جامعــة الزيتونــة الأردنيــة
Al-Zaytoonah University of Jordan
كلية العلوم وتكنولوجيا المعلومات
Faculty of Science and IT

"Tradition and Quality"

QF01/0408-4.0E	Course Plan for Master program - Study Plan Development and Updating Procedures/ Department

Study plan No.	2021/2022	University Specialization	Software Engineering
Course No.	0104712	Course name	Advanced Software Architecture and Design
Credit Hours	3	Prerequisite Co-requisite	

Course type	☐ MANDATORY UNIVERSITY REQUIREMENT	☐ UNIVERSITY ELECTIVE REQUIREMENTS	☐ FACULTY MANDATORY REQUIREMENT	☐ Support course family requirements	✔ Mandatory requirements	☐ Elective requirements
Teaching style	☐ Full online learning		✔ Blended learning		☐ Traditional learning	
Teaching model	☐ 2Synchronous: 1asynchronous		✔ 2 face to face : 1synchronous		☐ 3 Traditional	

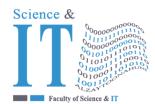**Faculty member and study divisions information (to be filled in each semester by the subject instructor)**

Name	Academic rank	Office No.	Phone No.	E-mail
Wael Alzaydat	Assistant Professor			Wael.alzyadat@zuj.edu.jo

Division number	Time	Place	Number of students	Teaching style	Approved model

Brief description

This course describes architectural patterns for various architectures, such as broker, discovery, and transaction patterns for service-oriented architectures, and addresses software quality attributes including maintainability, modifiability, testability, traceability, scalability, reusability, performance, availability, and security. Complete case studies illustrate design issues for different software architectures are also discussed in this course

Learning resources

Course book information (Title, author, date of issue, publisher ... etc)	• Mark Richards, 2015, Software Architecture Patterns, O'Reilly Media, Inc. • Sherman, S. and Hadar, I. 2015, Toward defining the role of the software architect: An examination of the soft aspects of this role, In Proceedings of the Eighth International Workshop on Cooperative and Human Aspects of Software Engineering • G. Hohpe, I. Ozkaya, U. Zhun and O. Zimmermann 2016, The Software Architect's Role in the Digital Age, IEEE Software, 33 • N. B. Harrison, P. Avgeriou and U. Zdun Using Patterns to Capture Architectural Decisions, IEEE Software, 24(4) • P. Kruchten, H. Obbink and J. Stafford 2006, The Past, Present, and Future for Software Architecture, IEEE Software, 23(2) • Shaw, M. and Clements, P. 2006, The golden age of software architecture., IEEE Software, 23(2) • G. Booch 2011, Draw Me a Picture, IEEE Software,, 28(1) • Hofstader, J. 2009, We Don't Need No Architects!, The Architecture Journal, 15 http://www.iasa.se/wp-content/uploads/2009/08/TAJ15.pdf

Supportive learning resources (Books, databases, periodicals, software, applications, others)	https://www.sei.cmu.edu/					
Supporting websites	https://www.iso.org/standard/35733.html					
The physical environment for teaching	✔ Class room	☐ labs	✔ Virtual educational platform		☐ Others	
Necessary equipment and software						
Supporting people with special needs						
For technical support						

Course learning outcomes (S= *Skills, C= Competences K= Knowledge,*)

No.	Course learning outcomes	The associated program learning output code
Knowledge		
K1	**Capacity for students will learn and understand the role of a software architect, in creating an extensible and maintainable software solution by applying abstract knowledge and well-known patterns to software architecture and design.**	**Mk3**
K2	**Use appropriate models for solution development**	**Mk2**
K3	**Students will develop an understanding on how programming languages are implemented, an invaluable step in successfully designing**	**Mk1**
K4	**applying emerging trends and paradigms in software architecture, and the challenges, risks and opportunities in migrating from a monolithic software architecture to microservices.**	**Mk2**
Skills		
S1	**The role of software architect or senior member of the development team**	**Ms1**
S2	**Students the opportunity for in-depth study of the advanced design and architectural and software development and process skills required for the successful design and development of complex software distributed systems**	**Ms2,Ms3**
S3	**Identify across all disciplines, requirements drive architecture, and the communication of the vision forms a critical component of realizing the end product or structure**	**Ms2**
Competences		
C1	**Exposed to current state-of-the art principles, methods and research of software design and architecture**	**Mc2**
C2	**Able identify and assess software vulnerabilities that impact the security of the underlying software system**	**Mc1**
C3	**Evaluate how Architecture as a Service (AaaS), serverless architecture systems and other emerging trends are impacting the field of software design and architecture.**	**Mc1, Mc2**
C4	**Design a software architecture solution from a presented case**	**Mc2,Mc3**

	study.	

Mechanisms for direct evaluation of learning outcomes

Type of assessment / learning style	Fully electronic learning	Blended learning	Traditional Learning (Theory Learning)	Traditional Learning (Practical Learning)
First exam	0	**0**	%20	0
Second / midterm exam	%30	**%30**	%20	30%
Participation / practical applications	0	**0**	10	30%
Asynchronous interactive activities	%30	**%30**	0	0
final exam	%40	**%40**	%50	40%

Note: Asynchronous interactive activities are activities, tasks, projects, assignments, research, studies, projects, work within student groups ... etc, which the student carries out on his own, through the virtual platform without a direct encounter with the subject teacher.

