2012. ISBN 978-1-4398-6255-1.
- [Larman2005]  
Craig LARMAN.  
*Applying UML and patterns – an introduction to object-oriented analysis and design and iterative development*.  
3<sup>rd</sup> edition, Prentice-Hall, Upper Sadel River (NJ), 2005;  
ISBN 0-13-148906-2 [UdeS QA 76.9 O35L3714, 2005].
- [Leffingwell2003]  
D. LEFFINGWELL, D. WIDRIG.  
*Managing software requirements – A use case approach*.  
2<sup>nd</sup> edition, Addison-Wesley, 2003, ISBN 0-321-12247-X; [UdeS QA 76.76 D47L44, 2003].
- [Pfleeger2010]  
Shari Lawrence PFLEEGER, Joanne M. ATLEE.  
*Software Engineering – Theory and Practice*.  
4<sup>th</sup> edition, Prentice Hall, 2005; ISBN 978-0-13-606169-4.
- [Pressman2005]  
Roger S. PRESSMAN.  
*Software Engineering — A practioner’s Approach*.  
6<sup>th</sup> edition, McGraw-Hill, 2005; ISBN 0-07-301933-X; [UdeS QA 76.758 P73, 2005].
- [Printz2012]  
Jacques PRINTZ.  
*Architecture logicielle — Concevoir des applications simples, sûres et adaptables*.  
Dunod, 3<sup>e</sup> édition, 2012; ISBN 978-2-10-057865-8.
- [Sommerville2011]  
Ian SOMMERVILLE.  
*Software Engineering*.  
9<sup>th</sup> edition, Addison-Wesley, 2011; ISBN 978-0-13-703515-1.
- [VanVliet2008]  
Hans VAN VLIET.  
*Software Engineering — Principles and Praticce*.  
3<sup>rd</sup> edition, Wiley, 2008; ISBN 978-0-470-03146-9; [UdeS QA 76.758 V55, 2008].
- [Wiegers2013]  
Stephen WITHALL.  
*Software Requirements*.  
Third edition, Microsoft Press, O’Reilly Media, inc. ; 2013; ISBN 978-0-735-62398-9.

[Withall2007]

Stephen WITHALL.

*Software Requirement Patterns.*

Microsoft Press, O'Reilly Media, inc. ; 2008; ISBN 978-0-7356-7966-5.

[WOL2004]

Wall-On-Line : l'e-gouvernement wallon.

*La boîte à outils : 15 méthodes d'implication des utilisateurs.*

[http://egov.wallonie.be/boite\\_outils\\_methodes/index.htm](http://egov.wallonie.be/boite_outils_methodes/index.htm)

(version en date du 17 décembre 2004 consultée le 11 mai 2007, disponible maintenant sous

<http://info.usherbrooke.ca/lavoie/projets/GLOGUS/wall-on-line.pdf>).

[Wood1995]

Jane WOOD, Denise SILVER.

Joint Application Development.

2<sup>nd</sup> edition, John Wiley & sons, 1995. ISBN 0-471-04299-4.

## Annexe

S	Mercredi	1	2	3	4	A	B	C	D	E	F	G	Thème	Éléments de contenu	Activités	Outils et exemples	Lectures, sources et modules
1	2015-01-07	✓				✓			✓	✓			Problématique de l'IE au sein - du génie logiciel - de l'organisation	Historique, vocabulaire Processus simple Liens avec autres cours Conservation de l'expertise Vision et traçabilité	Cours	Questionnaire, <b>Pointage_DDV</b>	<b>Hull : 1-2; Printz : 1-2.</b> Boehm, Bray, Jackson, Wiegers. <b>INF755_PRE.</b>
2	2015-01-14	✓	✓			✓	✓	✓	✓				Artéfacts de l'IE	DDV, SEL, SES; Rappel des notations de base	Cours, Étude de cas	DC, DFD, EA, EAE, DD. <b>Pointage_SEL.</b>	Lectures essentielles : <b>IE000</b> Approfondissement libre : <b>IE010</b>
3	2015-01-21	✓	✓			✓	✓	✓	✓						Atelier, <b>TP1</b>		
4	2015-01-28			✓			✓	✓	✓				Exploration	Collecte d'information Couverture Pertinence des techniques Choix des techniques Efficience	Cours Étude de cas	(a) profils d'acteurs, entrevue, sondage; (b) remue-méninges, cartographie, Delphi, JAD; (c) recherche doc., glossaire, wiki, CMS; (d) observation de tâches; (e) maquettage, prototypage.	<b>Hull : 5.</b> <b>WOL, IAAT.</b> Gottesdiener, Wood. <b>IE020, IE021, IE022, IE051.</b>
5	2015-02-04			✓			✓	✓	✓						Atelier, <b>TP2</b>		
6	2015-02-11			✓				✓	✓				Analyse	Analyse structurée (pilotée par les processus)	Cours, Étude de cas	AS : notations traditionnelles et unifiées.	<b>Hull : 3; Sommerville.</b> <b>IE030, IE031, TM040, TM060,</b> <b>TM070, TM080</b>
7	2015-02-18			✓				✓	✓						Atelier, <b>TP3</b>		
8	2015-02-25				✓					✓	✓		Spécification	Règles de spécification. Modèles de document.	Cours, Étude de cas	Approches textuelles et structurées. Automates et machines à états	<b>Hull : 4; Bray, GLOGUS.</b> IEEE 830 et 1233. <b>IE040, TM021, TM022.</b>
9	2015-03-04				✓					✓	✓				Atelier, <b>TP4</b>		
10	2015-03-11		✓	✓	✓					✓	✓		Vérification Validation Sélection Traçabilité	Techniques de vérification Principes de validation Règles de sélection Règles de traçabilité	Cours, Étude de cas	Revue internes, revues externes, histoire des cas, scénarios, story-board, grammaires formelles, analyse critériée, gestion des configurations	<b>Leffingwell, Hull : 7.</b> <b>VV010, TM023, TD010.</b>
11	2015-03-18		✓	✓	✓					✓	✓				Atelier, <b>TP5</b>		
12	2015-03-25			✓	✓			✓	✓				Analyse	Analyse pilotée par le problème	Cours, Étude de cas	PP : modèles statiques, modèles dynamiques et méta-modèles ( <i>structures, frames, and patterns</i> ) <b>SuperBroue_DDV, SuperBroue_SEL.</b>	Jackson, Fowler, Bray, Withall. <b>IE033, TM010.</b>
13	2015-04-01			✓	✓			✓	✓						Atelier, <b>TP6</b>		
14	2015-04-08			✓	✓			✓	✓				Analyse	Analyse orientée utilisateur (pilotée par les interactions) Analyse orientée objet (pilotée par le contexte)	Cours, Étude de cas	OOI : histoires de cas et cas d'utilisation. OOC : structure et composants. UML-C, UML-S, UML-I, UML-U. <b>Ascenseur_DDV, Ascenseur_SEL.</b>	<b>Hull : 3; Wiegers, Larman.</b> <b>IE032, IE034.</b>
15	2015-04-15	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Synthèse	Tous	Examen, <b>TP7</b>	Tous	Tous

**Notes.** L'échéance d'une livraison (**TP1 à TP7**) est le dimanche suivant le cours où il est mentionné au calendrier (heure limite : 23 h 59, l'heure des serveurs universitaires en faisant foi).  
La spécification de la livraison est transmise au moins une semaine avant l'échéance.