



FICHA TÉCNICA DE LA ASIGNATURA

Datos de la asignatura	
Nombre completo	Desarrollo de aplicaciones y servicios
Código	DTC-IMAT-322
Título	Grado en Ingeniería Matemática e Inteligencia Artificial
Impartido en	Grado en Ingeniería Matemática e Inteligencia Artificial [Tercer Curso]
Créditos	6,0 ECTS
Carácter	Obligatoria (Grado)
Departamento / Área	Departamento de Telemática y Computación
Responsable	Álvaro Ruiz Calzada, Cristian Fernández del Pozo & David Tortosa Escudero
Horario	Grupo A: Lunes y miércoles de 16h a 17h50. Grupo B: Lunes y martes de 18h15 a 20:05
Horario de tutorías	Via Teams a disponibilidad del estudiante y del profesor

Datos del profesorado	
Profesor	
Nombre	Álvaro Ruiz Calzada
Departamento / Área	Departamento de Telemática y Computación
Correo electrónico	arcalzada@icai.comillas.edu
Profesor	
Nombre	Cristian Fernández del Pozo
Departamento / Área	Departamento de Telemática y Computación
Correo electrónico	cfdelpozo@icai.comillas.edu
Profesores de laboratorio	
Profesor	
Nombre	David Tortosa Escudero
Departamento / Área	Departamento de Telemática y Computación
Correo electrónico	dtortosa@icai.comillas.edu

DATOS ESPECÍFICOS DE LA ASIGNATURA

Contextualización de la asignatura
Aportación al perfil profesional de la titulación
En esta asignatura se refuerza el itinerario de Software iniciado en Programación de 1º y se centra en el desarrollo de aplicaciones Web.
Prerrequisitos
Programación en python.



Competencias - Objetivos

Competencias

GENERALES

CG04	Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.
CG05	Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería
CG07	Capacidad para integrarse en equipos de trabajo y colaborar de forma activa con otras personas, áreas y organizaciones en la consecución de los objetivos ligados a las actividades de extracción de valor de los datos e inteligencia artificial.

ESPECÍFICAS

CE16	Capacidad para diseñar e implementar aplicaciones web que permitan publicar e intercambiar los resultados obtenidos por los modelos analíticos realizados mediante varias interfaces y servicios de comunicación
CE18	Conocimiento de tecnologías habilitadoras de la transformación digital para el desarrollo de soluciones innovadoras en las organizaciones.
CE37	Capacidad para presentar y defender un proyecto en el ámbito de los conocimientos y tecnologías específicas de la titulación, de naturaleza profesional, en el que se sinteticen e integren otras competencias adquiridas en las enseñanzas.

Resultados de Aprendizaje

RA1	Conocer el proceso de ingeniería implícito al desarrollo de soluciones software
RA2	Dominar las técnicas de modelado de software UML existentes para las etapas de análisis y diseño de aplicaciones
RA3	Dominar las técnicas de modelado de software UML existentes para las etapas de análisis y diseño de aplicaciones
RA4	Desarrollar aplicaciones web que permitan visualizar los resultados de los modelos analíticos
RA5	Desarrollar aplicaciones web que permitan visualizar los resultados de los modelos analíticos
RA6	Desarrollar la lógica de negocio de las aplicaciones web desde el lado del servidor
RA7	Dominar los elementos de programación web (sesiones y cookies) proporcionados por el protocolo HTTP
RA8	Conocer los frameworks más utilizados en el desarrollo de aplicaciones web (front-end y back-end)
RA9	Desarrollar servicios que proporcionen datos provenientes de resultados de modelos para ser consumidos por terceras aplicaciones
RA10	Desplegar aplicaciones en entornos locales y en la nube utilizando las características que facilita en entorno Cloud



BLOQUES TEMÁTICOS Y CONTENIDOS

Contenidos – Bloques Temáticos

1. Metodologías ágiles
 - SCRUM
 - Git Proyect
2. Integración continua
 - CI / CD
 - GitActions
3. Tecnologías web: cliente
 - HTML
 - CSS
 - Javascript
 - Protocolo HTTP
 - React
4. Tecnologías web: servidor
 1. Django
 2. APIREST
 3. Openapi y swagger
 4. Test
5. Despliegues
 - Docker

METODOLOGÍA DOCENTE

Aspectos metodológicos generales de la asignatura

Metodología Presencial: Actividades

Las actividades formativas se desarrollarán durante las 4 horas de clase a la semana que se distribuirán secuencialmente de la siguiente forma:

- **2h Teoría + 2h Práctica individual y colaborativa**

Estas sesiones estarán compuestas por:

