

---

## Plan de cours IFT324

Génie logiciel – été 2005

### Enseignant

Luc Lavoie

Luc.A.Lavoie@USherbrooke.ca

### Date

2005-05-05

### Statut

Version 1.0.0, en vigueur

---

<b>1</b>	<b>Introduction .....</b>	<b>2</b>
1.1	Objet et portée du document.....	2
1.2	Références .....	2
1.3	Définitions.....	5
1.4	Mise en contexte.....	5
1.5	Vue d'ensemble du document .....	5
<b>2</b>	<b>Objectifs.....</b>	<b>5</b>
2.1	Fiche signalétique .....	5
2.2	Objectifs spécifiques .....	6
<b>3</b>	<b>Organisation .....</b>	<b>6</b>
3.1	Modalités d'enseignement.....	6
3.2	Modalités d'évaluation .....	6
3.3	Calendrier .....	6
3.4	Modalités d'organisation .....	7
<b>4</b>	<b>Contenu.....</b>	<b>7</b>

# 1 Introduction

## 1.1 Objet et portée du document

Le document s'adresse aux étudiants inscrits à l'activité IFT324 « Génie logiciel » du trimestre d'été 2005. On y présente les objectifs, le contenu et l'organisation de l'activité.

## 1.2 Références

[Arnold94]

Robert S. ARNOLD;  
*Software reengineering*;  
IEEE Computer Society Press, 1994;  
ISBN 0-8186-3272-0.

[Braude01]

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

[Bray03]

K. BRAY;  
*An Introduction to requirements engineering*;  
Addison-Wesley, 2003;  
[QA 76.758 B744 2002]

[Gamma95]

Erich GAMMA, Richard HELM, Ralph JOHNSON, John VLISSIDES;  
*Design patterns - elements of reusable object-oriented software*;  
Addison-Wesley, Longman (MA), 1995;  
ISBN 0-201-63361-2.

[Hull04]

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

[IEEE Std 1233-1998]

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

[IEEE Std 830-1998]

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

[IEEE/EIA 12207]

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

[ISO/IEC 12207-1995]

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

[Jackson01]

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

[Jackson95]

Michael JACKSON;  
*Software Requirements & Specifications;*  
Addison Wesley, 1995;  
[QA 76.76 D47]33 1995]

[Jacobson94]

Ivar JACOBSON;  
*Object-Oriented Software Engineering;*  
ACM Press Book, Addison Wesley, 1994;  
ISBN 0-201-54435-0.

[Kotonya98]

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

[Kovitz98]

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.

[Lai97]

Michel LAI;  
*UML - La notation unifiée de modélisation objet;*  
Masson, Paris, 1997;  
ISBN 2-7296-0659-9.

[Larman04]

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), 2004;  
ISBN 0-13-148906-2.

[Lauesen02]

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

[Lavi03]

J. Z. Lavi, J. Kudish;  
*Systems Modeling & Requirements Specification Using ECSAM: Embedded Computer-Based Systems Analysis*;  
Dorset House, 2003;  
[TK7895 .E42 L38 2004]

[Leffingwell03]

D. Leffingwell, D. Widrig;  
*Managing Software Requirements: A Use Case Approach*;  
2/E, Addison Wesley, 2003;  
[QA 76.76 D47L44 2003]

[Maciaszek03]

L. Maciaszek;  
*Requirements Analysis and Systems Design: Developing Information Systems with UML*;  
2/E, Addison-Wesley, 2004.

[Pressman92]

Rober S. Pressman;  
*Software Engineering*;  
Mc Graw Hill, 1992;  
ISBN 0-07-050814-3.

[Robertson99]

S. Robertson, J. Robertson;  
*Mastering the Requirements Process*;  
Addison-Wesley, 1999;  
[QA 76.76 D47R636 1999]

[Rumbaugh91]

James RUMBAUGH, Michael BLAHA, William PREMERLANI, Frederick EDDY, William LORENSEN;  
*Object-oriented Modeling and design*;  
Prentice-Hall, Englewood Cliffs (NJ), 1991;  
ISBN 0-13-629841-9.

[Sawyer97]

P. Sawyer, I. Sommerville;  
Requirements Engineering: A good practice guide;  
John Wiley & Sons, 1997;  
[QA 76.76 D47S658 1997]

### 1.3 Définitions

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 requirements specification</i> ).

### 1.4 Mise en contexte

Non applicable.

