

## Plan de cours IGL301 – hiver 2008

### Spécification et vérification des exigences

#### Enseignant

Luc LAVOIE

Courriel : <mailto:Luc.Lavoie@USherbrooke.ca>

Bureau : D4-1010-12

Téléphone : (819) 821-8000 poste 62015

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

Disponibilité : sur rendez-vous.

#### Horaire

Lundi 16 h 30 à 17 h 20 D7-2009

Mardi 10 h 30 à 12 h 20 D7-2009

#### Version et statut

1.0.0 - En vigueur en date du 2008-01-07

---

<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>6</b>
<b>4</b>	<b>Organisation.....</b>	<b>8</b>
4.1	Modalités d'enseignement.....	8
4.2	Modalités d'évaluation.....	8
4.3	Calendrier .....	8

# 1 Introduction

## 1.1 Objet et portée du document

Le document s'adresse aux personnes inscrites au programme d'informatique de gestion délocalisé au Maroc. Il décrit l'activité IGL301 « **Spécification et vérification des exigences** » offerte au trimestre d'hiver 2008. 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>
PGC	plan de gestion de configuration (IEEE SCMP <i>software configuration management plan</i> ).
PGP	plan de gestion de projet (IEEE SPMP <i>software project management plan</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

[IGL301]

COLLECTIF GL ;

*IGL301 – Spécification et validation des exigences*

Notes complémentaires et synthétiques, Département d'informatique, Faculté des sciences, Université de Sherbrooke, Sherbrooke, Canada, avril 2007.

[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, avril 2007.

### 1.3.2 Références importantes

[Bray]

K. BRAY;

*An Introduction to requirements engineering;*

Addison-Wesley, 2003.

[Leffingwell]

D. LEFFINGWELL, D. WIDRIG;  
*Managing software requirements – A use case approach ;*  
2<sup>nd</sup> edition, Addison-Wesley, 2003.

[WOL]

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)  
(consulté le 11 mai 2007, version en date du 17 décembre 2004)

### 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]

[Davis2007]

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

[Hull2004]

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

[IEEE 1233]

*IEEE Guide for Developing System Requirements Specifications;*  
IEEE Std 1233-1998, IEEE, New York, 1998;  
[QA 76.76 S73I438 1998]

[IEEE 830]

*IEEE Recommended Practice for Software Requirements Specifications;*  
IEEE Std 830-1998, IEEE, New York, 1998;  
[QA 76.76 S73I44 1998]

[IEEE 12207]

*Industry Implementation of International Standard ISO/IEC 12207-1995;*  
IEEE 12207, IEEE, New York, 1995.

[ISO 12207]

*ISO/IEC 12207 - Information Technology—Software Life-Cycle Processes;*  
1995.

[Jackson2001]

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

[Jackson1995]

Michael JACKSON;  
*Software Requirements & Specifications*;  
Addison Wesley, 1995.

[Kotonya1998]

G. KOTONYA and I. SOMMERVILLE;  
*Requirements engineering: processes and techniques*;  
John Wiley, 1998;  
[QA 76.758 K67 1998]

[Kovitz1998]

B. L. KOVITZ;  
*Practical Software Requirements: A Manual of Content and Style*;  
Manning Publications Company, 1998;  
[QA 76.76 D47K68 1999]

[Kulak02]

D. Kulak, E. Guiney;  
*Use Cases: Requirements in Context*;  
2/E, Addison Wesley Professional, 2004.

[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.

[Lauesen2002]

S. Lauesen;  
*Software Requirements: Styles and Techniques*;  
Addison Wesley Professional, 2002  
[QA 76.754 L38 2002].

[Pressman2005]

PRESSMAN, Roger S.  
*Software Engineering - A practioner's Approach*.  
Sixth Edition, McGraw-Hill, 2005.  
ISBN 0-07-301933-X.

## 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. Son nom et sa constitution comme un domaine de connaissance propre coïncident avec un séminaire organisé par l'OTAN à Garmisch-Partenkirchen en Autriche en 1968. Böhm et Bauer en sont probablement les parrains.

Avant de construire une machine ou un artefact qui doit atteindre un but donné, il est préférable de déterminer et de détailler 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 : nous préférons utiliser le terme « spécification des exigences » pour désigner une des activités de l'ingénierie des exigences. Succinctement, la spécification des exigences est 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. Dans de nombreux contextes, les deux expressions peuvent cependant être utilisées l'une pour l'autre.

### 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*

IFT232 – 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/programmes/cours/IGL/igl301.htm>

## 2.3 Objectifs spécifiques

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

- ◇ connaître le processus de spécification des exigences ;
- ◇ connaître les relations entre le processus logiciel et la spécification des exigences ;
- ◇ connaître la structure d'un document de spécification des exigences ;
- ◇ maîtriser les techniques d'élicitation des exigences ;
- ◇ maîtriser les techniques de spécification des exigences ;
- ◇ vérifier les exigences ;
- ◇ générer des scénarios de test fonctionnel ;
- ◇ déduire une architecture technologique.

