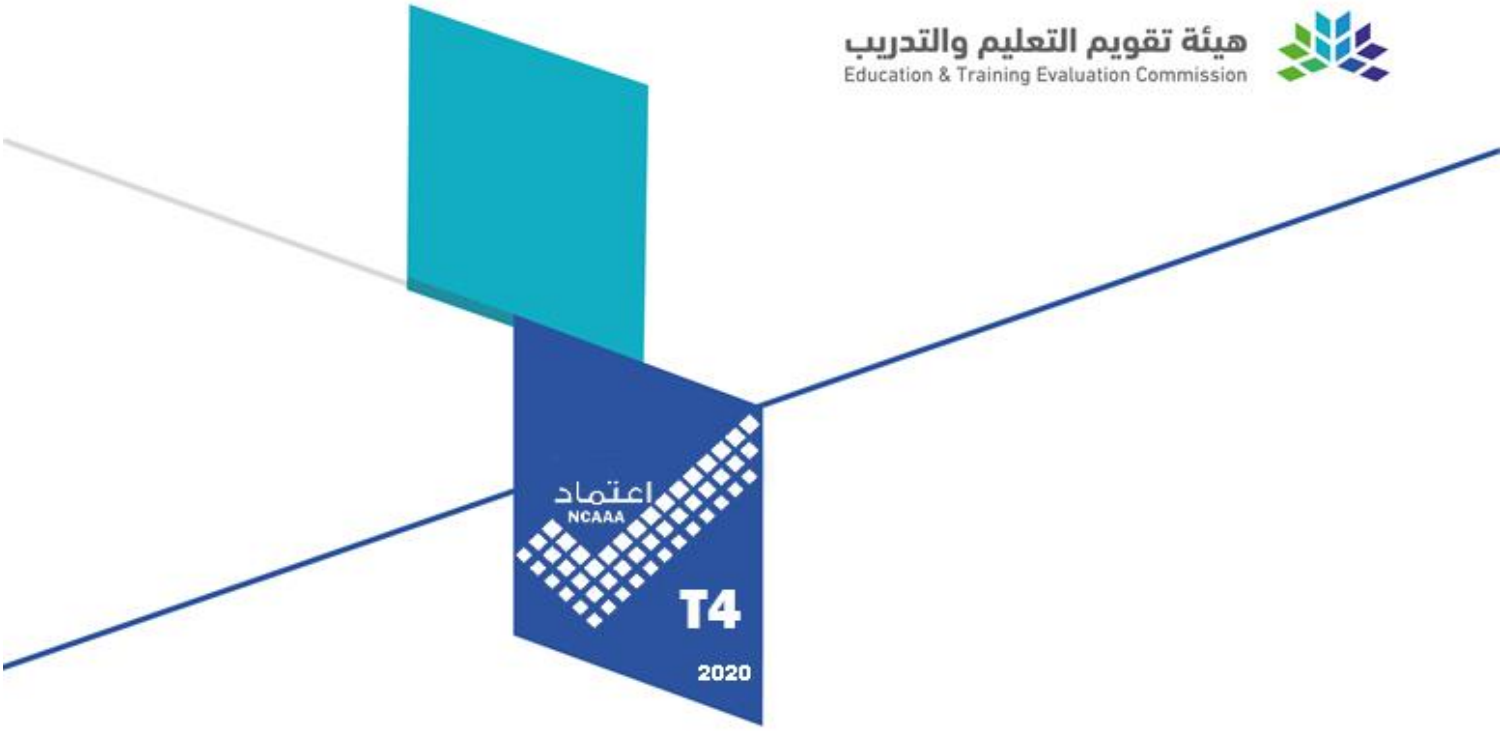
هيئة تقويم التعليم والتدريب
Education & Training Evaluation Commission



Course Specifications

Course Title:	**Software Design Patterns**
Course Code:	**CPIT252**
Program:	**Bachelor of Science in Information Technology**
Department:	**Information Technology**
College:	**Faculty of Computing and Information Technology**
Institution:	**Northern Border University, Rafha**

Table of Contents

A. Course Identification

1. Credit hours: 3
2. Course type **a.** University ☐ College ☐ Department ☒ Others ☐ **b.** Required ☒ Elective ☐
3. Level/year at which this course is offered: Level 9/Year 3
4. Pre-requisites for this course (if any)**:** CPIT251 – Software Engineering 1
5. Co-requisites for this course (if any)**:** Nil

6. Mode of Instruction (mark all that apply)

No	Mode of Instruction	Contact Hours	Percentage
1	**Traditional classroom**	60	100%
2	**Blended**		
3	**E-learning**		
4	**Distance learning**		
5	**Other**		

7. Contact Hours (based on academic semester)

No	Activity	Contact Hours
1	**Lecture**	45
2	**Laboratory/Studio**	15
3	**Tutorial**	
4	**Others** (specify)	
	Total	60

B. Course Objectives and Learning Outcomes

1. Course Description

This course is designed to provide students with skills and competences related to the use and application of software design patterns in real life design problems. It discusses the history of software design patterns and their categories (creational, structural, and behavioral). For each software design pattern students will know the circumstances in which it is applied and the consequences of its application as well as the structure and the participating classes and/or objects and their responsibilities inside the pattern. The course uses an object-oriented programming language (i.e., Java) for the practical application and implementation of the software design patterns.

2. Course Main Objective

The objective of this course is to study the software design patterns and their categories. Students will be able to choose, apply and discuss the application of a software design pattern in a given real world design problem.

