

## Plan de cours

IGL 301 – Spécification et vérification des exigences (hiver 2011)

### Enseignant

Luc LAVOIE

Courriel : Luc.Lavoie@USherbrooke.ca

Bureau : D4-2006

Téléphone : (819) 821-8000 poste 62015

Site : <http://pages.usherbrooke.ca/llavoie/>

Disponibilité : sur rendez-vous.

### Horaire

Mercredi 08:30 à 10:20 D4-2020

Jeudi 08:30 à 10:20 D4-2020

### Version et statut

1.0.0 - en vigueur (2011-01-05)

---

<b>1</b>	<b>Introduction .....</b>	<b>2</b>
1.1	Objet et portée du document .....	2
1.2	Définitions.....	2
1.3	Références .....	2
<b>2</b>	<b>Présentation .....</b>	<b>5</b>
2.1	Mise en contexte.....	5
2.2	Fiche signalétique .....	5
2.3	Objectifs spécifiques.....	6
<b>3</b>	<b>Contenu.....</b>	<b>7</b>
<b>4</b>	<b>Organisation.....</b>	<b>9</b>
4.1	Modalités d'enseignement .....	9
4.2	Modalités d'évaluation .....	9
4.3	Dispositions relatives au plagiat .....	10
4.4	Calendrier .....	11

# 1 Introduction

## 1.1 Objet et portée du document

Le document décrit l'activité IGL 301 « **Spécification et vérification des exigences** » offerte au trimestre d'hiver 2011 par le Département d'informatique de la Faculté des sciences. On y présente les objectifs, le contenu, l'organisation et les modalités d'évaluation du cours.

## 1.2 Définitions

IEEE	<i>The Institute of Electrical and Electronics Engineers, inc.</i>
SAS	spécification d'architecture du système.
SCL	spécification de conception du logiciel (IEEE SDD <i>software design document</i> ).
SES	spécification des exigences du système (IEEE SRS <i>software requirement specification</i> ).
UML	<i>Unified Modeling Language</i> .

## 1.3 Références

### 1.3.1 Références essentielles

[IGL 301]

COLLECTIF GL.

*IGL 301 – Spécification et validation des exigences (notes complémentaires et synthétiques).*

<http://pages.usherbrooke.ca/llavoie/enseignement/IGL301>

Département d'informatique, Faculté des sciences, Université de Sherbrooke,  
Sherbrooke, Canada, janvier 2011.

[GLOGUS]

LAVOIE, Luc.

*GLOGUS – recueil de modèles de documents pour le développement logiciel.*

<http://pages.usherbrooke.ca/llavoie/glogus.php>

Département d'informatique, Faculté des sciences, Université de Sherbrooke,  
Sherbrooke, Canada, janvier 2011.

### 1.3.2 Références importantes

[Bray2002]

K. BRAY.

*An Introduction to requirements engineering.*

Addison-Wesley, 2002.

ISBN 0-201-76792-9 ; UdeS QA 76.758 B744 2002.

[Elmasri2007]

ELMASRI, Ramez ; NAVATHE, Shamkant B.  
*Fundamentals of database systems.*  
Fifth Edition, Pearson Addison Wesley, 2007.  
ISBN 0-321-36957-2.

[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.

[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://pages.usherbrooke.ca/llavoie/projets/GLOGUS/wall-on-line.pdf>)

### 1.3.3 Références utiles

[Braude2001]

Eric J. BRAUDE.  
*Software engineering: an object-oriented perspective.*  
John Wiley & sons, 2001.  
ISBN 0-471-32208-3 [QA 76.758 B74 2000]

[Braude2011]

Eric J. BRAUDE, Michael E. BERNSTEIN.  
*Software engineering: modern approaches.*  
John Wiley & sons, 2011.  
ISBN 979-0-471-69208-9.

[Davis2007]

M. Davis ;  
*Requirements Bibliography,*  
<http://web.uccs.edu/adavis/UCCS/reqbib.htm>  
(consulté le 2007-03-15)

[GDT]

*Grand dictionnaire terminologique.*  
Office québécois de la langue française.  
<http://www.granddictionnaire.com>  
(consulté le 2011-12-15).

