Program Content

Semester	II		
Course Code:	IT2206		
Course Name:	Fundamentals of Software Engineering		
Credit Value:	4		
Core/Optional	Core		
Hourly Breakdown	Theory	Practical	Independent Learning
	45	30	125

Course Aim:

- The main objective of this course is to provide a broad understanding of the software engineering process, concepts and the systematic development and management of software projects.

Intended Learning Outcomes:

After following this course, students should be able to

- explain the software engineering principles and techniques that are used in developing quality software products.
- apply software engineering principles and techniques appropriately to develop a moderately complex software system.

Course Content: (Main Topics, Sub topics)

Topic	Theory (Hrs)	Practical (Hrs.)
1. Introduction	5	2
2. Software Processes	6	4
3. Agile Software Development	3	2
4. Requirement Engineering	7	6
5. System Modeling	6	4
6. Architectural Design	6	4
7. Design and Implementation	3	2
8. Software Testing	6	4
9. Software Evolution	3	2
Total	45	30

1. Introduction (5 hours)

1.1. Professional software development [Ref 1: Pg. (19-28)]
1.2. Software engineering ethics [Ref 1: Pg. (28-31)]
1.3. Case studies [Ref 1: Pg. (31-39)]

2. Software processes (6 hours)

2.1. Software process models [Ref 1: Pg. (45-54)]
2.2. Process activities [Ref 1: Pg. (54-61)]

2.3. Coping with change [Ref 1: Pg. (61-65)]
2.4. Process improvement [Ref 1: Pg. (65-68)]

3. Agile software development (3 hours)
3.1. Agile methods [Ref 1: Pg. (75-77)]
3.2. Agile development techniques [Ref 1: Pg. (77-84)]
3.3. Agile project management [Ref 1: Pg. (84-88)]
3.4. Scaling agile methods [Ref 1: Pg. (88-97)]

4. Requirements engineering (7 hours)
4.1. Functional and non-functional requirements [Ref 1: Pg. (105-111)]
4.2. Requirements engineering processes [Ref 1: Pg. (111-112)]
4.3. Requirements elicitation [Ref 1: Pg. (112-120)]
4.4. Requirements specification [Ref 1: Pg. (120-128)]
4.5. Requirements validation [Ref 1: Pg. (129-130)]
4.6. Requirements change [Ref 1: Pg. (130-134)]

5. System modeling (6 hours)
5.1. Context models [Ref 1: Pg. (141-144)]
5.2. Interaction models [Ref 1: Pg. (144-149)]
5.3. Structural models [Ref 1: Pg. (149-153)]
5.4. Behavioral models [Ref 1: Pg. (154-159)]
5.5. Model-driven architecture [Ref 1: Pg. (159-162)]

6. Architectural design (6 hours)
6.1. Architectural design decisions [Ref 1: Pg. (171-173)]
6.2. Architectural views [Ref 1: Pg. (173-175)]
6.3. Architectural patterns [Ref 1: Pg. (175-184)]
6.4. Application architectures [Ref 1: Pg. (184-191)]

7. Design and implementation (3 hours)
7.1. Object-oriented design using the UML [Ref 1: Pg. (198-209)]
7.2. Design patterns [Ref 1: Pg. (209-212)]
7.3. Implementation issues [Ref 1: Pg. (212-218)]
7.4. Open-source development [Ref 1: Pg. (219-221)]

8. Software testing (6 hours)
8.1. Development testing [Ref 1: Pg. (231-242)]
8.2. Test-driven development [Ref 1: Pg. (242-245)]
8.3. Release testing [Ref 1: Pg. (245-248)]
8.4. User testing [Ref 1: Pg. (249-251)]

9. Software evolution (3 hours)
9.1. Evolution processes [Ref 1: Pg. (258-261)]
9.2. Legacy systems [Ref 1: Pg. (261-270)]
9.3. Software maintenance [Ref 1: Pg. (270-279)]

Teaching /Learning Methods:
You can access all learning materials and this syllabus in the VLE: http://vle.bit.lk/, if you are a registered student of the BIT degree program. It is important to participate in learning activities given in the VLE to learn this course.

Assessment Strategy:

Continuous Assessments/Assignments:
The assignments consist of two quizzes, assignment quiz 1 (It covers the first half of the syllabus) and assignment quiz 2 (It covers the second half of the syllabus). The maximum mark for a question is 10 and the minimum mark for a question is 0 (irrespective of negative scores). Final assignment mark is calculated considering both assignments, and students will have to obtain at least 40% for each assignment. Students are advised to complete online assignments before the given deadline. It is compulsory to pass all online assignments to qualify to obtain the Level I, Diploma in IT (DIT), certificate.
In the course, case studies/Lab sheets will be introduced, and students have to participate in the learning activities.

Final Exam:
The final examination of the course will be held at the end of the semester. The paper consists of 40 MCQs and candidates have to answer all the 40 questions within 2 hours.

References/ Reading Materials:

- **Ref 1.** Sommerville Ian, Software Engineering, 10th Edition. Pearson.