El presente documento corresponde a la memoria de un título europeo (Joint Degree) creado dentro del marco del **Proyecto Open EU**, una iniciativa de colaboración interuniversitaria europea coordinada por la **Universitat Oberta de Catalunya (UOC)**, en la que la **Universidad Nacional de Educación a Distancia (UNED)** participa con algunas asignaturas junto también a la **Open University de los Países Bajos** (OUNL). Esta cooperación permite articular el plan de estudios integrando aportaciones de las tres instituciones, incluyendo asignaturas por cada una de ellas y los proyectos de fin de grado de la titulación. En coherencia con dicha coordinación, la memoria adopta la **estructura y el formato editorial propios de la UOC**; por ello, su presentación puede diferir del modelo habitual de la UNED (organización de apartados, plantilla y estilo de presentación). Se trata de una diferencia exclusivamente formal que no altera los contenidos, los resultados de aprendizaje ni los criterios de calidad, plenamente alineados con los estándares de ambas universidades y la normativa vigente.

Informe de autoevaluación

Grado de Desarrollo y Pruebas de Software

Universitat Oberta de Catalunya

ÍNDICE

1. Descripción general **2**
1.1. Introducción **2**
1.1.1. Contexto 2
1.1.2. Metas y objetivos del grado 4
1.1.3. Perfiles de los graduados y graduadas 5
1.1.4. Estrategias docentes específicas 5
1.2. Información básica **7**
1.3. Diseño e impartición conjuntos **8**
2. Resultados del aprendizaje **11**
2.1. Conocimientos **11**
2.2. Habilidades **12**
2.3. Competencias **13**
2.4. Profesiones reguladas **13**
3. Programa de estudios **14**
3.1. Plan de estudios **14**
3.2. Créditos 31
3.3. Carga lectiva **32**
3.3.1. Carga lectiva del grado y las asignaturas 32
3.3.2. Duración del grado 32
4. Admisión y reconocimientos **34**
4.1. Admisión **34**
4.1.1. Normativa y procedimientos generales de admisión 34
4.1.2. Criterios específicos de admisión en este grado 34
4.2. Reconocimiento de créditos **35**
4.2.1. Normativa y procedimientos generales de reconocimiento 35
4.2.2. Criterios específicos de reconocimiento de este grado 35
4.3. Movilidad 37
5. Aprendizaje, enseñanza y evaluación **38**
5.1 Aprendizaje y enseñanza **38**
5.2.1. Asignaturas de formación básica, obligatorias y optativas 38
5.1.2. Prácticas obligatorias 41

5.1.3. Trabajo final 41
5.2. Evaluación del estudiantado 42
5.2.1. Evaluación de las asignaturas de formación básica, obligatorias y optativas 42
5.2.2. Evaluación de las prácticas obligatorias 44
5.2.3. Evaluación del trabajo final 44
6. Acompañamiento al estudiantado 46
7. Recursos 47
7.1. Personal 47
7.2. Instalaciones 47
8. Transparencia y documentación 49
9. Garantía de la calidad 50
A.1. Anexos obligatorios 51
A.1.1. Documentos acreditativos de la personalidad jurídica de las instituciones asociadas 51
A.1.2. Convenio de colaboración 51
A.1.3. Documentos acreditativos de la base jurídica de cada socio 51
A.1.4. Lista de resultados del aprendizaje previstos 51
A.1.5. Programas de asignaturas de todos los socios 52
A.1.6. Estructura del plan de estudios 52
A.1.7. Documentos oficiales que indican los requisitos de admisión y los procedimientos de selección 52
A.1.8. Documentos oficiales que describen el procedimiento de reconocimiento de las cualificaciones 53
A.1.9. Normativa de evaluación del estudiantado 53
A.1.10. CV de los miembros del personal académico (todos los socios) 53
A.1.11. Documentos pertinentes que constituyen el sistema interno de garantía de la calidad 53
A.1.12. Suplemento al título (modelo) 53
A.2. Anexos adicionales 54
A.2.1. Actividades, metodologías y métodos de evaluación 54
A.2.2. Prácticas 54

Glosario

ANECA: Agencia Nacional de Evaluación de la Calidad y Acreditación

AQU Catalunya: Agencia para la Calidad del Sistema Universitario de Cataluña

BTSD: Bachelor's Degree in Techniques for Software Application Development (grado de Técnicas de Aplicaciones de Software)

CRUE: Conferencia de Rectores y Rectoras de las Universidades Españolas

EC: evaluación continua

ECTS: European Credit Transfer and Accumulation System (sistema europeo de transferencia y acumulación de créditos)

EF: evaluación final

EQANIE: European Quality Assurance Network For Informatics Education (Red Europea de Garantía de la Calidad de la Educación Informática)

