



1. Identificación

1.1. De la Asignatura

Curso Académico	2012/2013
Titulación	GRADO EN INGENIERIA INFORMÁTICA
Nombre de la Asignatura	SISTEMAS DISTRIBUIDOS
Código	3883
Curso	CUARTO
Carácter	OPTATIVA
Nº Grupos	1
Créditos ECTS	6
Estimación del volumen de trabajo del alumno	150
Organización Temporal/Temporalidad	Segundo Cuatrimestre
Idiomas en que se imparte	ESPAÑOL
Tipo de Enseñanza	Presencial

1.2. Del profesorado: Equipo Docente

Coordinador de la asignatura DIEGO SEVILLA RUIZ Grupo: 1	Área/Departamento	ARQUITECTURA Y TECNOLOGÍA DE COMPUTADORES/ INGENIERÍA Y TECNOLOGÍA DE COMPUTADORES
	Categoría	PROFESORES TITULARES DE ESCUELAS UNIVERSITARIAS
	Correo Electrónico / Página web / Tutoría electrónica	dsevilla@um.es http://ditec.um.es/personal/1 Tutoría Electrónica: Sí



	Teléfono, Horario y Lugar de atención al alumnado	Duración	Día	Horario	Lugar
		Anual	Martes	09:30- 11:00	868887571, Facultad de Informática B1.3.029
		Anual	Miércoles	09:30- 11:00	868887571, Facultad de Informática B1.3.029

2. Presentación

La asignatura pretende introducir al alumno a los problemas de diseño e implementación de aplicaciones distribuidas. Así, aborda los temas presentes en los textos clásicos de sistemas distribuidos, pero desde un punto de vista práctico, citando ejemplos de software real y actual. La asignatura se centra en los problemas derivados de la escalabilidad, eficiencia y distribución de aplicaciones, sobre todo en el entorno de Internet. Incluye el uso de patrones de diseño de aplicaciones distribuidas eligiendo ejemplos basados en sistemas RPC (CORBA), de mensajería, sistemas de Cloud Computing, etc. Finalmente, la asignatura introduce al alumno en patrones arquitecturales de diseño y desarrollo de aplicaciones distribuidas (usando un enfoque orientado a modelos) sobre RPC, Objetos Distribuidos, REST, sistemas de *middleware* de mensajería (MOM), sistemas *Peer-to-Peer*, sistemas *Map/Reduce* y bases de datos NoSQL.

3. Condiciones de acceso a la asignatura

3.1 Incompatibilidades

3.2 Recomendaciones

Sería interesante que el alumno contara con conocimientos de lenguajes de programación Orientados a Objetos como Java y C++, así como de técnicas de diseño basadas en Patrones, Modelado y Orientación a Objetos. También es importante que el alumno domine la programación de aplicaciones en entornos UNIX.



4. Competencias

4.1 Competencias Transversales

- Ser capaz de expresarse correctamente en español en su ámbito disciplinar. [Transversal1]
- Comprender y expresarse en un idioma extranjero en su ámbito disciplinar, particularmente el inglés. [Transversal2]
- Ser capaz de gestionar la información y el conocimiento en su ámbito disciplinar, incluyendo saber utilizar como usuario las herramientas básicas en TIC. [Transversal3]
- Considerar la ética y la integridad intelectual como valores esenciales de la práctica profesional. [Transversal4]
- Ser capaz de 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. [Transversal5]
- Ser capaz de trabajar en equipo y para relacionarse con otras personas del mismo o distinto ámbito profesional. [Transversal6]
- Desarrollar habilidades de iniciación a la investigación. [Transversal7]

4.2 Competencias de la asignatura y su relación con las competencias de la titulación

Competencia 1. Conocer las diversas posibilidades que existen a la hora de programar un Sistema Distribuido

Esta competencia de asignatura no se relaciona con ninguna competencia de titulación.

Competencia 2. Comprender el funcionamiento de distintas técnicas avanzadas para la construcción eficiente de aplicaciones que se ejecutan en red: patrones de acceso a objetos remotos, concurrencia y tolerancia a fallos

Esta competencia de asignatura no se relaciona con ninguna competencia de titulación.

Competencia 3. Aprender a identificar los cuellos de botella en el funcionamiento de las aplicaciones que se ejecutan en varios ordenadores simultáneamente

Esta competencia de asignatura no se relaciona con ninguna competencia de titulación.

Competencia 4. Aprender cómo el diseño de aplicaciones distribuidas difiere del diseño de aplicaciones convencionales

Esta competencia de asignatura no se relaciona con ninguna competencia de titulación.

Competencia 5. Ser capaz de aplicar de forma práctica las tendencias actuales de desarrollo de aplicaciones distribuidas

Esta competencia de asignatura no se relaciona con ninguna competencia de titulación.