[Hull2004]

E. HULL, K. JACKSON, J. DICK;  
*Requirements engineering*;  
2/E, Springer, 2004;  
[TA 168 H85 2005]

[IEEE1233]

*IEEE Guide for Developing System Requirements Specifications*;  
IEEE Std 1233-1998, IEEE, New York, 1998;  
[QA 76.76 S73I438 1998 – disponible au comptoir de la bibliothèque de Sciences et Génie]

[IEEE830]

*IEEE Recommended Practice for Software Requirements Specifications*;  
IEEE Std 830-1998, IEEE, New York, 1998;  
[QA 76.76 S73I44 1998 – disponible au comptoir de la bibliothèque de Sciences et Génie]

[IEEE12207]

*Industry Implementation of International Standard ISO/IEC 12207-1995*;  
IEEE Std 12207.0-1996, IEEE, New York, 1998.  
[QA 76.76 S73I44 1998 – disponible au comptoir de la bibliothèque de Sciences et Génie]

[Jackson1995]

Michael JACKSON;  
*Software Requirements & Specifications*;  
Addison Wesley, 1995; ISBN 0-201-87712-0.

[Jackson2001]

Michael JACKSON;  
*Problem frames*;  
ACM Press Book, Addison Wesley, 2001; ISBN 0-201-59627-X.

[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].

[Larman2005]

Craig LARMAN;  
*Applying UML and patterns - an introduction to object-oriented analysis and design and iterative development*;  
3/E, Prentice-Hall, Upper Sadel River (NJ), 2005; ISBN 0-13-148906-2.

[Pfleeger2010]

PFLEEGER, Shari Lawrence, Joanne M. ATLEE.  
*Software Engineering – Theory and Practice*.  
Fourth Edition, Prentice Hall, 2005; ISBN 978-0-13-606169-4.

[Pressman2005]

Roger S. PRESSMAN

*Software Engineering - A practioner's Approach.*

Sixth Edition, McGraw-Hill, 2005; ISBN 0-07-301933-X.

[Sommerville2007]

SOMMERVILLE, Ian

*Software Engineering.*

Height Edition, Addison-Wesley, 2007; ISBN 978-0-321-31379-9.

## 2 Présentation

### 2.1 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 aussi être vu comme une machine, mais il diffère des autres machines en ce sens qu'il est intangible. Le génie logiciel 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 Autriche 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 but donné, il est préférable de déterminer et de détailler d'abord quel est ce but. Ce qui nous amène à l'ingénierie des exigences, la partie du génie logiciel qui permet de déterminer quel système sera développé.

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 issues des activités préalables d'exploration et d'analyse.

### 2.2 Fiche signalétique

#### *Objectif*

Spécifier, valider et vérifier les exigences des clients; en déduire une architecture technologique.

#### *Contenu*

Spécifications fonctionnelles et non fonctionnelles. Diagramme de flux de données et modèles de données. Spécification textuelle des exigences. Cas d'utilisation et scénario. Validation des exigences. Génération de scénarios de tests d'acceptation. Élaboration de l'architecture. Présentation des normes de spécification IEEE.

**Concomitante**

IFT 232 – Méthodes de conception orientées objet

**Crédits**

3

**Organisation**

Cours : 3 heures par semaine

Travaux dirigés : 1 heure par semaine

Travail personnel : 5 heures par semaine

**Référence**

<http://www.usherbrooke.ca/fiches-cours/igl301.htm>

**2.3 Objectifs spécifiques**

Au terme de cette activité pédagogique, la personne l'ayant réussie sera capable de :

- ◇ appliquer le processus d'ingénierie des exigences ;
- ◇ établir les relations entre l'ingénierie des exigences et les autres processus du développement logiciel ;
- ◇ 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 ;
- ◇ générer des scénarios de test fonctionnel ;
- ◇ déduire une architecture technologique.

### 3 Contenu

#### 0. Introduction

- 0.1. Mise en contexte historique
- 0.2. Terminologie

Bray 1  
Bray 1, 2 ; GDT

#### 1. Procédés et processus de développement logiciel

- 1.1. Prédicatifs (cascades, V, etc.)
- 1.2. Itératifs (itératif simple, spirale, unifié, etc.)
- 1.3. Agiles (XP, Scrum, etc.)
- 1.4. Spécification : le matériel versus le logiciel
- 1.5. Normes IEEE, ISO, militaires et aérospatiales

Pressman 2  
Pressman 3, Leffingwell 3  
Pressman 4  
Leffingwell 3  
IEEE 830, 1233, 12207