3. Course Learning Outcomes

CLOs		Aligned PLOs
1	**Knowledge and Understanding:**	
1.1	Identify the main stages in the history of design patterns and the various software design patterns and their categories according to the GoF catalog.	K1
1.2	Recognize the circumstances in which a software design pattern is applied and the consequences of its application.	K2
2	**Skills:**	
2.1	Analyze the need and effectiveness of applying a software design pattern in a given situation to resolve a design related problem.	S1
2.2	Apply software design patterns in appropriate context by considering constraints related to user needs and requirements.	S2
2.3	Use and manipulate software design patterns either in the design or the implementation phases using respectively UML notions and a programming language like Java.	S3
3	**Values:**	

C. Course Content

No	List of Topics	Contact Hours
1	Introduction, UML, and Java Revision	6
2	History and principles of software design patterns	3
3	Creational Design Patterns (Singleton, Factory Method, Abstract Factory, …)	12
4	Structural Design Patterns (Adapter, Decorator, …)	12
5	Behavioral Design Patterns (Command, Observer, …)	12
Laboratory Works		
1	UML and Java implementation	1
2	Singleton Design Pattern	2
3	Factory Method Design Pattern	2
4	Abstract Factory Design Pattern	2
5	Adapter Design Pattern	2
6	Decorator Design Pattern	2
7	Command Design Pattern	2
8	Observer Design Pattern	2
Total		60

D. Teaching and Assessment

1. Alignment of Course Learning Outcomes with Teaching Strategies and Assessment Methods

Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
1.0	**Knowledge and Understanding**		
1.1	Identify the main stages in the history of design patterns and the various software design patterns and their	- KWLH (Know, Want, Learned, How) technique	- Writing - Oral

Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
	categories according to the GoF catalog.	- Class / Group discussion	
1.2	Recognize the circumstances in which a software design pattern is applied and the consequences of its application.	- KWLH (Know, Want, Learned, How) technique - Class / Group discussion	- Writing - Oral
2.0	**Skills**		
2.1	Analyze the need and effectiveness of applying a software design pattern in a given situation to resolve a design related problem.	- Problem-solving - Model-based learning - Lab-based learning	- Writing - Oral - Performance - Observation
2.2	Apply software design patterns in appropriate context by considering constraints related to user needs and requirements.	- Problem-solving - Model-based learning - Lab-based learning	- Writing - Oral - Performance - Observation
2.3	Use and manipulate software design patterns either in the design or the implementation phases using respectively UML notions and a programming language like Java.	- Problem-solving - Model-based learning - Lab-based learning	- Writing - Oral - Performance - Observation
3.0	**Values**		

2. Assessment Tasks for Students

#	Assessment task*	Week Due	Percentage of Total Assessment Score
1	Quiz-1	2	2
2	Quiz-2	8	3
3	Assignment-1	4	5
4	Assignment-2	10	5
5	Oral questions	1 – 12	5
6	Lab tasks	1 – 11	16
7	Midterm exam	6	20
8	Lab Exam	12	4
9	Final exam	13	40

***Assessment task** (i.e., written test, oral test, oral presentation, group project, essay, etc.)

E. Student Academic Counseling and Support

Arrangements for availability of faculty and teaching staff for individual student consultations and academic advice:

Every instructor has an announced office hours schedule. All students are encouraged to visit the concerned teacher according to the schedule. Students can also use Email address or Blackboard System to seek help or book an appointment.

F. Learning Resources and Facilities

1.Learning Resources

Required Textbooks	Freeman, E., Robson, E., Sierra, K., & Bates, B. (2020). *Head First design patterns:* Building Extensible and Maintainable Object-Oriented Software (2nd ed.). O'Reilly.
Essential References Materials	1. Gamma, E., Helm, R., Johnson, R., & Vlissides, J., (1994). *Design Patterns*: Elements of Reusable Object-Oriented Software (1st ed.). Addison-Wesley. 2. Lavieri, E., (2019). *Hands-On Design Patterns with Java* (1st ed.). Packt.
Electronic Materials	1. Blackboard System: https://lms.nbu.edu.sa/ 2. Northern Border University Electronic Library: https://www.nbu.edu.sa/AR/Deanships/Library_Issues 3. Saudi Digital Library (SDL): https://portal.sdl.edu.sa/english/
Other Learning Materials	Nil

2. Facilities Required

Item	Resources
Accommodation (Classrooms, laboratories, demonstration rooms/labs, etc.)	• Classroom • Laboratory
Technology Resources (AV, data show, Smart Board, software, etc.)	• Data Show (Projectors) in Classroom and Laboratory. • Desktop computers • Software: Java IDE such Netbeans, Eclipse etc.
Other Resources (Specify, e.g. if specific laboratory equipment is required, list requirements or attach a list)	Nil

G. Course Quality Evaluation

Evaluation Areas/Issues	Evaluators	Evaluation Methods
Effectiveness of teaching and assessment.	Students	Indirect
Quality of learning resources	Students	Indirect
Extent of achievement of course learning outcomes	Faculty	Direct

Evaluation areas (e.g., Effectiveness of teaching and assessment, Extent of achievement of course learning outcomes, Quality of learning resources, etc.)
Evaluators (Students, Faculty, Program Leaders, Peer Reviewer, Others (specify)
Assessment Methods (Direct, Indirect)

H. Specification Approval Data

Council / Committee	Information Technology Department Council
Reference No.	10
Date	27/02/2022