

Proyecto/Guía docente de la asignatura

Se debe indicar de forma fiel cómo va a ser desarrollada la docencia. Esta guía debe ser elaborada teniendo en cuenta a todos los profesores de la asignatura. Conocidos los espacios y profesorado disponible, se debe buscar la máxima presencialidad posible del estudiante siempre respetando las capacidades de los espacios asignados por el centro y justificando cualquier adaptación que se realice respecto a la memoria de verificación. Si la docencia de alguna asignatura fuese en parte online, deben respetarse los horarios tanto de clase como de tutorías). La planificación académica podrá sufrir modificaciones de acuerdo con la actualización de las condiciones sanitarias.

Asignatura	FUNDAMENTOS DE SISTEMAS EN TIEMPO REAL		
Materia	INGENIERÍA DE REDES, SISTEMAS Y SERVICIOS TELEMÁTICOS		
Módulo	MATERIAS ESPECÍFICAS DE LA MENCIÓN EN SISTEMAS ELECTRÓNICOS		
Titulación	GRADO EN INGENIERÍA DE TECNOLOGÍAS ESPECÍFICAS DE TELECOMUNICACIÓN		
Plan	512	Código	46654
Periodo de impartición	2º CUATRIMESTRE	Tipo/Carácter	OPTATIVA DE LA MENCIÓN
Nivel/Ciclo	GRADO	Curso	4º
Créditos ECTS	6		
Lengua en que se imparte	CASTELLANO		
Profesor/es responsable/s	MANUEL RODRÍGUEZ CAYETANO		
Datos de contacto (E-mail, teléfono...)	TELÉFONO: 983 423000 ext. 5541 E-MAIL: manuel.rodriguez@tel.uva.es		
Departamento	TEORÍA DE LA SEÑAL Y COMUNICACIONES E INGENIERÍA TELEMÁTICA		

1. Situación / Sentido de la Asignatura

1.1 Contextualización

A medida que los ordenadores se han ido haciendo más pequeños, rápidos, fiables y baratos, su campo de aplicación se ha ido ampliando a un gran número de nuevas áreas. Aunque inicialmente los sistemas informáticos se desarrollaron para resolver problemas de cálculo (sin restricciones temporales), su presencia en la actualidad se ha extendido a todo tipo de equipos, desde lavadoras controladas por microprocesador hasta sistemas de control de tráfico aéreo.

Un área en la que este uso ha aumentado considerablemente es aquella en que los sistemas no tienen como objetivo fundamental procesar información, aunque sí necesitan este procesamiento de información para desarrollar su función principal. Un ejemplo típico de este tipo de sistemas es el sistema de control de frenado ABS usado en automóviles. Aunque su cometido principal es garantizar un frenado seguro, sin bloqueo de ninguna rueda, necesita procesar información (la velocidad de giro de cada rueda) para calcular la presión de frenado que tiene que aplicar al elemento de frenado de cada una de ellas. Este tipo de sistemas se conocen como *sistemas de tiempo real*, e imponen nuevos requisitos tanto a los sistemas informáticos que forman parte de ellos como a las aplicaciones que se ejecutan en estos últimos. Básicamente, estos sistemas tienen como característica más importante que la corrección de su funcionamiento no sólo depende de la corrección de los resultados de los cálculos realizados sino también del instante temporal en el que se producen.

Además, estos sistemas suelen ser sistemas “críticos en cuanto a seguridad”. Es decir, garantizar su seguridad es un requisito esencial, puesto que un funcionamiento incorrecto del sistema puede tener consecuencias graves (incluso la pérdida de vidas humanas). En este ámbito, el término seguridad abarca dos aspectos:

- (1) el sistema debe ser seguro para el entorno, es decir, debe tener un funcionamiento correcto para no producir daños en el entorno (“safety-critical system”)
- (2) el sistema debe ser seguro frente a entradas erróneas o maliciosas provenientes de su entorno, para que la integridad del sistema (básicamente, su correcto funcionamiento) no se vea comprometida (“security-critical system”).