#### 2. Procédés et processus d'ingénierie des exigences

- 2.1. Exploration
- 2.2. Analyse
- 2.3. Spécification
- 2.4. Design (IPM)
- 2.5. Conception externe (IMM)
- 2.6. Liens avec la conception interne
- 2.7. Validation

Bray 2, Jacobson2000 2.3

#### 3. Documentation du processus d'IE

- 3.1. Document de vision, énoncé de portée et mandat
- 3.2. Document de spécification des exigences
- 3.3. Glossaire
- 3.4. Références
- 3.5. Spécification
- 3.6. Traçabilité

Leffingwell 16, GLOGUS  
GLOGUS  
GLOGUS  
GLOGUS  
Notes de cours  
Notes de cours

#### 4. Techniques d'interaction et d'exploration

- 4.1. Cartes d'acteurs
- 4.2. Entrevues et questionnaires
- 4.3. Ateliers
  - 4.3.1. *Brain storming*
  - 4.3.2. *Mind mapping* et tri par cartes
  - 4.3.3. Analyse experte (Delphi)
  - 4.3.4. *Focus Group*
  - 4.3.5. *Storyboarding*
  - 4.3.6. Survol de quelques autres méthodes en atelier
- 4.4. Analyse de documents
- 4.5. Observation et analyse des tâches
- 4.6. Méthodes participatives

WOL  
Bray 3, 9 ; WOL  
  
Bray 9, Leffingwell 12, WOL  
Notes de cours, WOL  
WOL  
WOL  
Leffingwell 13  
WOL  
Bray 3, 9 ; WOL  
Bray 3, 9 ; WOL  
WOL, notes de cours

#### 5. Méthodes d'analyse

- 5.1. Analyse pilotée par le problème

Bray, Jackson, notes de cours

5.1.1.	Présentation	
5.1.2.	Quelques exemples avec les Diagrammes de Jackson	
5.1.3.	Validation	
5.2.	Analyse structurée	Bray 4.3, 13.1 ; Pressman
5.2.1.	Présentation	
5.2.2.	Quelques exemples avec SA, SADT et SSADM	
5.2.3.	Validation	
5.3.	Analyse orientée objet	Leffingwell
5.3.1.	Présentation	
5.3.2.	Quelques exemples avec UML	
5.3.3.	Validation	
5.4.	Synthèse	Pfleeger 4
5.4.1.	Comparaison des méthodes	
5.4.2.	Du choix de la méthode	
<b>6.</b>	<b>Techniques de modélisation</b>	
6.1.	Présentation	Bray 4, 5
6.2.	Langue naturelle	Bray 14
6.3.	Diagramme de contexte	Bray 13.1, Sommerville 8.1
6.4.	Modèles conceptuels de données	Bray 13.3, Elmasri 3, 4
6.4.1.	Diagramme entité-relation	
6.4.2.	Dictionnaire de données	
6.4.3.	Diagramme entité-relation étendu	
6.5.	Diagrammes de Jackson ( <i>Jackson's Frames</i> )	Jackson2001
6.6.	<del>Diagramme de structure JSD</del>	<del>Van Vliet 12.2.3</del>
6.7.	Diagramme de flux de données	Bray 13.1, Pressman 8.6
6.8.	Tables de décision	Bray 14
6.9.	Diagrammes état-transition	Bray 12.6
6.9.1.	Automate	
6.9.2.	Machine à états	
6.9.3.	Diagramme d'états ( <i>State Chart</i> )	
6.9.4.	SDL	
6.10.	<del>Réseau de Petri</del>	<del>Bray 12.7</del>
6.11.	Pseudo-code	Bray 14
6.12.	Cas d'utilisation	Leffingwell 14, 21
6.13.	Maquettage	Bray 11.2, Sommerville 16.4
6.14.	Prototypage	Bray 11.3, Sommerville 17.4
6.15.	Notation UML	Larman 15, 16, 28, 29
6.15.1.	Diagrammes statiques	
6.15.2.	Diagrammes dynamiques	
<b>7.</b>	<b>De la spécification à l'architecture</b>	Leffingwell 25, notes de cours
7.1.	Modélisation de la structure et du comportement avec UML	
7.2.	Dérivation des diagrammes de classes	
7.3.	Dérivation des diagrammes de séquence	



## 8. De la spécification aux essais

Leffingwell 26, notes de cours

# 4 Organisation

## 4.1 Modalités d'enseignement

Les périodes de cours visent à expliquer la matière contenue dans les manuels de référence. L'étudiante, l'étudiant, est responsable d'effectuer préalablement les lectures correspondant au sujet de la semaine.