Schedule of simultaneous / face-to-face encounters and their topics

Week	Subject	learning style*	Reference **
1	Toward defining the role of the software architect: An examination of the soft aspects of this role	Lecture	Sherman, S. and Hadar, I. 2015, Toward defining the role of the software architect: An examination of the soft aspects of this role, In Proceedings of the Eighth
2	Layered Architecture	Lecture	Chapter 1: Mark Richards, 2015, Software Architecture Patterns, O'Reilly Media, Inc
3	A Systematic Review of System-of-Systems Architecture Research	lecture	John.K, Hans van Vliet, 2013, A Systematic Review of System-of-Systems Architecture Research, Proceedings of the 9th international ACM Sigsoft
4	Using Patterns to Capture Architectural Decisions	lecture	N. B. Harrison, P. Avgeriou and U. Zdun Using Patterns to Capture Architectural Decisions, IEEE Software, 24(4)
5	Event-Driven Architecture	Lecture	Chapter 2: Mark Richards, 2015, Software Architecture Patterns, O'Reilly Media, Inc
6	Microkernel Architecture	Lecture	Chapter 3: Mark Richards, 2015, Software Architecture Patterns, O'Reilly Media, Inc
7	Microservices Architecture Pattern	Lecture	Chapter 4: Mark Richards, 2015, Software Architecture Patterns, O'Reilly Media, Inc
8	Space-Based Architecture	Lecture	Chapter5: Mark Richards, 2015, Software Architecture Patterns, O'Reilly Media, Inc
9	**Midterm Exam**		
10	The Overall Value of Architecture Review in a	Lecture	C. Salinesi and O. Pastor (Eds.): CAiSE 2011 Workshops, LNBIP 83, pp. 302–307, 2011. © Springer-Verlag Berlin Heidelberg 2011

Science & IT
Faculty of Science & IT

جامعة الزيتونـــة الأردنيـة
Al–Zaytoonah University of Jordan
كلية العلوم وتكنولوجيا المعلومات
Faculty of Science and IT

"Tradition and Quality"

QF01/0408-4.0E	Course Plan for Master program - Study Plan Development and Updating Procedures/ Department

	Large-Scale Software Organization		
11	Enhancing Software Architecture Review Process via Knowledge Management	Lecture	Sherman, S., Hadar, I., Levy, M.: Enhancing Software Architecture Review Process via Knowledge Management. In: Proceedings of the Sixteenth Americas Conference on Information Systems, Lima, Peru (2010)
12	A framework for classifying and comparing software architecture evaluation methods	Lecture	M. A. Babar, L. Zhu and R. Jeffery, "A framework for classifying and comparing software architecture evaluation methods," 2004 Australian Software Engineering Conference. Proceedings., 2004, pp. 309-318, doi: 10.1109/ASWEC.2004.1290484.
13	Uml profiles for design decisions and non-functional requirements	Lecture	L. Zhu and I. Gorton, "Uml profiles for design decisions and non-functional requirements", In Proceedings of the Second Workshop on Sharing and Reusing Architectural Knowledge Architecture, Rationale, and Design intent, 2007, p. 8, IEEE Computer Society.
14	Architecture rationalization: a methodology for architecture verifiability, traceability and completeness	Lecture	A. Tang and J. Han, "Architecture Rationalization: A Methodology for Architecture Verifiability, Traceability and Completeness," in 12th Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems (ECBS), 2005, pp. 135-144.
15	Architecture decisions: demystifying architecture	Lecture	J. Tyree and A. Akerman, "Architecture decisions: demystifying architecture," in IEEE Software, vol. 22, no. 2, pp. 19-27, March-April 2005, doi: 10.1109/MS.2005.27.
16	**Final Exam**		

* **Learning styles: Lecture, flipped learning, learning through projects, learning through problem solving, participatory learning ... etc.**
** **Reference: Pages in a book, database, recorded lecture, content on the e-learning platform, video, website ... etc.**

Schedule of asynchronous interactive activities (in the case of e-learning and blended learning)

Week	Task / activity	Reference	Expected results
1-4	From a presented case study the student may be expected to design an architectural solution for a give system.		be expected to evaluate how emerging technological trends has and will have an impact on the design of the solution.
5-9	Presented the trend Technology used layer an architectural	Website: Software Architecture Conference https://conferences.oreilly.com/software - architecture/sa-ny Website: Serverless Conference http://serverlessconf.io/ Website: The Open Group Architecture Framework http://www.opengroup.org/subjectareas/enterprise/togaf	Realize the impact of layers

"Tradition and Quality"

QF01/0408-4.0E	**Course Plan for Master program - Study Plan Development and Updating Procedures/ Department**	
	Website: Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- System and software quality models https://www.iso.org/standard/35733.html Website: arc42 http://arc42.org/	
10-14	Pick up one of application domain such as Business information system, focus on an architecture task could be identified	Students will learn about computing as a service,