



Course Specifications

Course Title:	Software Architecture
Course Code:	502464-3
Program:	Bachelor in Information Technology
Department:	Department of Information Technology
College:	College of Computers and Information Technology
Institution:	Taif University

Table of Contents

A. Course Identification	3	
6. Mode of Instruction (mark all that apply)		3
B. Course Objectives and Learning Outcomes	3	
1. Course Description		3
2. Course Main Objective		3
3. Course Learning Outcomes		3
C. Course Content	4	
D. Teaching and Assessment	4	
1. Alignment of Course Learning Outcomes with Teaching Strategies and Assessment Methods		4
2. Assessment Tasks for Students		4
E. Student Academic Counseling and Support	5	
F. Learning Resources and Facilities	5	
1. Learning Resources		5
2. Facilities Required		5
G. Course Quality Evaluation	5	
H. Specification Approval Data	6	



A. Course Identification

1. Credit hours: 3
2. Course type a. University <input type="checkbox"/> College <input type="checkbox"/> Department <input checked="" type="checkbox"/> Others <input type="checkbox"/> b. Required <input checked="" type="checkbox"/> Elective <input type="checkbox"/>
3. Level/year at which this course is offered: 12/4
4. Pre-requisites for this course (if any): 502435-3
5. Co-requisites for this course (if any): non

6. Mode of Instruction (mark all that apply)

No	Mode of Instruction	Contact Hours	Percentage
1	Traditional classroom	7	100%
2	Blended	0	0
3	E-learning	0	0
4	Distance learning	0	0
5	Other	0	0

7. Contact Hours (based on academic semester)

No	Activity	Contact Hours
1	Lecture	50
2	Laboratory/Studio	20
3	Tutorial	0
4	Others (specify)	0
	Total	70

B. Course Objectives and Learning Outcomes

1. Course Description

This course introduces basic concepts and principles about software architecture and software architecture design. Practical approaches and methods for creating and analysing software architecture are presented with emphasis on the interaction between quality attributes and software architecture. Students will also gain experiences with examples in design pattern application and case studies in software architectures.

2. Course Main Objective

The main objective of this course is to learn the students basics of software architecture and requirements analysis, methods to model and design software architecture, domain analysis concepts, and pattern concepts.



3. Course Learning Outcomes

CLOs		Aligned PLOs
1	Knowledge and Understanding	
1.1	Recognize the concepts of software architecture including requirement analysis, domain analysis, and software architecture patterns.	K1
1.2	Identify the major software architecture description and modeling languages.	K1
1.3	Describe the requirements analysis activities and software architecture activities.	K1
2	Skills :	
2.1	Develop software architecture using different software architecture design methods considering quality attributes.	S1
2.2	Assess requirements engineering activities including mapping the architecture to a software design.	S2
3	Values:	
3.1		

C. Course Content

No	List of Topics	Contact Hours
1	Introduction to Software Architecture	5
2	Rationales for Software Architecture design	5
3	Modeling Software Architecture	10
4	Software Architecture design methods	10
5	Architecture requirements analysis	10
6	Domain analysis	10
7	Software Architecture Patterns	10
8	Mapping architecture to design	10
Total		70

D. Teaching and Assessment

1. Alignment of Course Learning Outcomes with Teaching Strategies and Assessment Methods

Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
1.0	Knowledge and Understanding		
1.1	Recognize the concepts of software architecture including requirement analysis, domain analysis, and software architecture patterns.	Lecture Discussion Lab work	Written Exams Assignments, Quizzes Practical Exam
1.2	Identify the major software architecture description and modeling languages.	Lecture Discussion Lab work	Written Exams Assignments, Quizzes Practical Exam



Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
1.3	Describe the requirements analysis activities and software architecture activities.		
2.0	Skills		
2.1	Develop software architecture using different software architecture design methods considering quality attributes.	Lecture Discussion Lab work	Written Exams Assignments, Quizzes Practical Exam
2.2	Assess requirements engineering activities including mapping the architecture to a software design.	Lecture Discussion Lab work	Written Exams Assignments, Quizzes Practical Exam
3.0	Values		
3.1			

2. Assessment Tasks for Students

#	Assessment task*	Week Due	Percentage of Total Assessment Score
1	Assignments and Quizzes	8	10%
2	Mid Exam	6	20%
3	Attendance/ class activities	weekly	10%
4	Labs	11	10%
5	Final Exam	12	50%

*Assessment task (i.e., written test, oral test, oral presentation, group project, essay, etc.)

E. Student Academic Counseling and Support

Arrangements for availability of faculty and teaching staff for individual student consultations and academic advice :

Academic advising and counseling of students is an important component of teaching; student academic advising is a mandatory requirement of College of Computers and Information Technology (CCIT). Appropriate student advising provides support needed for the student during times of difficulty. In addition, it helps the student to build a close relationship with his/her advisor and to provide student motivation and involvement with the institution.

In addition, since faculty are usually the first to recognize that a student is having difficulty, faculty members play a key role in developing solutions for the students or referring them to appropriate services. Faculty members also participate in the formal student-mentoring program.

Additional counseling is provided by course directors, who provide students with academic reinforcement and assistance and refer “at risk” students to the Vice Dean for Academic Affairs and the Vice Dean for female section.

F. Learning Resources and Facilities

1. Learning Resources

Required Textbooks	Bass, L., Clements, P. and Kazman, R, Software Architecture in Practice, Addison-Wesley, 2003
---------------------------	---



Essential References Materials	Taylor, R., Medvidovic, N., Dashofy, E, Software Architecture: Foundations, Theory, and Practice, Wiley, 2010
Electronic Materials	Presentations and recorded lectures
Other Learning Materials	<ul style="list-style-type: none"> • P. Clements, F. Bachmann, L. Bass, D. Garlan, J. Ivers, R. Little, P. Merson, R. Nord, J. Stafford, Documenting Software Architecture: Views and Beyond, Addison-Wesley Professional, 2 edition, 2010. • Some related research articles

2. Facilities Required

Item	Resources
Accommodation (Classrooms, laboratories, demonstration rooms/labs, etc.)	<ul style="list-style-type: none"> • A Lecture room appropriate for maximum 25 students with a personal computer, a data show and a smart board. • A Lab room appropriate for maximum 15 students with a personal computer, a data show and a smart board.
Technology Resources (AV, data show, Smart Board, software, etc.)	<ul style="list-style-type: none"> • Lab materials and required software & hardware
Other Resources (Specify, e.g. if specific laboratory equipment is required, list requirements or attach a list)	NON

G. Course Quality Evaluation

Evaluation Areas/Issues	Evaluators	Evaluation Methods
Effectiveness of Teaching	Students	Students surveys and Students course evaluation
Improvement of Teaching	Course Coordinator	Deficiencies based on the student Evaluation, faculty input, course file, and program assessment
Verifying Standards of Student Achievement	Curriculum Committee	<ul style="list-style-type: none"> • Review CAF (Course assessment file) • Alumni surveys. • Periodic exchange and remarking of tests or a sample of assignments with staff at another

Evaluation areas (e.g., Effectiveness of teaching and assessment, Extent of achievement of course learning outcomes, Quality of learning resources, etc.)

Evaluators (Students, Faculty, Program Leaders, Peer Reviewer, Others (specify))

Assessment Methods (Direct, Indirect)

H. Specification Approval Data

Council / Committee	IT Department Council/ Executive program committee
Reference No.	11



Handwritten signature or mark in a speech bubble shape.