Entre los dominios de aplicación en los que los sistemas son críticos en cuanto a la seguridad tenemos la aviónica (sistemas electrónicos en aviones) tanto comercial como militar, sistemas de control de tráfico aéreo, sistemas de control de ferrocarriles, sistemas de seguridad en automoción, tanto activa (control de frenado ABS, control de estabilidad, etc.) como pasiva (airbags, por ejemplo), o dispositivos médicos.

Dentro de los sistemas de tiempo real existen sistemas con características especiales, como los sistemas empotrados (“embedded”). Básicamente éstos se caracterizan por realizar una función dedicada dentro de un sistema eléctrico o mecánico mayor (en el que se encuentran integrados), y presentan algunos requisitos extra comparados con otros sistemas de propósito general, como son un bajo consumo eléctrico, ser capaces de funcionar en un entorno con condiciones severas y tener un bajo precio por unidad. Para lograr estos requisitos los recursos de procesamiento (básicamente capacidad de la CPU, memoria RAM y espacio de almacenamiento secundario) son más limitados que en otros sistemas, lo que impone condiciones especiales para su programación (por ejemplo, sistemas operativos más limitados -sin memoria virtual- o inexistentes, menos memoria RAM disponible para los programas, etc.). Por lo general estos sistemas están basados en el uso de microcontroladores (CPUs con memoria e interfaces a periféricos integrados en un único chip). Los sistemas empotrados están recibiendo recientemente un creciente interés dada su aplicación a la Internet de

las cosas (IoT), que se define como una red de dispositivos físicos que incluyen sensores y software para proporcionar conectividad en red, y que permite a estos dispositivos recoger e intercambiar datos de todo tipo.

La presente asignatura abordará el estudio de las características principales de los sistemas en tiempo real, especialmente las relacionadas con la planificación de tareas con requisitos temporales. Se estudiarán, además, los requisitos necesarios para los lenguajes de programación que deberán usarse en estos sistemas y se abordarán los aspectos fundamentales del desarrollo de aplicaciones para sistemas en tiempo real.

Una parte importante de la asignatura estará dedicada al desarrollo una aplicación para un sistema en tiempo real. Para este desarrollo se utilizará el lenguaje de programación Ada, diseñado desde sus inicios para sistemas críticos en cuanto a la seguridad. Debido a sus características, Ada permite detectar durante la compilación muchos errores que en otros lenguajes sólo son detectables durante la fase de depuración de la ejecución del programa, reduciendo en un factor importante el coste de desarrollo. Además, permite una fácil interoperabilidad con otros lenguajes muy extendidos, como C, C++ o java, y desarrollar software para todo tipo de sistemas informáticos, desde ordenadores de propósito general hasta sistemas empotrados sin sistema operativo. Además, el subconjunto del lenguaje Ada denominado SPARK permite analizar formalmente la corrección de los programas desarrollados, reduciendo la complejidad y duración de la fase de pruebas de los mismos.

Ada es un lenguaje estandarizado por ISO y con amplio uso en el dominio de los sistemas de tiempo real. Entre las compañías y proyectos que lo utilizan en sus desarrollos podríamos citar a EuroControl (sistema de control de tráfico aéreo), Airbus o Boeing (aviación comercial), Inmarsat (comunicación vía satélite) o TGV (sistema de control de los ferrocarriles de alta velocidad franceses).

Las prácticas también incluirán la programación básica de placas Arduino, escogidas como sistema empotrado de bajo coste para la conexión con sensores y actuadores que interactúan con el entorno.

1.2 Relación con otras materias

Esta asignatura se apoya en la asignatura "Programación" de la materia "Informática" del "Bloque de Materias Instrumentales" que se imparte en el primer cuatrimestre del primer curso, en la asignatura "Fundamentos de Ordenadores y Sistemas Operativos", de la materia de "Fundamentos de Ordenadores" del "Bloque de Materias Básicas de Telecomunicaciones", que se imparte en segundo cuatrimestre del primer curso y en la asignatura de "Ingeniería de Sistemas Software" de la materia "Fundamentos de Sistemas Software" del "Bloque de Materias Básicas de Telecomunicaciones" que se imparte en el segundo cuatrimestre del segundo curso. En dichas asignaturas se proporcionarán los conceptos básicos de programación, desarrollo de sistemas software y sistemas operativos, que facilitarán al alumno la comprensión de otros mostrados en la presente asignatura.