## 3 Contenu

### 1. Introduction

- |  |                             |
|--|-----------------------------|
| 1.1. Mise en contexte historique                     | Bray 1                      |
| 1.2. Terminologie                                    | Bray 1, 2 ; GDT             |
| 1.3. Procédés et processus de développement logiciel | Leffingwell 3, Pressman 2-4 |
| 1.4. Spécification : le matériel versus le logiciel  | Leffingwell 3               |
| 1.5. Normes IEEE, ISO, militaires et aérospatiales   | IEEE 830, 1233              |

### 2. Processus de spécification des exigences

Bray 2

### 3. Techniques d'exploration

- |   |                            |
|---|----------------------------|
| 3.1. Cartes d'acteurs                                       | WOL                        |
| 3.2. Entrevues et questionnaires                            | Bray 3, 9 ; WOL            |
| 3.3. Ateliers   |                            |
| 3.3.1. <i>Brain storming</i>                                | Bray 9, Leffingell 12, WOL |
| 3.3.2. <i>Mind mapping</i>                                  | Notes de cours             |
| 3.3.3. Analyse experte (Delphi)                             | WOL                        |
| 3.3.4. <i>Focus Group</i>                                   | WOL                        |
| 3.3.5. Storyboarding  | Leffingwell 13             |
| 3.3.6. <i>Survot de quelques autres méthodes en atelier</i> | WOL                        |

3.4. Analyse de documents	Bray 3, 9 ; WOL
3.5. Observation et analyse des tâches	Bray 3, 9 ; WOL
3.6. Méthode participative	WOL, notes de cours
<b>4. Documentation du processus d'IE</b>	
4.1. Document de vision	Leffingwell 16
4.2. Document de spécification des exigences	GLOGUS
4.3. Le glossaire	GLOGUS
4.4. La liste des références	GLOGUS
<b>5. Techniques de spécification</b>	
5.1. Présentation	Bray 4, 5
5.2. Analyse structurée	Bray 4.3, 13.1
5.2.1. Diagramme de contexte	
5.2.2. Diagramme de flux de données	
5.2.3. Modèle conceptuel de données	
5.2.4. Dictionnaire de données	
5.3. Langue naturelle	Bray 14
5.4. Tables de décision	Bray 14
5.5. Pseudo-code	Bray 14
5.6. Diagrammes état-transition	Bray 12.6
5.6.1. Automate	
5.6.2. Machine à états	
5.6.3. <i>State Chart</i>	
5.6.4. SDL	
5.7. Diagramme de structure JSD et méthode EB3	Bray 13.4
5.8. Réseau de Petri	Bray 12.7
5.9. Cas d'utilisation	Leffingwell, 14, 21
5.10. Diagramme d'activités	Leffingwell 24
5.11. Maquettage	Notes de cours
5.12. Prototypage	Notes de cours
<b>6. Des spécifications à l'architecture</b>	Leffingwell 25
6.1. Modélisation de la structure et du comportement avec UML	
6.2. Diagramme de classes	
6.3. Diagramme de séquence	
<b>7. Des spécifications aux essais</b>	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 des deux examens individuels, l'évaluation porte sur :

- ◇ deux travaux pratiques individuels ;
- ◇ un projet à être réalisés en équipe de deux personnes ou trois personnes.

Le correcteur ou la correctrice peut soustraire jusqu'à 5% de chaque évaluation pour la qualité du français. Des consignes supplémentaires ou des modifications pourront être communiquées au cours du trimestre.

La durée de l'examen période est de 110 minutes et celle de l'examen final est de trois heures – aucune documentation n'est permise et l'usage de la calculatrice est interdit.

**Tableau 1 – Sommaire des évaluations**

Évaluation	Valeur	Commentaire
examen de mi-session	30 %	Individuel
examen final	40 %	Individuel et récapitulatif
TP 1	5 %	Individuel
TP 2	5 %	Individuel
Projet	20 %	en équipe de deux ou trois
<b>Total</b>	<b>100 %</b>	

Toute situation éventuelle de plagiat se verrait traitée selon les normes du « Règlement des études » de l'Université de Sherbrooke, disponible à cette adresse :

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



### 4.3 Calendrier

**Tableau 2 – Calendrier des activités**

N°	date	activité	contenu	évaluation
1	2008-01-07	cours	1	
2	2008-01-14	cours + TD	2	énoncé TP1
3	2008-01-21	cours + TD	3.1, 3.2, 3.3,	
4	2008-01-28	cours + TD	3.3, 3.5, 3.6, 4	remise TP1 / énoncé TP2
5	2008-02-04	cours + TD	5.1, 5.2	
6	2008-02-11	cours + TD	5.3, 5.4	remise TP2
7	2008-02-18	cours	5.5	énoncé du projet
8	2008-02-25	examen	(1 – 5.5)	examen de mi-session
9	2008-03-03	relâche		
10	2008-03-10	cours + TD	5.6	
11	2008-03-17	cours	5.7	
12	2008-03-24	cours + TD	5.8	
13	2008-03-31	cours + TD	5.9, 5.10	
14	2008-04-07	cours + TD	5.11, 5.12	
15	2008-04-14	cours	6, 7	remise projet
16	à déterminer	examen	(1 – 7)	examen final

Notes :

- Il n'y a pas cours le lundi 24 mars ni le mercredi 16 avril.
- Les jours de remise sont les lundis.
- L'examen de mi-session a lieu le lundi 25 février.
- La date de l'examen final est fixée par la Faculté.