



Ingeniería de Sistemas y Servicios Telemáticos

Guía de Aprendizaje – Información al estudiante

1. Datos Descriptivos

Asignatura	Ingeniería de Sistemas y Servicios Telemáticos
Materia	M10. Tecnología específica de Telemática
Departamento responsable	Ingeniería de Sistemas Telemáticos
Créditos ECTS	4,5
Carácter	Obligatoria itinerario
Titulación	Grado en Ingeniería de Tecnologías y Servicios de Telecomunicación
Curso	Cuarto
Especialidad	Itinerario en Telemática

Curso académico	2013-2014
Semestre en que se imparte	Segundo
Idioma en que se imparte	Castellano
Página Web	Sitio Moodle de la asignatura



2. Profesorado

NOMBRE Y APELLIDO	DESPACHO	Correo electrónico
José María del Álamo Ramiro	B-204.1	jmdela@dit.upm.es
Juan Carlos Dueñas López	C-210	jcduenas@dit.upm.es
Juan Carlos Yelmo García (Coordinador)	C-217	jcyelmo@dit.upm.es

3. Conocimientos previos requeridos para poder seguir con normalidad la asignatura

Asignaturas superadas	N/A
Otros resultados de aprendizaje necesarios	<ul style="list-style-type: none">• Fundamentos de Telemática• Programación• Análisis y Diseño de Software• Computación en red

4. Objetivos de Aprendizaje

COMPETENCIAS ASIGNADAS A LA ASIGNATURA Y SU NIVEL DE ADQUISICIÓN		
Código	Competencia	Nivel
CG1-13	Todas las asignaturas del Plan de Estudios contribuyen en mayor o menor medida a la consecución de las competencias generales del perfil de egreso.	1
CECT-1	Capacidad para aprender de manera autónoma nuevos conocimientos y técnicas adecuados para la concepción, el desarrollo, o la explotación de sistemas y servicios de telecomunicación	1
CECT-2	Capacidad de utilizar aplicaciones de comunicación e informáticas (ofimáticas, bases de datos, cálculo avanzado, gestión de proyectos, visualización, etc.) para apoyar el desarrollo y explotación de redes, servicios y aplicaciones de telecomunicación y electrónica.	1
CECT-7	Conocimiento y utilización de los fundamentos de la programación en redes, sistemas y servicios de telecomunicación.	2
CE-TL1	Capacidad de construir, explotar y gestionar las redes, servicios, procesos, y aplicaciones de telecomunicaciones, entendidas éstas como sistemas de captación, transporte, representación, procesado, almacenamiento, gestión y presentación de información multimedia, desde el punto de vista de los servicios telemáticos.	1
CE-TL6	Capacidad de diseñar arquitecturas de redes y servicios telemáticos	2
CE-TL7	Capacidad de programación de servicios y aplicaciones telemáticas, en red y distribuidas.	3

LEYENDA: Nivel de adquisición 1: Básico
Nivel de adquisición 2: Medio
Nivel de adquisición 3: Avanzado

RESULTADOS DE APRENDIZAJE DE LA ASIGNATURA			
Código	Resultado de aprendizaje	Competencias asociadas	Nivel de adquisición
RA1.1	Conocer las actividades básicas del proceso de desarrollo de sistemas y servicios software y los principales modelos de ciclo de vida	CG1-13, CECT-1, CECT-2, CE-TL1	3
RA1.2	Conocer y saber utilizar Scrum como modelo de proceso para la gestión y desarrollo en proyectos de software	CG1-13, CECT-1, CECT-2, CE-TL1	2
RA1.3	Conocer los principios básicos de gestión de configuración de sistemas software en evolución y conocer y saber utilizar las herramientas básicas de esta actividad	CG1-13, CECT-1, CECT-2, CE-TL1	2
RA2.1	Conocer y comprender el concepto de requisito de un sistema software y sus diferentes tipos	CG1-13, CECT-1, CECT-2, CE-TL1	2
RA2.2	Conocer, comprender y saber utilizar diferentes enfoques para la definición y análisis de requisitos del software	CG1-13, CECT-1, CECT-2, CE-TL1	2
RA2.3	Conocer los principios básicos de gestión de requisitos del software y conocer y saber utilizar las herramientas básicas de esta actividad	CG1-13, CECT-1, CECT-2, CE-TL1	2
RA3.1	Conocer y saber utilizar la notación UML para el modelado de sistemas software	CG1-13, CECT-1, CECT-2, CE-TL6	3
RA3.2	Conocer distintos estilos arquitectónicos de un sistema software, comprender sus diferencias y saber elegir el más adecuado de acuerdo a los requisitos del sistema	CG1-13, CECT-1, CE-TL6	3
RA3.3	Conocer elementos comunes para el diseño detallado de un sistema software	CG1-13, CECT-1, CE-TL6	2
RA4.1	Conocer el proceso y las actividades de transición, configuración, despliegue y operación	CG1-13, CECT-1, CE-TL1	2