1.3 Prerrequisitos

No existen condiciones previas excluyentes para cursar esta asignatura, aunque sí recomendaciones lógicas

que el alumno debería tener en cuenta. En concreto, es recomendable haber cursado con anterioridad las asignaturas de “Programación”, “Fundamentos de Ordenadores y Sistemas Operativos”, e “Ingeniería de Sistemas Software”.

2. Competencias

2.1 Generales

- GBE3. Capacidad para resolver problemas con iniciativa, creatividad y razonamiento crítico.
- GBE4. Capacidad para diseñar y llevar a cabo experimentos, así como analizar e interpretar datos.
- GC1. Capacidad de organización, planificación y gestión del tiempo.
- GC2. Capacidad para comunicar, tanto por escrito como de forma oral, conocimientos, procedimientos, resultados e ideas relacionadas con las telecomunicaciones y la electrónica.
- GC3. Trabajar en cualquier contexto, individual o en grupo, de aprendizaje o profesional, local o internacional, desde el respeto a los derechos fundamentales, de igualdad de sexo, raza o religión y los principios de accesibilidad universal, así como la cultura de paz.

2.2 Específicas

- T7. Conocimiento y utilización de los fundamentos de la programación en redes, sistemas y servicios de telecomunicación.

3. Objetivos

Al finalizar la asignatura el alumno deberá ser capaz de:

1. Comprender las principales características distintivas de un sistema en tiempo real
2. Comprender y analizar el funcionamiento de distintos algoritmos de planificación de procesos específicos de los sistemas en tiempo real
3. Diseñar, desarrollar y desplegar aplicaciones para sistemas en tiempo real

4. Contenidos y/o bloques temáticos

Bloque 1: Sistemas en tiempo real

Carga de trabajo en créditos ECTS: 6

a. Contextualización y justificación

Véase el apartado 1.1.

b. Objetivos de aprendizaje

Véase el apartado 3.

c. Contenidos

Tema 1: Introducción a los sistemas en tiempo real

1. Definición y estructura típica
2. Requisitos principales
3. Tipos de sistemas en tiempo real

Tema 2: Introducción al lenguaje de programación Ada

1. Conceptos básicos
1. Conceptos avanzados

Tema 3: Desarrollo de aplicaciones para sistemas en tiempo real

1. Concurrencia, sincronización y comunicación entre procesos
2. Implementación de restricciones temporales
3. Interconexión con sistemas reales de sensores y actuadores: sistemas Arduino

Tema 4: Planificación de procesos en tiempo real

1. Requisitos para la planificación de procesos en tiempo real
2. Algoritmos de planificación de procesos en tiempo real

d. Métodos docentes

- Clase magistral participativa
- Taller de prácticas guiadas en el laboratorio

e. Plan de trabajo

Véase el Anexo I

f. Evaluación

La evaluación de la adquisición de competencias se basará en:

- Prueba escrita al final del cuatrimestre
- Funcionamiento y estructura de los programas desarrollados en el laboratorio por el alumno, así como la documentación asociada a dichos programas
- Respuestas a las preguntas planteadas en los enunciados de prácticas

g Material docente

Es fundamental que las referencias suministradas este curso estén actualizadas y sean completas. Los profesores tienen acceso, a la plataforma Leganto de la Biblioteca para actualizar su bibliografía recomendada ("Listas de Lecturas"). Si ya lo han hecho, pueden poner tanto en la guía docente como en el Campus Virtual el enlace permanente a Leganto.

g.1 Bibliografía básica

- Alan Burns, and Andy Wellings. *Real-time Systems and Programming Languages*. Addison-Wesley, 2009.
- John Barnes. *Programming in Ada 2005*. Addison-Wesley, 2006.