### 1.5 Vue d'ensemble du document

Le plan est divisé comme suit :

- ◇ Introduction : la présente section.
- ◇ Objectifs : description des objectifs de l'activité.
- ◇ Organisation : organisation de l'activité, modalités d'évaluation, calendrier.
- ◇ Contenu : inventaire du contenu des cours magistraux.

## 2 Objectifs

### 2.1 Fiche signalétique

#### *Objectif*

Connaître les critères de qualité du logiciel et être en mesure d'utiliser une gamme d'outils pour analyser, concevoir et développer des systèmes satisfaisant ces critères.

#### *Contenu*

Définition et objectifs. Modèles de cycle de vie. Éléments d'un environnement de développement: méthodes, notations et outils logiciels. Méthodes d'analyse et de conception: concepts, cohésion, couplage. Méthodes basées sur les flux ou les structures de données. Méthodes orientées objets. Techniques de validation et vérification. Essais. Implantation et maintenance. Prospective en génie logiciel.

### *Préalable*

Avoir obtenu 24 crédits du programme

## **2.2 Objectifs spécifiques**

- ◇ Connaître les principales étapes de la gestion d'un projet de développement logiciel.
- ◇ Être capable d'identifier les processus, les activités et les phases du cycle de vie d'un produit logiciel.
- ◇ Être capable de constituer un dossier de projet de développement logiciel.
- ◇ Être capable d'évaluer la qualité d'un livrable.
- ◇ Être capable d'évaluer les ressources requises pour la production d'un livrable simple.
- ◇ Être capable de travailler au sein d'une équipe en respectant un plan de projet.
- ◇ Être capable d'insérer la démarche de conception dans le cadre général du génie logiciel.

## **3 Organisation**

### **3.1 Modalités d'enseignement**

Le cours comporte 42 exposés magistraux de 50 minutes, un examen périodique d'une durée de 1h50 et un examen final de 3h selon le calendrier présenté en 3.3.

Le contenu du cours est principalement tirée de [Braude01] complétés par des modules de cours et des gabarits à être distribués par l'enseignant.

### **3.2 Modalités d'évaluation**

L'évaluation comprend 5 travaux pratiques (PGP 5%, PGC 5%, SES 10%, SAS 10%, SCL 5%), un examen périodique (25%) et un examen final (40%).

Les travaux se font en équipe de deux ou trois personnes, les dates de remise sont fixées en fonction de l'évolution des cours.

L'examen périodique aura lieu le vendredi 10 juin de 08h30 à 10h20 et l'examen final à une date à être fixée par la Faculté des sciences.

### **3.3 Calendrier**

L'horaire régulier du cours est établi sur onze semaines (du 5 mai 2005 au 15 juillet 2005) et comporte deux périodes de deux heures les jeudis de 10:30 à 12:20 et les vendredis de 08:30 à 10:20. Comme il y aura suspension des cours trois vendredi (20 mai, 24 juin et 1er juillet), il a fallu prévoir des modalités de reprise pour un total de 6 périodes. Ce seront les périodes du lundi de 10:30 à 11:20 les 30 mai, 6, 13, 20, 27 juin et 4 juillet.

### 3.4 Modalités d'organisation

- ◇ Toutes les remises de travaux doivent être par l'entremise du casier de dépôt identifié au cours qui se trouve au sous-sol du D-4. Aucun retard n'est accepté. Un travail non remis dans les délais se voit attribué la note « 0 ».
- ◇ Tout courriel adressé à l'enseignant ou au à l'auxiliaire doit comprendre un objet débutant par « IFT324 »

## 4 Contenu

Les références au manuel de Braude sont indiquées entre crochets.

### 1. Introduction

- 1.1. Objet et portée du cours
- 1.2. Documents de référence
- 1.3. Définitions
- 1.4. Préalables
- 1.5. Présentation du contenu
- 1.6. Organisation

### 2. Présentation [0, 1]

- 2.1. Historique
- 2.2. Démarche qualité
- 2.3. Processus qualité
- 2.4. Normes courantes
- 2.5. Assurance de la qualité versus contrôle de la qualité
- 2.6. Vérification versus validation
- 2.7. Audit versus inspection
- 2.8. Critères de qualité
- 2.9. Cycle de vie
  - 2.9.1. Exploration
  - 2.9.2. Analyse
  - 2.9.3. Architecture
  - 2.9.4. Conception
  - 2.9.5. Mise en oeuvre
  - 2.9.6. Mise en service
  - 2.9.7. Entretien-évolution
  - 2.9.8. Retrait de service
- 2.10. Processus continus
  - 2.10.1. Gestion de projet
  - 2.10.2. Gestion de configuration
  - 2.10.3. Documentation
  - 2.10.4. Vérification et validation
  - 2.10.5. Mesure et support logistique
  - 2.10.6. Assurance et contrôle de qualité