RESULTADOS DE APRENDIZAJE DE LA ASIGNATURA			
Código	Resultado de aprendizaje	Competencias asociadas	Nivel de adquisición
RA4.2	Saber usar herramientas de automatización de construcción e integración continua	CG1-13, CECT-1, CECT-7, CE-TL7	2
RA4.3	Saber cómo aplicarlas sobre una plataforma de despliegue de componentes Java de servidor	CG1-13, CECT-1, CECT-7, CE-TL7	3
RA5.1	Conocer y comprender los conceptos fundamentales de la ingeniería de servicios	CG1-13, CECT-1, CECT-7, CE-TL7	2
RA5.2	Conocer y saber utilizar algunas de las tecnologías y herramientas para provisión de servicios telemáticos	CG1-13, CECT-1, CECT-7, CE-TL1 CE-TL7	2
RA6.1	Comprender los procesos de validación y verificación y sus relaciones con otras fases del ciclo de vida	CG1-13, CECT-1, CE-TL1	2
RA6.2	Conocer las principales técnicas y principios de pruebas	CG1-13, CECT-1, CE-TL1	2
RA6.3	Saber usar herramientas para pruebas en diferentes niveles: unitario, integración y sistema	CG1-13, CECT-1, CE-TL1	2

LEYENDA: Nivel de adquisición 1: Conocimiento
Nivel de adquisición 2: Comprensión y aplicación
Nivel de adquisición 3: Análisis y síntesis

5. Sistema de evaluación de la asignatura

INDICADORES DE LOGRO		
Ref	Indicador	Relacionado con RA
11.1	Reconocer el propósito y utilidad de cada una de las actividades básicas del proceso de desarrollo de software y organizar y relacionar estas actividades para conformar diferentes modelos de ciclo de vida	RA1.1
11.2	Reconocer los principios organizativos, roles, actividades, artefactos y elementos de control y revisión de Scrum y aplicar este modelo en proyectos de desarrollo de software	RA1.2
11.3	Reconocer los principios y actividades relacionados con la gestión de configuración de sistemas software en evolución y utilizar herramientas de soporte a dichas actividades en proyectos de desarrollo de software	RA1.3
12.1	Reconocer los diferentes tipos de requisitos de sistemas software	RA2.1
12.2	Reconocer los principales enfoques y principios organizativos para la definición y análisis de requisitos del software y aplicar dichos principios a la especificación de sistemas y servicios	RA2.2
12.3	Reconocer los principios y actividades relacionados con la gestión de requisitos del software y utilizar herramientas de soporte a dichas actividades en proyectos de desarrollo de software	RA2.3
13.1	Generar e interpretar diagramas UML para el modelado de aspectos estáticos y dinámicos de un sistema software	RA3.1
13.2	Modelar la arquitectura de alto nivel de un sistema software a partir de la especificación de requisitos	RA3.2
13.3	Conocer qué son y para qué se utilizan algunos de los patrones de diseño software orientado a objetos	RA3.3
14.1	Utilizar una herramienta de construcción automática	RA4.1, RA4.2
14.2	Saber empaquetar, desplegar, y arrancar/parar una aplicación web sobre una plataforma Java de servidor	RA4.2, RA4.3
15.1	Conocer el propósito y la utilidad de algunos de los enfoques para el desarrollo de sistemas software distribuidos	RA5.1
15.2	Aplicar los conceptos adquiridos para crear componentes para la provisión de servicios distribuidos	RA5.2