- **Clases magistrales expositivas y participativas:**
 - Sesiones de 2h de Teoría.
 - El profesor realizará un exposición de los contenidos teóricos, combinando la clase magistral con la programación en directo (livecoding) de pequeños ejemplos ilustrativos.
 - Los códigos generados en el aula estarán a disposición del alumno.
- **Ejercicios prácticos y resolución de problemas:**
 - El alumno planteará dudas sobre los conceptos aprendidos tanto en las sesiones teóricas como en las prácticas o en el desarrollo del proyecto.
 - Se crearán dinámicas interactivas de resolución de las dudas entre toda la clase con programación de ejemplos por parte del profesor y los alumnos de forma colaborativa.
- **Sesiones Prácticas con uso de software:**
 - Las prácticas de la asignatura se evaluarán acorde a los siguientes criterios/aspectos:



◦ **Aspectos a tener en cuenta:**

- **Calidad de código:** Se valorará la estructura y organización del código, su eficiencia y optimización, así como su sostenibilidad y legibilidad.
- **Documentación:** Cada pareja deberá mantener una documentación actualizada del proceso de desarrollo, incluyendo el diseño inicial, los cambios realizados y los desafíos encontrados. Esta documentación será requerida al final del curso para su revisión.
- **Equidad en la colaboración:** Es importante que ambas personas de la pareja participen activamente en el desarrollo del videojuego. Se recomienda establecer mecanismos de trabajo colaborativo y equitativo para garantizar la participación y el aprendizaje de todos los integrantes.
- **Retroalimentación entre parejas:** Se fomenta la retroalimentación constructiva entre las parejas, lo que permitirá compartir conocimientos, resolver problemas en conjunto y mejorar la calidad de los videojuegos desarrollados.

• **Actividades de evaluación continua del rendimiento:**

- Se realizarán pruebas, desarrollarán prácticas complementarias a las semanales y retos gamificados.
- Test de autoevaluación del tema: el alumno identifica si domina los conceptos teóricos.
- Participación en clase.

- **Tutoría para la resolución de dudas:** esta actividad se realizará de forma implícita durante el resto de actividades descritas.

CG04, CG05, CG07, CE16,
CE18, CE37

Metodología No presencial: Actividades

• **Ejercicios prácticos y resolución de problemas:**

- El alumno recibirá de forma periódica tareas a realizar entre las sesiones presenciales. Estas tareas reflejarán los conocimientos adquiridos por el alumno entre sesión y sesión.
- Dichos ejercicios tendrán que ser enviados para su evaluación antes de la fecha especificada por el profesor. Posteriormente, se subirá la solución a la plataforma de enseñanza.

• **Sesiones prácticas con uso de software:** Esto incluye el trabajo fuera del aula tanto en ejercicios individuales, como el desarrollo individual y por parejas del proyecto final.

- Al finalizar una sesión teórica, el alumno trabajará fuera del aula en la sesión práctica posterior. Dicha práctica pondrá a prueba los conocimientos del alumno sobre lo aprendido en la sesión teórica. El alumno tendrá que llegar a la sesión práctica con el 80% de la misma desarrollada. Esto variará en función de la complejidad de dicha práctica.
- Una vez comience el proyecto final, el alumno tendrá que desarrollar progresivamente y de forma equitativa dicho proyecto con su compañero fuera del aula.
 - Se valorarán los criterios indicados en la metodología presencial.
 - Será necesario demostrar la colaboración de los compañeros en el proyecto final (control de versiones o similar).
- Se valorará negativamente el uso de herramientas o código de terceros siempre que el alumno no sepa justificar la comprensión del mismo.

• **Estudio personal:** El objetivo principal del trabajo no presencial es llegar a entender y comprender los conceptos teóricos de la asignatura, así como ser capaz de poner en práctica estos conocimientos para resolver los diferentes tipos de problemas. Después de cada explicación teórica el profesor subirá a la web todos los códigos desarrollados y el alumno deberá revisarlos y plantearse cuestiones "Whatif" para asimilar mejor los conceptos teóricos. La forma de estudiar la asignatura será la siguiente:

- Estudiar los conceptos explicados con una papel y un lápiz, sin necesidad de trabajar con el ordenador. El alumno debe saber analizar, diseñar y resolver problemas de un forma abstracta sin el ordenador.
- Analizará el código suministrado por el profesor con el fin de asentar los conceptos teóricos.

CG04, CG05, CG07, CE16,
CE18, CE37



- Por último, deberá ser capaz de programar los ejercicios realizados por el profesor desde cero: sin copiar y pegar ni mirar los códigos suministrados.