2.11. Les normes applicables

2.12. Les critères applicables

### **3. Gestion de projet [2]**

3.1. Planification et programme

3.2. Estimation

3.3. Outils et méthodes

3.4. Intégration au cycle de vie du logiciel

3.5. Documentation

3.6. Revues

3.7. Facteurs humains

### **4. Exploration [3]**

Besoins

Opportunité.

Faisabilité

Vérification et validation

Intégration au cycle de vie du logiciel

Documentation

Revue

Facteurs humains

### **5. Gestion de configuration et documentation [2.7.3]**

Outils et méthodes

Intégration au cycle de vie du logiciel

Estimation

Documentation

Revue

Facteurs humains

### **6. Vérification, validation, assurance et contrôle de qualité [2.6.6, 9]**

Outils et méthodes

Intégration au cycle de vie du logiciel

Estimation

Documentation

Revue

Facteurs humains

### **7. Analyse [4]**

Vérification et validation

Intégration au cycle de vie du logiciel

Documentation

Revue

Facteurs humains

### **8. Architecture [5]**

Vérification et validation



Intégration au cycle de vie du logiciel  
Documentation  
Revue  
Facteurs humains

## **9. Critères et techniques de conception**

### 9.1. Présentation des principaux critères

- 9.1.1. Maintenabilité
- 9.1.2. Disponibilité
- 9.1.3. Sécurité
- 9.1.4. Robustesse
- 9.1.5. Flexibilité
- 9.1.6. Coût
- 9.1.7. Performance
- 9.1.8. Interopérabilité
- 9.1.9. Autres critères

### 9.2. Présentation des principales techniques

- 9.2.1. Découplage
- 9.2.2. Cohésion structurelle
- 9.2.3. Cohésion logique
- 9.2.4. Cohésion temporelle
- 9.2.5. Cohésion procédurale
- 9.2.6. Cohésion fonctionnelle
- 9.2.7. Traçabilité
- 9.2.8. Autres techniques

## **10. Conception et mise en oeuvre [6, 7]**

Vérification et validation  
Intégration au cycle de vie du logiciel  
Niveaux (système, classe, méthode)  
Mode (fermé, ouvert)  
Documentation  
Revue  
Facteurs humains

## **11. Essais [8, 9]**

Vérification et validation  
Niveaux (système, classe, méthode)  
Mode (fermé, ouvert)  
Intégration au cycle de vie du logiciel  
Documentation  
Revue  
Facteurs humains

## **12. Entretien et évolution [10]**

Outils et méthodes



Intégration au cycle de vie du logiciel  
Estimation  
Documentation  
Revue  
Facteurs humains

### **13. Mesure et support logistique**

Outils et méthodes  
Intégration au cycle de vie du logiciel  
Estimation  
Documentation  
Revue  
Facteurs humains

---

IFT324 Génie logiciel  
été 2005  
Questionnaire

Nom

\_\_\_\_\_

Courriel

\_\_\_\_\_

Matricule

\_\_\_\_\_

Programme

\_\_\_\_\_

*IFT 287*

Quand avez-vous suivi le cours IFT287 ?

\_\_\_\_\_

Qui était l'enseignant ?

\_\_\_\_\_

Quel résultat avez-vous obtenu ?

\_\_\_\_\_

Quels sont les éléments que vous avez le mieux assimilés ?

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

Quels sont les éléments que vous avez le moins bien assimilés ?

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

Quels sont les éléments qui vous ont servis en stage ?

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

Quels sont les éléments que vous auriez désiré apprendre et qui vous auraient été utiles en stage ?

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

*Vocabulaire*

Définissez brièvement les termes suivants :

Analyse

---

---

---

Conception

---

---

---

Composition

---

---

---

Généralisation (dérivation)

---

---

---

Vérification

---

---

---

Validation

---

---

---

Antécédent (*pre-condition*)

---

---

---

Conséquent (*post-condition*)

---

---

---