INDICADORES DE LOGRO		
Ref	Indicador	Relacionado con RA
16.1	Ser capaz de diseñar un conjunto de pruebas funcionales (caja blanca y negra) para un módulo	RA1, RA2
16.2	Utilizar una herramienta de gestión de pruebas	RA1, RA3
16.3	Utilizar una herramienta de ejecución de pruebas unitarias	RA2, RA3



EVALUACION SUMATIVA			
Breve descripción de las actividades evaluables	Momento	Lugar	Peso en la calif.
Primer examen parcial	Semana 7	Aula	25%
Primera presentación oral y entregables hasta la fecha de la presentación de acuerdo con el plan de proyecto	Semana 10	Aula	25%
Segundo examen parcial	Semana 15	Aula	25%
Segunda presentación oral y entregables hasta la fecha de la presentación de acuerdo con el plan de proyecto	Semana 16	Aula	25%
			Total: 100%



CRITERIOS DE CALIFICACIÓN

En convocatoria ordinaria, los alumnos serán evaluados mediante **evaluación continua**, según los criterios especificados más abajo. No obstante, en cumplimiento de la Normativa de Evaluación de la Universidad Politécnica de Madrid, los alumnos que lo deseen podrán ser evaluados mediante una única prueba final. El plazo para renunciar a la evaluación continua será de un mes desde el comienzo de la actividad docente de la asignatura y se comunicará por el alumno interesado mediante escrito presentado en el registro de la ETSI Telecomunicación y dirigido al Sr. Director del Departamento de Ingeniería de Sistemas Telemáticos. La presentación de este escrito implica la renuncia automática a la evaluación continua.

Las actividades de evaluación continua de la asignatura contemplan la valoración de actividades individuales y de trabajo en grupo de los alumnos.

Evaluación de actividades individuales: Dos exámenes escritos en horario de clase sobre los contenidos teóricos de la asignatura, el material de autoestudio y las prácticas individuales.

Primer examen: A mitad de semestre, en torno a la semana 7

Segundo examen: Al final del semestre, en torno a la semana 15

Evaluación de actividades de trabajo en grupo: Evaluación de entregables de proyecto (documentos, artefactos software, prototipos, demostraciones, etc.) a lo largo del semestre en función del plan de trabajo del proyecto y dos presentaciones orales realizadas por cada equipo de proyecto.

Primera presentación: A mitad del semestre, en torno a la semana 10.

Segunda presentación: Al final del semestre, en torno a la semana 16

La calificación final del alumno se obtendrá como media aritmética de la evaluación individual y la evaluación de grupo con el mismo peso. Se aprueba la asignatura con una nota igual o mayor a 5 sobre 10, siempre que en la evaluación de cada examen individual y presentación de proyecto se haya obtenido una calificación igual o mayor a 4.

Convocatoria ordinaria. Evaluación mediante una única prueba final

Este examen sólo lo realizan los alumnos que renuncien a la evaluación continua.

La prueba final consiste en un examen escrito individual que se califica sobre 10 puntos.

La fecha de este examen se determina en el calendario fijado por Jefatura de Estudios.

Convocatoria extraordinaria

La prueba final extraordinaria consiste en un examen escrito individual que se califica sobre 10 puntos. La fecha de este examen se determina en el calendario fijado por Jefatura de Estudios.