5. Contenidos

TEMA 1 Introducción a los Sistemas Distribuidos

Sistemas Distribuidos: Introducción y Conceptos // Sistemas Distribuidos versus Sistemas Centralizados // Sistemas Operativos Distribuidos // Middleware: Tipos // Sistemas Grid y Cloud // Modelos Distribuidos de Componentes // Introducción a los objetivos, patrones de diseño y herramientas conceptuales de la asignatura

TEMA 2 Paradigmas de programación de objetos distribuidos

IPC y Sistemas RPC: Sistemas de Objetos Distribuidos // Introducción a CORBA // El lenguaje IDL // El mapping de IDL a C // El mapping de IDL a Java // Sistemas de Información en Sistemas distribuidos // Servicios de Nombres // Servicios de Trading // Patrones: Iterator, Federation // CORBA Naming Service

TEMA 3 Patrones para mejorar la eficiencia y escalabilidad de aplicaciones distribuidas



Aspectos específicos de diseño para aplicaciones distribuidas // Eficiencia y escalabilidad en sistemas distribuidos // Patrones para eficiencia y escalabilidad // Diseño de interfaces remotas (fat operations, coarse object models, wrapper) // Escalabilidad en el número de clientes // Manejo de threads de ejecución

TEMA 4 Nuevos paradigmas de desarrollo en el Web

Message-Oriented-Middleware (MOM) // Espacios de tuplas // Servicios Web // Modelos de Componentes Distribuidos // Tecnologías REST y SOA // Map-Reduce // Almacenes de datos NoSQL

PRÁCTICAS

Práctica 1 Introducción a Eclipse y al modelado de aplicaciones distribuidas :Global

Eclipse // EMF // Metamodelado // Aplicaciones distribuidas con Eclipse

Práctica 2 Introducción a Erlang :Relacionada con los contenidos Tema 4 y Tema 3

Entorno de ejecución y herramientas // Ejemplos de clientes y servidores // Escalabilidad // Gestión de clústeres

Práctica 3 Tecnologías REST, SOA y NoSQL :Relacionada con los contenidos Tema 4 y Tema 3

Introducción a REST de forma práctica // JSON y otros modelos de datos ligeros // Bases de datos NoSQL (CouchDB, MonogoDB, Memcache) // Map-Reduce

6. Metodología Docente

Actividad Formativa	Metodología	Horas Presenciales	Trabajo Autónomo	Volumen de trabajo
Presentación en el aula de los conceptos propios de la materia haciendo uso de metodología expositiva con lecciones magistrales participativas y medios audiovisuales.		28	28	56



Actividad Formativa	Metodología	Horas	Trabajo	Volumen
		Presenciales	Autónomo	de trabajo
Seminarios de prácticas		15	30	45
Tutorías de seguimiento de proyectos y prácticas		15	15	30
Realización de proyecto de investigación		0	17	17
Evaluación		2		2

7. Horario de la asignatura

<http://www.um.es/web/informatica/contenido/estudios/grados/informatica/2012-13#horarios>

8. Sistema de Evaluación

Competencia Evaluada	Métodos / Instrumentos	Realización de boletines de prácticas
	Criterios de Valoración	Realización y entrega de los boletines de prácticas que se irán rellenando en sus correspondientes sesiones presenciales en los laboratorios de prácticas, entregándose generalmente en la sesión siguiente de prácticas.
	Ponderación	50%
Competencia Evaluada	Métodos / Instrumentos	Trabajo de investigación con exposición
	Criterios de Valoración	Realización, entrega y exposición de un proyecto de investigación sobre una tecnología distribuida proporcionado por el profesor (se propondrán varios al principio de la asignatura).
	Ponderación	50%



Fechas de exámenes

<http://www.um.es/web/informatica/contenido/estudios/grados/informatica/2012-13#exámenes>

9. Bibliografía (básica y complementaria)

-  ANDREW S. TANENBAUM, "SISTEMAS OPERATIVOS DISTRIBUIDOS". Prentice-Hall Hispanoamericana, 1995. ISBN: 968-880-627-7.
-  BROSE, VOGEL, DUDDY. "JAVA PROGRAMMING WITH CORBA", 3 EDITION, John Wiley & Sons, 2001. ISBN: 0-471-60695-2.
-  SCHMIDT, STAL, ROHNERT, BUSCHMAN. "PATTERN-ORIENTED SOFTWARE ARCHITECTURE. VOLUME 2: PATTERNS FOR CONCURRENT AND NETWORKED OBJECTS". John Wiley & Sons, 2000. ISBN: 0-471-60695-2.
-  GEORGE COULOURIS, JEAN DOLLIMORE Y TIM KINDBERG, "SISTEMAS DISTRIBUIDOS. COCEPTOS Y DISEÑO". 3a EDICIÓN. Pearson Educación, 2001. ISBN: 84-7829-049-4.
-  HENNING, VINOSKI. "PROGRAMACIÓN EN CORBA CON C++". Pearson Educación, 2002, ISBN: 84-7829-048-6.
-  Dave Steinberg, Frank Budinsky, Marcelo Paternostro, Ed Merks. EMF, Eclipse Modeling Framework, Second Edition. Addison-Wesley, 2009.
-  Programming Erlang, Joe Armstrong, Pragmatic Bookshelf. ISBN: 1-9343560-0-X
-  RESTful Web Services, Richardson, Ruby, O'Reilly Media; First Edition edition (May 15, 2007), ISBN: 0596529260
-  SOA with REST: Principles, Patterns & Constraints for Building Enterprise Solutions with REST, Pautasso et al., ISBN: 0137012519, Prentice Hall

10. Observaciones y recomendaciones