

	Fiche module	Mise à jour :
	Cycle de formation d'ingénieurs en Informatique	Page 1 sur 4

Module : Design Patterns			Code
			ING-4-GLSI-S7-P3
Période	Semestre 7	Volume horaire	31.5
		ECTS	3

Responsable	Ines Mouakher	email	Ines.mouakher@fseg.utm.tn
Equipe pédagogique	Ines Mouakher		

1. Objectifs de Module (*Savoirs, aptitudes et compétences*)

Ce module porte sur des paradigmes avancés de l'analyse et de la conception par objets qui s'appuie sur un ensemble de principes, de règles et de patrons de conception (design patterns). De plus, il permet d'élargir les connaissances des élèves en utilisant un langage support orienté objet, Java ou C++. Ce cours aborde également la notion de qualité du processus de développement en introduisant les méthodes de refactoring qui nous permettent d'obtenir un code simple, maintenable et évolutif.

Acquis d'apprentissage :

A la fin de cet enseignement, l'élève sera capable de :

- Maîtriser, des paradigmes avancés de l'analyse et de la conception par objets (**C1.2**)
- Caractériser la qualité de la conception par objet ainsi que l'identification des bad smell dans le code (**C1.2**)
- Simuler et tester les solutions élaborées en appliquant des designs patterns et des techniques de refactoring (**C1.3**)
- Concevoir un code orienté objet simple, maintenable et évolutif. (**C1.1**)

2. Pré-requis(*autres UE et compétences indispensables pour suivre l'UE concernée*)

- Connaissances de base d'un langage Objet (Java ou C++) .
- Connaissances de la modélisation Objet avec le langage UML

3. Répartition d'Horaire de Module

<i>Intitulé de l'élément d'enseignement</i>	<i>Total</i>	<i>Cours</i>	<i>TD</i>	<i>Atelier</i>	<i>PR</i>
Module : Design Patterns	31.5	13.5	7.5	10.5	0

4. Méthodes pédagogiques et moyens spécifiques au Module

(*pédagogie d'enseignement, ouvrages de références, outils matériels et logiciels*)

- Pédagogie d'enseignement : Pédagogie interactive
- Méthodologie de travail : Cours magistral, travaux dirigés, Atelier
- Outils matériels et logiciels : Vidéoprojecteur, tableau blanc, Supports de Cours

Bibliographie		
Titre	Auteur(s)	Edition
Design Patterns : Elements of Reusable Object-Oriented Software	Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides	Addison-Wesley; 1995
Design Patterns en Java	Laurent Debrauwer	Eni Editions, Mai 2013
Agile Software Development: Principles, Patterns, and Practices.	Robert C. Martin	Prentice Hall; 2002
Refactoring: Improving the Design of Existing Code	Martin Fowler	Addison-Wesley, 2000

5. Contenu (<i>Descriptifs et plans des cours / Déroulement / Détail de l'évaluation de l'activité pratique</i>)	Durée allouée	
Module 1 : Design Patterns		
Séance 1 <ul style="list-style-type: none"> ● Présentation générale <ul style="list-style-type: none"> ○ Fondements ○ La conception dans les processus agiles ○ Familles de patterns (GoF, Grasp) ○ Les anti-patterns 	Cours	1h30 H
Séance 2 <ul style="list-style-type: none"> ● Qualité de Conception dans le développement orienté objet <ul style="list-style-type: none"> ○ Notions de cohésion et de couplage ○ Principes SOLID <ul style="list-style-type: none"> ▪ Ouvert/fermé ▪ Responsabilité unique ▪ Substitution de Liskov ▪ Séparation des interfaces ▪ L'inversion des dépendances 	Cours/T D	3H
Séance 3 <ul style="list-style-type: none"> ● Les Patterns créateurs (Creational Patterns) <ul style="list-style-type: none"> ○ Singleton (Singleton) ○ Fabrique (Factory Method) ○ Fabrique Abstraite (Abstract Factory) ○ Monteur (Builder) ○ Prototype (Prototype) 	Cours	3H
Séance 4 <ul style="list-style-type: none"> ● Les Patterns de création (Creational Patterns) 	TD/ Atelier	3H

<ul style="list-style-type: none"> ○ Savoir reconnaître et implémenter les principaux patterns de création ○ Implémentation des patterns en Java ou C++ 		
Séance 5 <ul style="list-style-type: none"> ● Les Patterns structuraux (Structural Patterns) <ul style="list-style-type: none"> ○ Composite (Composite) ○ Adaptateur (Adapter) ○ Décorateur (Decorator) ○ Façade (Facade) ○ Proxy (Proxy) ○ Poids-mouche (Flyweight) ○ Pont (Bridge) 	Cours	3H
Séance 6 <ul style="list-style-type: none"> ● Les Patterns structuraux (Structural Patterns) <ul style="list-style-type: none"> ○ Savoir reconnaître et implémenter les principaux patterns de structure ○ Implémentation des patterns en Java ou C++ 	TD/ Atelier	3H
Séance 7 <ul style="list-style-type: none"> ● Les Patterns structuraux (Structural Patterns) 	TD/ Atelier	
Séance 8 <ul style="list-style-type: none"> ● Les Patterns comportementaux (BehavioralPatterns) <ul style="list-style-type: none"> ○ Itérateur (Iterator) ○ Stratégie (Strategy) ○ Patron de méthode (Template Method) ○ État (State) ○ Observateur (Observer) ○ Commande (Command) ○ Visiteur (Visitor) ○ Chaîne de responsabilité (Chain of responsibility) ○ Interpréteur (Interpreter) ○ Médiateur (Mediator) ○ Memento (Memento) 	Cours	3H
Séance 9 <ul style="list-style-type: none"> ● Les Patterns comportementaux (BehavioralPatterns) <ul style="list-style-type: none"> ○ Savoir reconnaître et implémenter les principaux patterns de comportement ○ Implémentation des patterns en Java ou C++ 	TD/ Atelier	3H
Séance 10 <ul style="list-style-type: none"> ● Les Patterns comportementaux (BehavioralPatterns) <ul style="list-style-type: none"> ○ Savoir reconnaître et implémenter les principaux patterns de comportement ○ Implémentation des patterns en Java ou C++ 	Atelier	3H

Séance 11

- Le refactoring
 - Définition et principes du refactoring
 - Quelques « mauvaises odeurs » (bad smell)
 - Exemples de refactoring
 - Refactoring vers les Design Patterns

 Cours/A
telier 3H

6. Mode d'évaluation de Module(*nombre, types et pondération des contrôles*)

<i>Eléments d'enseignement</i>	Coeff	DS	EX	TP	PR
Module – Design Patterns	2	30%	50%	20%	0%

Pour valider le module, les étudiants passeront un examen dont le coefficient est de 50%, un TP dont le coefficient est de 20% et un DS dont le coefficient est 30%.

La durée de tous les examens (Examen, DS...) est de 1h30.

Quant à le DS, il est planifié après l'écoulement des 7 semaines et portera sur toutes les thématiques enseignées tout au long des 21 heures.

Quant à l'examen, il est planifié après l'écoulement des 11 semaines et portera sur toutes les thématiques enseignées tout au long des 31.5 heures.

Concernant le TP, il est planifié une semaine avant l'examen portera sur les connaissances acquises tout au long du module.

Le module est validé si l'étudiant obtient une moyenne supérieure ou égal à 10 sur 20.