g.2 Bibliografía complementaria

- Jan Skansholm. *Ada 95: From the Beginning*. Addison-Wesley, 1996
- Ada 95 Lovelace tutorial. 1997. <http://www.adahome.com/Tutorials/Lovelace/lovelace.htm>
- Laurent Pautet. *Ada 95 tutorial*. 1998. <http://www.infres.enst.fr/~pautet/Ada95/a95list.htm>
- *Ada programming wikibook*. 2015. http://en.wikibooks.org/wiki/Ada_Programming
- Ada Resource Association. 2015. <http://www.adaic.org/Arduino> - Learn the basics, 2015. <http://arduino.cc/en/Tutorial/HomePage>

g.3 Otros recursos telemáticos (píldoras de conocimiento, blogs, videos, revistas digitales, cursos masivos (MOOC), ...)

h. Recursos necesarios

- Documentación de apoyo
- Entorno de trabajo en la plataforma Moodle ubicada en el Campus Virtual de la Universidad de Valladolid u otra plataforma virtual alternativa.
- Laboratorio de prácticas, con al menos un ordenador para cada dos alumnos, para las sesiones de laboratorio. Cada ordenador contará con un entorno de desarrollo en el lenguaje Ada y un kit Arduino.

i. Temporalización

CARGA ECTS	PERIODO PREVISTO DE DESARROLLO
6	Semanas 1 a 15

5. Métodos docentes y principios metodológicos

- Clase magistral participativa
- Taller de prácticas guiadas en el laboratorio

6. Tabla de dedicación del estudiante a la asignatura

ACTIVIDADES PRESENCIALES o PRESENCIALES A DISTANCIA ⁽¹⁾	HORAS	ACTIVIDADES NO PRESENCIALES	HORAS
Clases teórico-prácticas (T/M)	25	Estudio y trabajo autónomo individual	45
Seminarios (S)	5	Estudio y trabajo autónomo grupal	45
Laboratorios (L)	30		
Total presencial	60	Total no presencial	90
TOTAL presencial + no presencial			150

(3) Actividad presencial a distancia es cuando un grupo sigue una videoconferencia de forma síncrona a la clase impartida por el profesor.

7. Sistema y características de la evaluación

INSTRUMENTO/PROCEDIMIENTO	PESO EN LA NOTA FINAL	OBSERVACIONES
Informes de prácticas de laboratorio y revisión del funcionamiento de los programas desarrollados (la revisión del funcionamiento de cada práctica será presencial, el envío de la documentación evaluable se realizará a través del campus virtual)	50%	Es condición necesaria (pero no suficiente) para superar la asignatura alcanzar una calificación igual o superior a 5 sobre 10 puntos en la nota total de laboratorio (informes+funcionamiento de los programas).
Examen final escrito	50%	Es condición necesaria (pero no suficiente) para superar la asignatura alcanzar una calificación igual o superior a 5 sobre 10 puntos en este examen.

CRITERIOS DE CALIFICACIÓN

- **Convocatoria ordinaria:**
 - o Si un alumno no alcanza los requisitos mínimos descritos en la tabla anterior, su calificación final en la asignatura será el mínimo entre el valor calculado según la ponderación descrita en la tabla y 4,5 puntos sobre 10. Por ejemplo, si obtiene un 4 sobre 10 en la nota total de laboratorio (no supera los requisitos mínimos) y un 6 sobre 10 en el examen de teoría (que supondría una nota final de 5 sobre 10 según la ponderación dada en la tabla si no se tuviesen en cuenta los requisitos), la nota final de la asignatura será de 4,5 sobre 10 puntos. En caso de no cumplir los requisitos en ninguna de ambas partes, la nota final de la asignatura será la media de las dos notas.
- **Convocatoria extraordinaria:**
 - o Se mantiene (dentro de un mismo curso académico) la calificación obtenida en cada instrumento de la tabla siempre que dicha calificación sea igual o superior a 5 sobre 10 puntos
 - o El alumno deberá realizar de nuevo el examen final escrito si la nota obtenida en él para convocatorias anteriores del mismo curso académico es inferior a 5 sobre 10 puntos
 - o El alumno deberá realizar de nuevo la práctica o prácticas suspensas (puntuación inferior a 5 sobre 10 puntos), siguiendo los enunciados planteados para la convocatoria extraordinaria, si la nota total de prácticas obtenida en convocatorias anteriores del mismo curso académico es inferior a 5 sobre 10 puntos

8. Consideraciones finales

El Anexo I mencionado en la guía, donde se describe la planificación detallada, se entregará al comienzo de la asignatura.

