Pokhara University

Faculty of Science and Technology

Course No.: CMP 278

Full Marks: 100

Course Title: Object Oriented Design and Modeling using UML (3-1-2)

Pass Marks: 45

Nature of the Course: Theory & Practical

Total Lectures: 45 hrs.

Level: Bachelor

Program: BE (Software)

1. Course Description

This course is designed to provide the fundamental concepts of object oriented analysis and design of software systems using the tools of Unified Modeling Language. This provides the students with knowledge of how design of software processes starts and how the UML is used in modeling every aspect of the whole development process.

2. General Objectives

- To familiarize the students with the fundamental concepts of Object Oriented Analysis and Design.
- To provide students with the knowledge of Object-Oriented Modeling language - the Unified Modeling Language (UML).
- To develop the skills in the students to design the software system in object-oriented framework using UML.

3. Methods of Instruction

Lectures, Tutorials, Case Studies, Discussion, Readings and Practical Works.

4. Contents in Detail

Specific Objectives	Contents
Understand the basic Object Oriented concepts.	**Unit 1: Object Oriented Fundamentals (10 hours)** 1.1. Introduction 1.2. Object Oriented Analysis and Design 1.3. Overview of UML: Fundamentals and Notations 1.4. Review of Software Process Models - Waterfall, Iterative, Evolutionary Models - Agile Model 1.5. Unified Process 1.6. Inception and Use Cases 1.7. Use Case Diagrams and Activity Diagrams 1.8. Object Oriented Development Cycle
Analyze systems and develop models using object-oriented approaches.	**Unit 2: Object Oriented Analysis (12 hours)** 2.1. Building Conceptual Model 2.2. Adding Associations and Attributes 2.3. Representation of System Behaviors 2.4. Package Diagrams 2.5. Developing Class Diagrams from Domain Model
Design systems using an object oriented approach.	**Unit 3: Object Oriented Design (14 hours)** 3.1. Analysis to Design 3.2. Describing and Elaborating Use Cases 3.3. Interaction Diagrams - Sequence Diagrams - Communication Diagrams 3.4. Objects and Patterns - Introduction to GRASP Pattern 3.5. Determining Visibility 3.6. Updating Class Diagrams 3.7. Concept of Object Diagrams and State Diagrams

Specific Objectives	Contents
Develop and implement code from design documents.	**Unit 4: Implementation (9 hours)** 4.1. Programming and Iterative, Evolutionary Development 4.2. Mapping Design into Codes 4.3. Creating Class Definition from Design Class Diagrams 4.4. Creating Methods from Interaction Diagrams 4.5. Collection Classes in Code 4.6. Exceptions and Error Handling 4.7. Order of Implementation 4.8. Introduction to Test Driven Development 4.9. Concept of Deployment Diagram and Component Diagram