6. Contenidos y Actividades de Aprendizaje

CONTENIDOS ESPECÍFICOS		
Bloque / Tema / Capítulo	Apartado	Indicadores Relacionados
Tema 1: Soporte al proceso de desarrollo	1.1 El ciclo de vida de desarrollo de software	I1.1
	1.2 Metodologías ágiles: Scrum	I1.2
	1.3 Gestión de configuración del software	I1.3
Tema 2: Ingeniería de requisitos	2.1 Requisitos del software	I2.1
	2.2 Métodos y buenas prácticas	I2.2
	2.3 Gestión de requisitos	I2.3
Tema 3: Modelado y diseño de sistemas software	3.1 Lenguaje Unificado de Modelado (UML)	I3.1
	3.2 Diseño de arquitectura software	I3.2
	3.3 Diseño detallado	I3.3
Tema 4: Plataformas y entornos de despliegue	4.1 Procesos de transición	I4.1
	4.2 Operación y mantenimiento de servicios	I4.1, I4.2
	4.3 Plataforma Java de servidor (JEE)	I4.1, I4.2
Tema 5: Ingeniería de servicios	5.1 Arquitecturas de sistemas distribuidos	I5.1
	5.2 Tecnologías para el desarrollo de servicios	I5.2
Tema 6: Prueba de sistemas y servicios	6.1 Procesos y principios de verificación y validación	I6.1
	6.2 Automatización de pruebas	I6.2, I6.3
	6.3 Técnicas de prueba	I6.2



7. Breve descripción de las modalidades organizativas utilizadas y de los métodos de enseñanza empleados

CLASES DE TEORÍA	Se utilizará la lección magistral, con apoyo de recursos audiovisuales, para la exposición de contenidos, explicación de conocimientos, demostraciones, presentación de casos prácticos y aplicaciones, etc.
CLASES DE PROBLEMAS	Se resolverán en clase ejercicios que servirán para aplicar los conocimientos adquiridos en las clases de teoría.
PRÁCTICAS	Se implementarán en el laboratorio una serie de ejercicios prácticos que cumplirán unas determinadas especificaciones.
TRABAJOS AUTÓNOMOS	El alumno preparará temas en auto-estudio y realizará ejercicios para practicar y afianzar los conocimientos aprendidos.
TRABAJOS EN GRUPO	Los alumnos forman equipos de trabajo con división de perfiles y responsabilidad bajo la supervisión de un profesor para el desarrollo de prototipos de sistemas y servicios
TUTORÍAS	El alumno podrá hacer uso de tutorías personalizadas dentro de los horarios establecidos.

8. Recursos didácticos

BIBLIOGRAFÍA

Apuntes propios de la asignatura

Tema 1

- K.S. Rubin. Essential Scrum: A Practical Guide to the Most Popular Agile Process. Addison-Wesley, 2012. Disponible en safaribooksonline.
- R. Aiello, L. Sachs. Configuration Management Best Practices: Practical Methods that Work in the Real World. Addison-Wesley Professional; 2010. Disponible en safaribooksonline.

Tema 2

- K.E. Wiegers, J. Beatty. Software Requirements. Microsoft Press, 3rd edition, 2013.
- M. Cohn. User Stories Applied: For Agile Software Development. Addison-Wesley, 2004. Disponible en safaribooksonline.

Tema 3

- G. Booch, J. Rumbaugh, I. Jacobson: *El lenguaje unificado de modelado*, 2ª Edición, Pearson Educación S.A., 2006.
- R. Miles, K. Hamilton: *Learning UML 2.0*, O'Really Media, 2006. Disponible en safaribooksonline
- L. Bass, P. Clements, R. Kazman: *Software architecture in practice*, 3rd Edition, Addison-Wesley Professional (2012). Disponible en safaribooksonline.
- E. Freeman, E. Freeman, K. Sierra, B. Bates: *Head first design patterns*, O'Reilly Media (2004). Disponible en safaribooksonline.

Tema 4

- J. Humble, D. Farley. Continuous Delivery: Reliable Software Releases through Build, Test and Deployment Automation, 2010. Addison-Wesley Professional. Disponible en safaribooksonline.
- T.M. O'Brien, J. Casey, B. Fox. Maven: the complete reference. Sonatype Inc. 2011.
- Goncalves. Beginning Java EE 7, 2013. Disponible en Safaribooksonline.
- G. Zambon. Beginning JSP, JSF and Tomcat: Java Web Development, 2012. Disponible en safaribooksonline.

Tema 5

- B. A. Christudas, M. Barai, V. Caselli. Service Oriented Architecture with Java. 2011. Disponible en Safaribooksonline.
- J. Webber; S. Parastatidis, I. Robinson. REST in Practice. 2010. Disponible en safaribooksonline.