RESUMEN HORAS DE TRABAJO DEL ALUMNO

HORAS PRESENCIALES				
Clases magistrales expositivas y participativas	Sesiones prácticas con uso de software	Tutorías para resolución de dudas	Ejercicios prácticos y resolución de problemas	Actividades de evaluación continua del rendimiento
40.00	16.00	5.00	4.00	2.00
HORAS NO PRESENCIALES				
Sesiones prácticas con uso de software	Ejercicios prácticos y resolución de problemas	Estudio personal	Proyectos	
32.00	5.00	20.00	56.00	
CRÉDITOS ECTS: 6,0 (180,00 horas)				

EVALUACIÓN Y CRITERIOS DE CALIFICACIÓN

Actividades de evaluación	Criterios de evaluación	Peso
Exámenes: <ul style="list-style-type: none">Prueba Intersemestral.Examen final.	<ul style="list-style-type: none">Prueba Intersemestral (15%):<ul style="list-style-type: none">Su objetivo es evaluar la progresión de los estudiantes hasta ese punto y brindarles una retroalimentación temprana sobre su rendimiento.Se evaluarán los conocimientos aprendidos hasta dicha fecha en la asignatura (SCRUM, HTML, CSS, JavaScript, HTTP y React).Examen final (45%):<ul style="list-style-type: none">Se evaluará la totalidad del contenido de la asignatura, incluyendo el temario de la prueba intersemestral.	60 %
Proyecto final <ul style="list-style-type: none">Que se entregará por parejas al finalizar el curso.	Proyecto final al finalizar el curso. <ul style="list-style-type: none">Se evaluará la calidad del proyecto de acuerdo a las especificaciones.También se tomará en cuenta la documentación y la presentación de la aplicación.	30 %
Sesiones prácticas: <ul style="list-style-type: none">Retos colaborativos	La actitud, participación y realización de las prácticas semanales y los retos planteados en sesiones	10 %



- | | |
|--|-------------------------------|
| <ul style="list-style-type: none">• Trabajos no presenciales• Prácticas | colaborativas e individuales. |
|--|-------------------------------|

Calificaciones

Teoría

- Examen intertrimestral: **15%**
- Examen final: **45%**

Práctica

- Proyecto final: **30%**

Evaluación continua:

- Entregas intermedias del proyecto, participación en clase, resolución de trabajos adicionales que se propongan, entre otros: **10%**

Será necesario obtener una **nota mínima de 5** en el examen de teoría y en el proyecto final. En caso de no superar una de las dos partes, la nota que aparecerá en la convocatoria será de 3,5.

Para validar la autoría del proyecto, se realizará una **Defensa del proyecto**. La defensa se evaluará con **Apto / No Apto** y consistirá en realizar una serie de modificaciones / evolutivos sobre el proyecto final que cada pareja de estudiantes haya presentado. No obstante, la defensa será individual. Una calificación de No apto en la defensa implica la no corrección del proyecto. Por tanto, tener que recuperarlo en la siguiente convocatoria.

Tanto el examen final como el proyecto final son recuperables en convocatoria extraordinaria. En el caso del proyecto final, se asignará un nuevo enunciado.

La evaluación continua no es recuperable. La nota que se obtenga en el examen teórico de la convocatoria extraordinaria supondrá el 60%; equivalente al bloque de teoría (intertrimestral + final).

NOTA ORDINARIA {Nota final ≥ 5 && Nota proyecto final ≥ 5 } = $0,15 * \text{Nota intertrimestral} + 0,45 * \text{Nota final} + 0,30 * \text{Nota proyecto final} + 0,10 * \text{Nota evaluación continua}$.

NOTA EXTRAORDINARIA {Nota final ≥ 5 && Nota proyecto final ≥ 5 } = $0,60 * \text{Nota final} + 0,30 * \text{Nota proyecto final} + 0,10 * \text{Nota evaluación continua}$.

NOTA CONVOCATORIA {Nota final < 5 || Nota proyecto final < 5 }: 3,5

BIBLIOGRAFÍA Y RECURSOS

Bibliografía Básica

- Material proporcionado por el profesor.
- APIs de las distintas tecnologías estudiadas.

En cumplimiento de la normativa vigente en materia de **protección de datos de carácter personal**, le informamos y recordamos que puede consultar los aspectos relativos a privacidad y protección de datos que ha aceptado en su matrícula entrando en esta web y pulsando "descargar"

[https://servicios.upcomillas.es/sedelectronica/inicio.aspx?csv=02E4557CAA66F4A81663AD10CED66792](https://servicios.upcomillas.es/sedeelectronica/inicio.aspx?csv=02E4557CAA66F4A81663AD10CED66792)



COMILLAS

UNIVERSIDAD PONTIFICIA

ICAI

ICADE

CIHS

GUÍA DOCENTE

2024 - 2025
