


**Programma del Modulo "Service - Oriented Software Engineering"**

- Codice: DT0203
- Tipo di corso: Opzionale (Laurea Magistrale in Informatica percorso GSEEM), Opzionale (Laurea Magistrale in Informatica percorso NEDAS), Obbligatorio (Laurea Magistrale in Informatica percorso SEAS), Opzionale (Laurea Magistrale in Informatica percorso UBIDIS)
- Livello del corso: Lauree Magistrali
- Semestre: 1

Numero di crediti ECTS: (Laurea Magistrale in Informatica) 6 (carico 150 ore)

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

<b>1</b>	<b>Obiettivi del corso</b>	<p>L'obiettivo di questo corso è quello di introdurre le architetture orientate ai servizi (SOA) come un mezzo per costruire applicazioni distribuite che utilizzano Servizi Web (WS). Il corso si propone l'obiettivo di approfondire la conoscenza degli aspetti fondamentali e dei principi di SOA, delle tecnologie WS, così come le relative tecniche di Ingegneria del Software. Al termine del corso gli studenti: (i) capiranno le nozioni alla base di SOA e dei Servizi Web, come pure gli standard associati; (ii) comprenderanno le tecniche offerte dall'Ingegneria del Software per realizzare WS flessibili e riutilizzabili; (iii) capiranno come i modelli dei processi di business e i modelli di programmazione orientati ai servizi possono essere utilizzati come base per la progettazione e l'implementazione di sistemi orientati ai servizi; (iv) conosceranno un certo numero di framework attuali che facilitano lo sviluppo e l'implementazione di applicazioni orientate ai servizi, sia lato client sia lato server; (v) capiranno la nozione di composizione di servizi come strumento per sviluppare applicazioni orientate ai servizi più complesse.</p>
<b>2</b>	<b>Contenuti del corso e risultati formativi (descriptori di Dublino)</b>	<p>Gli argomenti trattati nel corso comprendono:</p> <ul style="list-style-type: none"> <li>• Sviluppo di Sistemi basati sul Riutilizzo del Software</li> <li>• Sviluppo di Sistemi basati a Componenti</li> <li>• Architetture Orientate ai Servizi (SOA)</li> <li>• Sviluppo di Sistemi Distribuiti orientati ai Servizi (WEB Services, REST Services, Microservices, Data as a Service (DaaS), Load Balancer)</li> <li>• Tecnologie Abilitanti per Architetture Orientate ai Servizi (XML, SOAP, WSDL, WADL, etc)</li> <li>• Framework di Sviluppo a supporto delle Architetture Orientate ai Servizi (ANT, MAVEN, SPRING WS, JAX WS, APACHE AXIS, APACHE CXF, etc.)</li> <li>• Composizione di Servizi</li> <li>• Numerose Sessioni Pratiche in Classe</li> </ul> <p>Alla fine del corso, lo studente dovrebbe:</p> <ul style="list-style-type: none"> <li>• avere una profonda conoscenza delle nozioni di servizi Web, gli standard Web Service e le Service-Oriented Architecture (SOA);</li> <li>• avere la conoscenza e la comprensione dei processi di sviluppo per l'ingegneria di servizi che possono essere seguiti per realizzare servizi Web riutilizzabili e flessibili;</li> <li>• saper analizzare e discutere i principi dell'architettura orientata ai servizi e dei relativi processi di sviluppo di ingegneria;</li> <li>• avere familiarità con una serie di framework che supportano la progettazione e l'implementazione di applicazioni orientate ai servizi, sia lato client sia lato server;</li> <li>• saper spiegare e illustrare come i modelli dei processi di business e modelli di programmazione orientate ai servizi possono essere utilizzati come base per la progettazione e l'implementazione di sistemi orientati ai servizi;</li> <li>• saper spiegare e illustrare il concetto di composizione di servizi come strumento per</li> </ul>

		<p>lo sviluppo di applicazioni orientate ai servizi più complesse;</p> <ul style="list-style-type: none"> <li>• avere la capacità di risolvere problemi, dimostrando la capacità di usare (un sottoinsieme) i framework a supporto della realizzazione di applicazioni orientate ai servizi;</li> <li>• avere la capacità di astrazione e modularità nella progettazione di applicazioni orientate ai servizi.</li> </ul>
<b>3</b>	<b>Prerequisiti</b>	Programmazione orientata agli oggetti, linguaggio Java, nozioni di base di XML.
<b>4</b>	<b>Modalità e lingua di insegnamento</b>	<p>Lezioni e esercizi pratici</p> <p><b>Lingua:</b> Inglese</p> <p><b>Testi/Bibliografia</b></p> <ul style="list-style-type: none"> <li>• Ian Sommerville, <i>Software Engineering</i>. (vol. 10th Edition) 2016.</li> <li>• Michael P. Papazoglou, <i>Web Services &amp; SOA, Principles and Technology</i>. (vol. 2nd Edition) 2012.</li> </ul>
<b>5</b>	<b>Metodi di accertamento</b>	<p>TEST INTERMEDIO (OPZIONALE - da decidere durante il corso): agli studenti verranno assegnati compiti a casa relativi agli argomenti del corso (preparazione del tutorial, simulazione di lezioni, piccoli progetti, ...). In alternativa, agli studenti verranno poste domande riguardanti gli aspetti dell'ingegneria del software appresi durante le prime lezioni, compresi tutti gli argomenti della PARTE I e i Principi SOA della PARTE II</p> <p>TEST FINALE: Agli studenti verrà fornita la specifica di un sistema orientato ai servizi da realizzare applicando i principi e i metodi dell'ingegneria del software trattati in classe e utilizzando le opportune tecnologie SOA e strutture Java. Gli studenti presenteranno il sistema e discuteranno il modo in cui è stato realizzato. Contestualmente, se il test intermedio non è stato dato o non è stato superato, agli studenti verranno poste domande riguardanti gli aspetti dell'ingegneria del software appresi durante le prime lezioni, compresi tutti gli argomenti della PARTE I e i principi SOA della PARTE II.</p>