Design Patterns in Java

1. **Course number and name:** 020MCJES3 Design Patterns in Java

2. **Credits and contact hours:** 4 ECTS credits, 35 contact hours (lectures + labs)

3. **Instructor's or course coordinator's name:** Rima Kilany

4. **Text book:**
   a. **Other supplemental materials:**
   Handouts and course material posted on the Web (Moodle)

5. **Specific course information**
   a. **Catalog description:**
   This course exposes the principles of Object Oriented Programming in Java. It describes: java.awt, and javax. swing librairies in order to develop graphical user interfaces, java.io to handle Inputs/Outputs, java.net to be able to develop applications that communicate by using TCP or UDP. The course details the 23 design patterns of the book: Design Patterns: Elements of Reusable Object-Oriented Software (GOF) and shows how and when to use creational/structural/behavioral design patterns in a Java application.

   b. **Prerequisites or co-requisites:**

   c. **Required:** Elective for CCE students; required for CCE software engineering option students

6. **Specific goals for the course**
   a. **The student will be able to:**
   - Develop an object-oriented Java application and use the main Java packages.
   - Make a productive use of the object-oriented design patterns in application development in order to have open, extensible and reusable applications.

   b. **KPI:**

KPI	a2	e3	k1	k2	k3
Covered	x	x	x	x	x
Assessed	x	x	x	x	x

7. **Brief list of topics to be covered**

Lecture	Description
1	Introduction to the Java language (interpreted vs compiled)
2	Java language syntax rules
3	classes, packages and encapsulation

4	static members, inheritance and polymorphism
5	Java interfaces, Upcast, Downcast, and exceptions
6-7	graphical user interfaces
8-9	inputs/outputs
10	Multithreading
11-12-13	Network Communication (UDP-TCP)
14	Introduction to design patterns
15	Introduction to UML: essential diagrams for modeling a solution with design patterns
16	The GOF design patterns: Strategy, State, Façade,
17	Singleton, Factory Method,
18	Builder, prototype,
19	Visitor, Chain of Responsibility,
20	Memento, Flyweight, Bridge,
21	Adapter, Proxy
22	Template Method, Decorator, Iterator
23	Abstract Factory, Command
24	Observer, Composite
25-28	Lab