University of Computer Studies, Yangon

Faculty of Information Science

2018-2019 Academic Year

Subject Code	CS-405	Subject Name	Unified Modeling Language (UML)
Semester	Second	Course Coordinator	Dr Khine Khine Oo
Credit	3		
Weeks	16 weeks		
Total Hour	40 Hours		
No of periods	48 Periods (1 period :50 Mins)/ 3 periods per week		

Course Description

Regardless of the software development approach, from the classic waterfall to extreme programming (XP), all of the experts agree that quality software development requires both analysis and design. The Unified Modeling Language (UML) provides a common, standard notation for recording both analysis models and design artifacts. This course delves into the processes of both object-oriented analysis and object-oriented design using UML as the notation language. The course will be supported by practical lectures involving the development of object-oriented design for a real world application.

Course Objective

- To understand the importance of modeling in the software development life cycle
- To understand the usage of UML notation and symbols
- To analyze and design systems and software solutions using the object-oriented approach
- To Employ the UML notation to create effective and efficient system designs

Learning Outcomes

- Gain a working knowledge Star UML
- Apply knowledge of OOAD to design an object-oriented system which can be implemented in an object-oriented language
- Learn the meaning of every UML notation and when to use them
- Learn the industry best practices to build UML
- Interpret any UML diagram effectively
- Increase the knowledge of software development productivity

Major Topic covered in the course

- Introduction to UML
- Modelling with Objects
- Class and Object Diagrams
- Interaction Diagrams
- Business Modelling
- Analysis
- Design
- Implementation

- State Charts
- Component Diagram

Overview of Learning Activities

- **Lectures**: Key concepts will be explained in lectures in which course material will be presented and the subject matter will be illustrated with examples.
- **Tutorial sessions**: Focus on analyzing and problem solving of given system.
- **Practical sessions**: Provide practice for developing the design with Star UML software

TextBook

- Mark Priestley, “Practical Object-oriented Design with UML” 2nd Edition

Reference Book

- GradyBooch ,James Rumbaugh ,Ivar Jacobson, “Unified Modeling Language User Guide”, Addison Wesley ,First Edition October 20, 1998, ISBN: 0-201-57168-4.

Tools

1. Star UML

Learning Assessment

Paper Exam	:	60%
Practical Assessment	:	10%
Project	:	10%
Quiz	:	10%
Assignment	:	5%
Class Participation	:	5%

Course Policies

- Individual deliverables are to be submitted individually and group work is collaborative.
- Class sessions will be interactive, though guided by the instructor with the help of a number of questions, as well as small cases and problems.
- All assignments are to be completed by the student alone with no help from any other person.
- Students are allowed to discuss about homework and project problems with others.
- Students are not allowed to copy the solutions from another colleague.

CS-405 **: Unified Modeling Languages** **Second Semester**

Text Book **:** Practical Object Oriented Design with UML (2nd Edition)

Period **:** 48 periods for 15 Weeks (3 period * 16 weeks)

No.	Chapter	Page	Period	Remark
	Chapter 1 Introduction to UML			
1.	1.1 Model and modelling 1.2 Methodology 1.3 Unified Modeling Language 1.4 Design model and code	1-11	1	Detail
	Chapter 2 Modelling with objects			
2.	2.1 The object model 2.2 Classes and Objects 2.3 Object properties	14-19	1	Detail
3.	2.4Avoiding data replication 2.5 Link 2.6 Association 2.7 Message passing	21-25	1	Detail
4.	2.8 Polymorphism 2.9 Dynamic binding Exercise	27-32	1	Detail
	Chapter 8 Class and Object Diagram			
5	8.1 Data type 8.2 Classes 8.3 Describing objects with classes	142-145	1	Detail
6.	8.4 Association 8.5 Generalization and Specialization	150-156	1	Detail
7.	8.6 Inheritance of attributes and operation 8.7 Aggregation 8.8 Composition	160-166	1	Detail
8.	8.9 Association classes 8.10 N-ary association 8.11 Qualified Association 8.12 Interfaces	168-176	1	Detail

	8.13 Template			
9.	Exercise	178	1	Draw ClassDiagram
	Chapter 9 Interaction Diagram			
10.	9.1 Collaboration 9.2 Classifier Rule 9.3 Association Rule	187-190	1	Detail
11.	9.4 Interaction diagram 9.5 Object creation 9.6 Object destruction	192-196	1	Detail
12.	9.7 Role multiplicity and iterated messages 9.8 Multi object 9.9 Conditional message 9.10 Messages to self Exercise	197-203	2	Draw Sequence and Collaboration Diagrams
	Chapter 4 Business modelling			
13.	4.1 Informal requirement 4.2UsecaseModelling 4.3 Describing Usecase	53-58	1	Detail
14.	4.4 Structuring the Usecase Model 4.5 Completing the Usecase Model	61-65	1	Detail
15.	4.6 Domain Modelling 4.7 Glossaries Exercises	67-72	1	Draw Use case Diagram
	Chapter 5 Analysis			
16.	5.1 Purpose of Analysis 5.2 Object design 5.3 Software Architecture	75-78	1	Detail
17.	5.4Usecase Realization	82	1	Detail
18.	5.5 Recording new booking 5.6 Canceling booking 5.7 Updating booking	88-92	1	Detail
19.	5.8 Completing the analysis model	94	1	Detail

20.	Exercises	96	1	
	Chapter 6 Design			
21.	6.1 Receiving input from the user 6.2 Producing output 6.3 Persistent data storage	98-104	2	Should Know
22.	6.4 The design model 6.5 Detail class diagram 6.6 Dynamic modeling behavior, Exercise	109-111 119	1	Should Know
	Chapter 7 Implementation			
23.	7.1 Implementation diagram 7.2 Implementation strategy 7.3 Application framework	121-125	1	Should Know
24.	7.5 Implementation of classes 7.6Implementation of association 7.7Implementation of operation Exercises	130-137 140	1	Should Know
	Chapter 10 State Chart			
25.	10.1 State dependent behavior 10.2 State event and transition	209-210	1	Detail
26.	10.3 Initial and final state 10.4 Guard condition	211-212	1	Detail
27.	10.5 Action 10.6 Activity 10.7 Composite states 10.8History state	214-221	1	Detail
28.	10.10 Dynamic modeling in practice 10.11 Time events 10.12 Activity state	221-228	1	Detail
29.	Exercises	231	2	Draw State Chart Diagram
	Chapter 11 Component diagram			
30.	11.1 Dependencies 11.2 Component and Artifacts 11.3 Component diagrams	236-239	1	Detail

31.	11.4 Some common physical relationships 11.5 Compilation dependency Exercises	239-241 246	1	Detail
32.	Revision		3	
33.	Project		8	
34.	Review		3	