



Programme of Module "Service - Oriented Software Engineering"

- Code: DT0203
- Type of course unit: Elective (Master Degree in Computer Science curriculum GSEEM), Elective (Master Degree in Computer Science curriculum NEDAS), Compulsory (Master Degree in Computer Science curriculum SEAS), Elective (Master Degree in Computer Science curriculum UBIDIS)
- Level of course unit: Postgraduate Degrees
- Semester: 1

Number of ects credits: (Master Degree in Computer Science) 6 (workload 150 hours)

Teachers: Marco Autili (marco.autili@univaq.it)

1	Course objectives	<p>The objective of this course is to introduce Service-oriented Architecture (SOA) as a way of building distributed applications using Web Services (WS). The course aims at deepening the understanding of key aspects and principles of SOA and WS technologies, as well as related Software Engineering methodologies. At the end of the course the students will: (i) understand the notions of WSS, WS standards, and SOA; (ii) understand the service engineering development processes that can be followed to realize reusable and flexible WSs; (iii) understand how business process models and service-oriented programming models can be used as a basis for the design and the implementation of service-oriented systems; (iv) be familiar with a number of frameworks that support the development and deployment of service-oriented applications, both WS clients and WS providers; (v) have been introduced to the notion of service composition as a means for developing complex service-oriented applications.</p>
2	Course content and learning outcomes (dublin descriptors)	<p>Topics of the module include:</p> <ul style="list-style-type: none"> • Reuse-based System Development • Component-based System Development • Service Oriented Architecture (SOA) • Service-oriented Distributed System Development (WEB Services, REST Services, Microservices, Data as a Service (DaaS), Load Balancer) • SOA-enabling Technologies (XML, SOAP, WSDL, WADL, etc) • SOA-supporting Frameworks (ANT, MAVEN, SPRING WS, JAX WS, APACHE AXIS, APACHE CXF, etc.) • Service Composition • Numerous Practical Sessions in Classroom <p>On successful completion of this module, the student should :</p> <ul style="list-style-type: none"> • have profound knowledge of the notions of Web Services, Web Service standards, and Service-oriented Architecture (SOA); • have knowledge and understanding of the service engineering development processes that can be followed to realize reusable and flexible Web Services; • analyse and discuss the Service-oriented Architecture principles and the related engineering development processes; • be familiar with a number of frameworks that support the development and deployment of service-oriented applications, both WS clients and WS providers; • explain and illustrate how business process models and service-oriented programming models can be used as a basis for the design and the implementation of service-oriented systems; • explain and illustrate the notion of service composition as a means for developing complex service-oriented applications;

		<ul style="list-style-type: none"> • demonstrate skill in problem-solving, demonstrate ability to use (subset of) the programming framework to realize service-oriented applications; • demonstrate capacity of abstraction and modularity when designing service-oriented applications.
3	Course prerequisites	Object-oriented programming, Java language, and XML basics.
4	Teaching methods and language	<p>Lectures and practical exercises</p> <p>Language: English</p> <p>Reference textbooks</p> <ul style="list-style-type: none"> • Ian Sommerville, <i>Software Engineering</i>. (vol. 10th Edition) 2016. • Michael P. Papazoglou, <i>Web Services & SOA, Principles and Technology</i>. (vol. 2nd Edition) 2012.
5	Assessment methods	<p>MID-TERM TEST (OPTIONAL - to be decided during the course): either students will be assigned a homework concerning the course topics (tutorial preparation, lecture simulation, small project, ...), or students will be asked questions concerning the Software Engineering aspects learned during the first lessons including all topics of PART I and SOA Principles of PART II. FINAL TEST: Students will be given the specification of a Service-oriented System to be implemented by applying the learned service-oriented engineering principles and methods, and by using the SOA enabling technologies and the Java frameworks taught and demonstrated during the classroom activities. Students will present the system and discuss the way it has been realized. Contextually, if the mid-term test was not given or was not passed, students will be asked questions concerning the Software Engineering aspects learned during the first lessons including all topics of PART I and SOA Principles of PART II.</p>