

Guía docente de la asignatura

**Fecha de aprobación por la Comisión
Académica: 13/07/2024****Desarrollo de Sistemas de
Software Basados en
Componentes y Servicios
(M50/56/2/8)****Máster**

Máster Universitario en Ingeniería Informática

MÓDULO

Tecnologías Informáticas 1

RAMA

Ingeniería y Arquitectura

**CENTRO RESPONSABLE
DEL TÍTULO**

Escuela Internacional de Posgrado

Semestre

Primero

Créditos

4

Tipo

Obligatorio

**Tipo de
enseñanza**

Presencial

PRERREQUISITOS Y/O RECOMENDACIONES

Desarrollo de sistemas software que satisfagan todos los requisitos del usuario y se comporten de forma fiable, sean asequibles de mantener y eficientes.

Conocimiento general de las teorías, principios, métodos y prácticas de la Ingeniería del Software.

Familiaridad en el uso de entornos de desarrollo integrado de software: Eclipse (preferible), NetBeans, VisualStudio,....

BREVE DESCRIPCIÓN DE CONTENIDOS (Según memoria de verificación del Máster)

Modelos de componentes, Arquitecturas orientadas a servicios, Arquitecturas dirigidas por eventos, Arquitecturas para computación de altas prestaciones, Componentes software para sistemas distribuidos, Software intermediario (Middleware), Sistemas empujados y móviles, Sistemas ubicuos e inteligencia ambiental

COMPETENCIAS

COMPETENCIAS BÁSICAS

- CB6 - Poseer y comprender conocimientos que aporten una base u oportunidad de ser originales en desarrollo y/o aplicación de ideas, a menudo en un contexto de investigación.
- CB7 - Que los estudiantes sepan aplicar los conocimientos adquiridos y su capacidad de resolución de problemas en entornos nuevos o poco conocidos dentro de contextos más amplios (o multidisciplinares) relacionados con su área de estudio.
- CB8 - Que los estudiantes sean capaces de integrar conocimientos y enfrentarse a la complejidad de formular juicios a partir de una información que, siendo incompleta o limitada, incluya reflexiones sobre las responsabilidades sociales y éticas vinculadas a la aplicación de sus conocimientos y juicios.
- CB9 - Que los estudiantes sepan comunicar sus conclusiones y los conocimientos y razones últimas que las sustentan a públicos especializados y no especializados de un modo claro y sin ambigüedades.
- CB10 - Que los estudiantes posean las habilidades de aprendizaje que les permitan continuar estudiando de un modo que habrá de ser en gran medida autodirigido o autónomo.

COMPETENCIAS GENERALES

- G01 - Capacidad para proyectar, calcular y diseñar productos, procesos e instalaciones en todos los ámbitos de la ingeniería informática.
- G04 - Capacidad para el modelado matemático, cálculo y simulación en centros tecnológicos y de ingeniería de empresa, particularmente en tareas de investigación, desarrollo e innovación en todos los ámbitos relacionados con la Ingeniería en Informática.
- G08 - Capacidad para comprender y aplicar la responsabilidad ética, la legislación y la deontología profesional de la actividad de la profesión de Ingeniero en Informática.

COMPETENCIAS ESPECÍFICAS

- CE04 - Capacidad para modelar, diseñar, definir la arquitectura, implantar, gestionar, operar, administrar y mantener aplicaciones, redes, sistemas, servicios y contenidos informáticos.
- CE05 - Capacidad de comprender y saber aplicar el funcionamiento y organización de Internet, las tecnologías y protocolos de redes de nueva generación, los modelos de componentes, software intermediario y servicios.
- CE06 - Capacidad para asegurar, gestionar, auditar y certificar la calidad de los desarrollos, procesos, sistemas, servicios, aplicaciones y productos informáticos.
- CE07 - Capacidad para diseñar, desarrollar, gestionar y evaluar mecanismos de certificación y garantía de seguridad en el tratamiento y acceso a la información en un sistema de procesamiento local o distribuido.
- CE08 - Capacidad para analizar las necesidades de información que se plantean en un entorno y llevar a cabo en todas sus etapas el proceso de construcción de un sistema de información.
- CE09 - Capacidad para diseñar y evaluar sistemas operativos y servidores, y aplicaciones y sistemas basados en computación distribuida.
- CE10 - Capacidad para comprender y poder aplicar conocimientos avanzados de computación de altas prestaciones y métodos numéricos o computacionales a problemas de ingeniería.
- CE11 - Capacidad de diseñar y desarrollar sistemas, aplicaciones y servicios informáticos



- en sistemas empotrados y ubicuos.
- CE15 – Capacidad para la creación y explotación de entornos virtuales, y para la creación, gestión y distribución de contenidos multimedia.

