Course Title: Software Engineering and Project Management

Course No. : ICT Ed. 457 Nature of course: Theory + Project

Level: Bachelor Credit Hour: 3 hours (2T+1P)

Semester: Fifth Teaching Hour: 64 hours (32+32)

---

1. Course Description

The purpose of this course is to introduce the fundamental concepts of Software and professional development techniques. It aims to provide in depth knowledge regarding process models, agile development, requirement engineering, software design, software validation, software evolution and maintenance including software management terminologies.

2. General Objectives

Through this course, students shall be able:

- to evaluate and relate different software processes, system models and architectural designs and assess their suitability in a given context
- to describe basic concepts and principles of requirements engineering, software implementation, testing and maintenance
- to describe the software configuration process and quality assurance
- to apply the software project manage practices and principle in software development.

3. Course Outlines:

Specific Objectives	Contents	Lecture Hours
• Understand the concept of professional development and software engineering ethics • Study different types of system through case studies	**Unit 1: Introduction to Software Engineering** 1.1. Professional Software Development 1.2. Software Engineering Ethics **Case Studies** • Prepare summary report of the following case studies and present it in the classroom: -An Embedded System -An Information System -A Sensor Based Data Collection System -A Support Environment **Practical Works** • Visit any local organization to identify problem of their business process • Discuss with your class teacher and prepare software project proposal to address the problem identified	3 + 6
• Describe the types of software development process • Comparison of different software process model	**Unit 2: Software Processes and Agile Software Development** 2.1. Software Process Models 2.2. Process Activities 2.3. Coping with Change 2.4. Agile Methods 2.5. Agile Development Techniques	6

• Handle changing business environment with software change management • Understand agile development models and agile project management	2.6. Agile Project Management 2.7. Scaling Agile Method	
• Identify types of requirements • Explain requirement engineering process	**Unit 3: Requirements Engineering** 3.1. Functional and Non-Functional Requirements 3.2. Requirements Engineering Processes 3.3. Requirements Elicitation 3.4. Requirements Specification 3.5. Requirements Validation **Practical Works** • Gather functional requirements for the software project and prepare requirement document	**4 + 4**
• Understand importance of software design • Discuss about different design models • Make distinction between different architectural patterns and application architectures	**Unit 4: Architectural Design And System Modeling** 4.1. Context Models 4.2. Interaction Models 4.3. Structural Models 4.4. Behavioural Models 4.5. Architectural Design Decisions 4.6. Application Architectures **Practical Works** • Prepare design document for the software project	**6 + 4**
• Understand the need of software validation • Discuss different stages in testing and its process • Understand the concept of software evolution process • Understand Software maintenance • Understand the concepts of legacy system	**Unit 5: Software Testing and Software Evolution** 5.1. Development Testing 5.2. Test-Driven Development 5.3. Release Testing 5.4. User Testing 5.5. Evolution Processes 5.6. Legacy Systems 5.7. Software Maintenance **Practical Works** • Prepare test case document for the software project	**5 + 4**
• Explain software project management and planning • Discuss about project estimation techniques	**Unit 6: Software Management** 6.1. Project Management 6.1.1 Risk Management 6.1.2 Managing People	**8**

• Understand the COCOMO model • Discuss about risk management • Know about software management: quality, software standards, version management, change management and version management	6.1.3 Teamwork 6.2. Project Planning 6.2.1 Software Pricing 6.2.2 Project Scheduling 6.2.3 Agile Planning 6.2.4 Estimation Techniques 6.3. Software Quality and Standards 6.4. Version Management 6.5. Change Management 6.6. Release Management	
• Transform theoretical knowledge to solve real world problems	**Unit 7: Software Engineering Project (Practical Works)** 7.1 Design and develop software project in any of the high level language for partial fulfillment of the course Software Engineering.	**14**

4 Instructional Techniques

The instructional techniques for this course are divided into two groups. First group consists of general instructional techniques applicable to most of the units. The second group consists of specific instructional techniques applicable to particular units.

4.1 General Techniques

Reading materials will be provided to students in each unit. Lecture, Discussion, use of multi-media projector, brain storming are used in all units.

4.2 Specific Instructional Techniques

Demonstration is an essential instructional technique for all units in this course during teaching learning process. Specifically, demonstration with practical works will be specific instructional technique in this course. The details of suggested instructional techniques are presented below:

Unit 1: Self reading, and making study reports
Unit 2: Comparison about different software process model and Assign group discussion task about agile development models
Unit 3: Homework and Assignment on Requirement engineering process
Unit 4: Homework and Assignment to design different system models
Unit 5: Group Discussion on Software testing strategies, Software Evolution and legacy systems
Unit 6: Self reading, creating and presenting on different topics related to software management
Unit 7: Assign to develop a software that can solve real world problem

5 Evaluation

Internal Assessment	External Project Demo Exam/Viva	Semester Examination	Total Marks
40 Points	20 Points	40 Points	100 Points

***Note**: Students must pass separately in internal assessment, external practical exam and semester examination.*

a. Internal Evaluation (40 Points):

Internal evaluation will be conducted by subject teacher based on following criteria:

1)	Class Attendance	5 points
2)	Learning activities and class performance	5 points
3)	First assignment (written assignment)	10 points
4)	Second assignment (Case Study/project work with presentation)	10 points
5)	Terminal Examination	10 Points
	Total	40 points

b. Semester Examination (40 Points)

Examination Division, Dean office will conduct final examination at the end of semester.

1)	Objective question (Multiple choice 10 questions x 1mark)	10 Points
2)	Subjective answer questions (6 questions x 5 marks)	30 Points
	Total	40 points

c. **External Project Demonstration Exam/Viva (20 Points):**

External Examination will be evaluated on the basis of following:

1. Project Report: 10 points
2. Project Demonstration: 5 points
3. VIVA: 5 points

6 Recommended books and References materials (including relevant published articles in national and international journals)

Prescribed Text Book:

Ian Sommerville. (2015). *Software Engineering* (10th Ed.). Pearson Education

Ian Sommerville (2020). *Engineering Software Products: An Introduction to Modern Software Engineering*, Pearson Education

References materials:

Pressman, R. S. (2010). *Software Engineering: A practitioner's Approach*, 7th Ed. Boston, Mass: McGraw Hill.

John Ousterhout (2021). *A Philosophy of Software Design*, 2nd Ed.