Les travaux dirigés présentent des exercices individuels ou en groupe selon les exigences du programme et les besoins des étudiantes et des étudiants.

Les travaux pratiques consistent en des prestations nécessitant l'utilisation de concepts, de méthodes et de techniques présentées en cours. Ces travaux ne comprennent pas de programmation.

## 4.2 Modalités d'évaluation

En plus de l'examen de mi-parcours et de l'examen final, l'évaluation comprend des travaux pratiques accomplis dans le cadre du projet de session. Ces travaux sont réalisés en équipe de deux à trois personnes. Un bilan de fin de projet, individuel, complète l'évaluation.

Tableau 1 – Sommaire des évaluations

Évaluation	Valeur	Commentaire
Examen de mi-parcours	30 %	Individuel
Examen récapitulatif	40 %	Individuel
TP1 – Démarrage du projet	5 %	En équipe de deux ou trois
TP2 – Exploration	5 %	En équipe de deux ou trois
TP3 – Analyse	5 %	En équipe de deux ou trois
TP4 – Spécification	5 %	En équipe de deux ou trois
TP5 – Bilan de fin de projet	10 %	Individuel
<b>Total</b>	<b>100 %</b>	

Tout étudiant, toute étudiante, qui omet de remettre un travail au moment prescrit par l'échéancier doit rencontrer l'enseignant afin de déterminer une nouvelle date de remise. Dans tous les cas, une pénalité de 10 % par jour de retard est imposée.

L'évaluation est faite en tenant compte de la clarté des documents. 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.

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.

<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)

#### 4.2.1 Projet de session

Le projet de session consiste en :

1. le démarrage du projet,
2. l'exploration du domaine,
3. l'analyse du produit,
4. la spécification du produit,
5. le bilan du projet.

Chacun des extrants du projet choisi doit être remis à la date prescrite selon la planification hebdomadaire. De plus, à la fin de la session, le projet complet **présenté de manière professionnelle** doit être préparé sous la forme d'une archive au format zip et transmis par courriel à l'enseignant. Les logiciels et les formats de fichiers seront fixés par les énoncés de travaux. Les travaux ne sont pas retournés aux auteurs à la fin du cours.

#### 4.2.2 Examens et contrôles

La durée de l'examen de mi-parcours est de 110 minutes et celle de l'examen final est de trois heures – aucune documentation n'est permise et l'usage d'appareils informatiques, électroniques ou de communication (ordinateur, calculatrice, téléphone, etc.) est interdit.

### 4.3 Dispositions relatives au 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 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 référé au Secrétaire de la Faculté des sciences.

---

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

## 4.4 Calendrier

**Tableau 2 – Calendrier des activités**

N <sup>o</sup>	Semaine	Activité	Contenu	Évaluation (remises)
1	2011-01-03	cours + TD	0, 1	
2	2011-01-10	cours + TD	1	
3	2011-01-17	cours + TD	2, 3	
4	2011-01-24	cours + TD	4	TP1
5	2011-01-31	cours + TD	5.1, 6.1 – 6.5	
6	2011-02-07	cours + TD	5.1, 6.1 – 6.5	
7	2011-02-14	cours + TD	5.2, 6.7 – 6.11	TP2
8	2011-02-21	examen	(1 – 5.1) + (6.1 – 6.5)	examen de mi-parcours
9	2011-02-28	relâche		
10	2011-03-07	cours + TD	5.2, 6.7 – 6.11	
11	2011-03-14	cours + TD	5.3, 6.12 – 6.16	TP3
12	2011-03-21	cours + TD	5.3, 6.12 – 6.16	
13	2011-03-28	cours + TD	5.4	
14	2011-04-04	cours + TD	7	TP4
15	2011-04-11	cours + TD	8	
16	2011-04-18	remise	--	TP5
17	à déterminer	examen	(1 – 8)	examen récapitulatif

Notes :

- Les jours de remise sont les mercredis.
- Il y a relâche le mercredi 26 janvier (Carnaval).
- La date de l'examen de mi-parcours est fixée par la Faculté (entre le 15 et le 26 février).
- La date de l'examen final est fixée par la Faculté (entre le 15 et le 29 avril).