COMPETENCIAS TRANSVERSALES

- CT01 – Capacidad de análisis y síntesis: Encontrar, analizar, criticar (razonamiento crítico), relacionar, estructurar y sintetizar información proveniente de diversas fuentes, así como integrar ideas y conocimientos
- CT02 – Capacidad de organización y planificación así como capacidad de gestión de la información.
- CT03 – Capacidad para el uso y aplicación de las TIC en el ámbito académico y profesional.
- CT04 – Capacidad de comunicación en lengua extranjera, particularmente en inglés.
- CT05 – Capacidad de trabajo en equipo.
- CT06 – Capacidad para innovar y generar nuevas ideas
- CT08 – Capacidad para proyectar los conocimientos, habilidades y destrezas adquiridos para promover una sociedad basada en los valores de la libertad, la justicia, la igualdad y el pluralismo

RESULTADOS DE APRENDIZAJE (Objetivos)

Conocer y saber aplicar las arquitecturas más adecuadas en los que se fundamentan los sistemas software, los componentes que los integran, las interfaces que se definen entre ellos, los patrones de diseño que supervisan su composición y la restricciones a la hora de aplicarlos.

Saber planificar la evolución de los sistemas software y evaluar el grado de cumplimiento de los requisitos de calidad de servicio que se exigen a los sistemas software complejos en la actualidad.

Entender los fundamentos, herramientas y distribuciones libres disponibles del software de intermediación sobre el que se apoya la construcción de sistemas software basado en componentes distribuidos.

Comprender los aspectos más destacados y características del enfoque orientado a servicios y las arquitecturas de servicios para el desarrollo de software, y saber aplicar las metodologías y tecnologías más apropiadas en cada caso.

Diseñar y utilizar marcos de trabajo para la construcción de sistemas software en los siguientes dominios de aplicaciones: sistemas distribuidos, componentes para computación de altas prestaciones, sistemas empotrados y móviles, incluyendo una aplicación de sistemas ubicuos y otra de inteligencia ambiental

PROGRAMA DE CONTENIDOS TEÓRICOS Y PRÁCTICOS

TEÓRICO

1. Desarrollo de software basado en componentes y servicios.
 - 1.1. Formalización de los sistemas abiertos y basados en componentes.
 - 1.2. Técnicas de diseño y desarrollo basadas en componentización del software



2. Servicios Web

2.1. Limitaciones del software intermediario convencional

2.2. Middleware y arquitecturas de servicios

2.3. Servicios Web (SW) contemporáneos

2.4. Notaciones y lenguajes

3. Modelado de procesos de negocio

3.1. Desarrollo de software, basado en SW, para procesos de negocio

3.2. Composición de SW: orquestación y coreografía.

3.3. Notaciones de modelado actuales (BPEL)

4. Sistema ubicuos e inteligencia ambiental

4.1. Introducción a la Computación Ubicua

4.2. Marcos de trabajo actuales de desarrollo

4.3. Servicios colaborativos

4.4. Modelado ontológico con OWL

Seminarios:

- Especificación de componentes de software con UML/OCL
- Introducción al diseño e implementación de servicios web RESTful
- Desarrollo de clientes de servicios web en dispositivos móviles
- OWL y el modelado semántico de ontologías para la Web

PRÁCTICO

1. Programación de servicio web RESTful con EJB y ORM

2. Orquestación de servicios Web utilizando WS-BPEL

3. Desarrollo de cliente responsivo para dispositivos móviles para el servicio web RESTful

4. Desarrollar una ontología de un sistema ubicuo utilizando un marco de trabajo y editor de ontologías de fuentes abiertas

BIBLIOGRAFÍA

BIBLIOGRAFÍA FUNDAMENTAL



Manuel I. Capel. Desarrollo de Software y Sistemas Basados en Componentes y Servicios. Garceta grupo editorial, 2016 (Madrid).

Thomas Erl. SOA: Principles of Service Design. Prentice Hall, 2008 (recurso electrónico: <http://proquest.safaribooksonline.com/9780132344821>).

Frederick Hirsh, Jonh Kemp, Jani Ilka. Mobile Web Services: Architecture & Implementation. (2006). John Wiley and Sons, 2006.

Clements Szyperski. Component Software: Beyond Object-oriented programming. Pearson Education, 2002 (Londres)

Paulo Veríssimo y Luís Rodrigues. Distributed Systems for System Architects. Kluwer Academic, 2004.

BIBLIOGRAFÍA COMPLEMENTARIA

-Baldauf, M., Dustdar, S., and Rosenberg, F. A survey of context aware systems. International Journal of Adhoc and Ubiquitous Computing, 2(4):263–267 (2012).