Tema 6

- J. D. McCaffrey. Software Testing: Fundamental Principles and Essential Knowledge. BookSurge Publishing, April 2009.
- R. D. Drabick. Best Practices for Formal Software Testing process: A menu of testing tasks, 2013
- L. Crispin, J. Gregory. Agile Testing: A Practical Guide for Testers and Agile Teams, Addison-Wesley, 2008.
- K. Beck. Test Driven Development: By Example. Addison-Wesley Professional, Nov. 2002.



RECURSOS WEB	Página web de la asignatura: http://www.dit.upm.es/asignaturas/grado-tel/isst.html
	Sitio Moodle de la asignatura ISST en el servidor moodle del departamento: http://moodle.dit.upm.es/
EQUIPAMIENTO	Laboratorio
	Aula
	Sala de trabajo en grupo

9. Cronograma de trabajo de la asignatura

Semana	Actividades en Aula	Actividades en Laboratorio	Trabajo Individual	Trabajo en Grupo	Actividades de Evaluación	Otros
Semana 1 (7 horas)	Tema 1.1 (2 h) Tema 1.2 (1 h)		Autoestudio (2h) Práctica individual (2 h)			
Semana 2 (8 horas)	Casos de estudio (1 h)	Tema 1.3 (2 h)	Autoestudio (2 h) Práctica individual (2 h)	Proyecto (1 h)		
Semana 3 (8 horas)	Tema 2.1 (1 h) Tema 2.2 (2 h)		Autoestudio (2h) Práctica individual (1 h)	Proyecto (2 h)		
Semana 4 (8 horas)	Casos de estudio (1 h)	Tema 2.3 (2 h)	Autoestudio (2h) Práctica individual (1 h)	Proyecto (2 h)		
Semana 5 (8 horas)	Tema 3.1 (2 h) Tema 3.2 (1 h)		Práctica individual (2 h)	Proyecto (3 h)		
Semana	Actividades en Aula	Actividades en Laboratorio	Trabajo Individual	Trabajo en Grupo	Actividades de Evaluación	Otros
Semana 6 (8 horas)	Caso de estudio (1 h)	Tema 3.3 (2 h)	Autoestudio (2 h)	Proyecto (3 h)		
Semana 7 (6 horas)	Caso de estudio (1 h)			Proyecto (3 h)	Examen parcial (2 h)	



Semana 8 (8 horas)	Tema 4.1 (1 h) Tema 4.2 (2 h)		Práctica individual (2 h)	Proyecto (3 h)		
Semana 9 (8 horas)	Caso de estudio (1 h)	Tema 4.3 (2 h)	Práctica individual (2 h)	Proyecto (3 h)		
Semana 10 (6 horas)	Caso de estudio (1 h)			Proyecto (3 h)	Presentación de proyecto (2 h)	
Semana 11 (8 horas)	Tema 5.1 (2 h) Tema 5.2 (1 h)		Autoestudio (1 h) Práctica individual (1 h)	Proyecto (3 h)		
Semana 12 (8 horas)	Caso de estudio (1 h)	Tema 5.3 (2 h)	Autoestudio (1 h) Práctica individual (1 h)	Proyecto (3 h)		
Semana	Actividades en Aula	Actividades en Laboratorio	Trabajo Individual	Trabajo en Grupo	Actividades de Evaluación	Otros
Semana 13 (8 horas)	Tema 6.1 (2 h) Tema 6.2 (1 h)		Autoestudio (1 h) Práctica individual (1 h)	Proyecto (3 h)		
Semana 14 (8 horas)	Caso de estudio (1 h)	Tema 6.3 (2 h)	Práctica individual (2 h)	Proyecto (3 h)		
Semana 15 (6 horas)	Caso de estudio (1 h)			Proyecto (3 h)	Examen parcial (2 h)	
Semana 16 (6 horas)	Caso de estudio (1 h)			Proyecto (3 h)	Presentación de proyecto (2 h)	

Nota: Para cada actividad se especifica la dedicación en horas que implica para el alumno



POLITÉCNICA