-Branca, G. and Atzori, L. A survey of SOA technologies in NGN network architectures. IEEE Communications Surveys Tutorials, 14(3):644–661 (2012).

-Bell, M. SOA Modeling Patterns for Service Oriented Discovery Analysis. Wiley, 2010.

-Kessig, Z. Building Web Applications with Erlang: Working with REST and Web. O'Reilly, 2012.

-Marcs, E.D., Bell M. Service Oriented Architecture (SOA): A Planning and Implementation Guide for Business and Technology. Wiley, 2006.

-Margolis, B. SOA for the business developer : concepts, BPEL, and SCA. MC Press, 2007.

-Pashtan, A. Mobile Web Services. Cambridge University Press, 2005.

-Raz, D., Juhola, A., Serrat-Fernandez, J., and Galis, A. (2006). Fast and efficient context-aware services. John Wiley and Sons, 2006.

-Stavropoulos, T., Vrakas, D., and Vlahavas, I. A survey of service composition in ambient intelligence environments. Artificial Intelligence Review, 40(3):247–270 (2013).

-Taylor, H. Event-driven Architecture: How SOA Enables the Real-time Enterprise. Addison-Wesley, 2009

ENLACES RECOMENDADOS

A principio de curso se avisará de la plataforma Web y páginas auxiliares donde se encontrarán los enlaces recomendados para la asignatura.

METODOLOGÍA DOCENTE



- MD01 Clases Teóricas-Expositivas
- MD02 Resolución de Problemas
- MD03 Resolución de Casos Prácticos
- MD04 Aprendizaje basado en Proyectos
- MD05 Prácticas en Laboratorio
- MD06 Taller de Programación
- MD09 Demos
- MD10 Exposición de Trabajos Tutelados
- MD11 Conferencias
- MD16 Tutorías Académicas

EVALUACIÓN (instrumentos de evaluación, criterios de evaluación y porcentaje sobre la calificación final)

EVALUACIÓN ORDINARIA

La teoría y las prácticas se evalúan independientemente, representado cada parte un 50% de la calificación final del curso.

La evaluación de la parte teórica constará de, al menos 2, exámenes con cuestiones/respuestas alternativas realizables en horas lectivas.

Para la evaluación de las prácticas se requiere la entrega de trabajos prácticos en fechas predeterminadas y la defensa de la tercera práctica ante la clase. Las fechas de las exposiciones se fijarán a principio de curso.

Todo lo relativo a la evaluación se registrará por la Normativa de evaluación y calificación de los estudiantes vigente en la Universidad de Granada, que puede consultarse en: https://lsi.ugr.es/lsi/normativa_examenes

Para aprobar la asignatura es necesario obtener una calificación numérica superior o igual a 5,0 (sobre 10,0). No obstante, además del requisito anterior, se establece adicionalmente que tanto la calificación correspondiente a la parte teórica como a la parte práctica sean mayores de 3,5 (sobre 10,0).

En el caso de evaluación continua, los seminarios se podrán evaluar teniendo en cuenta la asistencia, problemas resueltos, entrevistas y presentación de los trabajos.

Para los alumnos que se acojan a la evaluación única final, las pruebas y la evaluación estarán regidas por los criterios que se exponen más adelante.

EVALUACIÓN EXTRAORDINARIA

La evaluación extraordinaria se registrará por los mismos criterios y constará de las mismas pruebas que la de la evaluación única final.

EVALUACIÓN ÚNICA FINAL

Consistirá en dos pruebas de evaluación, con la descripción siguiente de cada una:



Evaluación de la teoría: los estudiantes realizarán una única prueba escrita que constará de preguntas de teoría: tipo test (35%), resolución de ejercicios (50%), supuesto práctico (15%)

Evaluación de las prácticas: los estudiantes realizarán una prueba en el laboratorio, que consistirá en la programación de un supuesto práctico similar a los realizados durante las prácticas regladas de la asignatura.

La ponderación de cada parte será del 50% en la calificación final y para aprobar la asignatura se han de cumplir los siguientes requisitos:

-La calificación de la prueba de teoría ha de ser igual o superior al 40% del máximo que se pueda obtener en dicha prueba

-La calificación de la prueba de prácticas ha de ser igual o superior al 40% del máximo que se pueda obtener en dicha prueba

INFORMACIÓN ADICIONAL

Para ser evaluado en la modalidad "continua" es necesario realizar todos los exámenes y entregar las prácticas en la fecha que se establezca para cada uno de estos controles.

Información de interés para estudiantado con discapacidad y/o Necesidades Específicas de Apoyo Educativo (NEAE): [Gestión de servicios y apoyos](https://ve.ugr.es/servicios/atencion-social/estudiantes-con-discapacidad) (<https://ve.ugr.es/servicios/atencion-social/estudiantes-con-discapacidad>).