EX: examen

FQ-EHEA: Framework for Qualifications in the European Higher Education Area (Marco de cualificaciones del espacio europeo de educación superior)

ISCED: International Standard Classification of Education (clasificación internacional normalizada de la educación)

MCER: Marco común europeo de referencia para las lenguas

MEC: Marco europeo de cualificaciones

NVAO: Nederlands Vlaamse Accreditatie Organisatie (Organización de Acreditación Holandesa-Flamenca)

OUNL: Universidad Abierta de los Países Bajos

QA: Quality Assurance (garantía de la calidad)

QPED: Quality-focused Programming Education (educación de la programación centrada en la calidad)

PS: prueba de síntesis

TIC: tecnologías de la información y la comunicación

UNED: Universidad Nacional de Educación a Distancia

1. DESCRIPCIÓN GENERAL

1.1. INTRODUCCIÓN

1.1.1. Contexto

En los últimos 50 años, las TIC han tenido un profundo impacto en todos los aspectos de nuestra vida: sociedad, economía, salud o cultura, entre otros. Sin embargo, la adopción de las TIC sigue siendo un proceso continuo con un potencial sin explotar que conocemos como *transformación digital*.

El software es uno de los pilares que hace posible esta transformación digital, ya que automatiza procesos, proporciona servicios y controla y conecta dispositivos. Actualmente, el software constituye el elemento central de la mayoría de los productos, servicios e industrias. Esta tendencia ya se predijo en 2011, en el famoso ensayo "Why software is eating the world" (Por qué el software se está comiendo el mundo), del ingeniero de software y empresario Marc Andreessen.

Como resultado, la demanda de profesionales del desarrollo de software, por ejemplo, es cada vez mayor y en 2021 llegó a duplicarse. Esta demanda creciente ha sido reconocida a escala mundial por miembros de los sectores público y privado como un reto para el éxito de la transformación digital. Veamos algunos ejemplos:

- En 2021, la Comisión Europea adoptó una estrategia para alcanzar los objetivos de transformación digital en Europa para 2030, el Itinerario hacia la Década Digital. Estos objetivos, denominados Brújula Digital 2030, incluyen el propósito en materia de competencias digitales de llegar como mínimo a la cifra de 20 millones de especialistas en TIC empleados en la UE (con una participación equilibrada en términos de género). Sin embargo, en 2019, el 55 % de las empresas que contrataban

a especialistas en TIC declararon tener dificultades para cubrir las vacantes. En particular, la Comisión Europea estimó que el déficit actual entre la oferta y la demanda de profesionales de las TIC en la UE alcanza los 500.000 profesionales. Incluso entre los expertos en TIC se destaca la enorme escasez de profesionales que puedan ocupar posiciones de desarrollador de software.

- La Oficina de Estadísticas Laborales de EE. UU. publica un manual de perspectivas ocupacionales en el que se prevé un crecimiento del 22 % en el número de puestos de trabajo para desarrolladores de software, analistas de control de la calidad y evaluadores entre 2020 y 2030. Para poner el dato en contexto, este crecimiento se considera muy superior a la media, incluso dentro del sector de las TIC.
- En España, informes como el Mapa de Empleo (Fundación Telefónica, Infojobs y Tecnoempleo) o la Guía Spring Professional del Mercado Laboral (Adecco) identifican los perfiles de desarrollador de software, desarrollador web o administrador de sistemas como los más demandados en España, con una tendencia al alza. Además, el X Informe Infoempleo Adecco muestra que los grados universitarios de informática ocupan el tercer lugar en la demanda educativa en España, solo superados por los grados de administración de empresas e ingeniería industrial.
- En los Países Bajos, las posiciones de desarrollador de software y aplicaciones son identificadas como ocupaciones con gran escasez y alta demanda de especialistas con formación superior. El informe de empleos emergentes 2020 de LinkedIn para los Países Bajos señala que el sector tecnológico impulsa la mayoría de las profesiones emergentes (como los ingenieros *full-stack*) y las ofertas de empleo.

Otra tendencia notable en el sector de las TIC es la globalización del puesto de trabajo. La pandemia de la covid-19 impulsó el teletrabajo, lo que ha permitido a las empresas crear equipos internacionales. En este contexto, las compañías ya han dejado de estar limitadas por su ubicación, sus proyectos pueden abarcar varios países y los

desarrolladores tienen que poder colaborar a distancia y de forma eficaz con clientes, compañeros y grupos de interés.

El grado de Desarrollo y Pruebas de Software permite al estudiantado convertirse en profesionales del desarrollo de software en un entorno global. Para ello, proporciona una combinación de habilidades técnicas (algoritmos, prácticas recomendadas de ingeniería de software, experiencia de usuario, diseño de bases de datos, computación en la nube, etc.) y habilidades blandas o *soft skills* (trabajo en equipo, trabajo a distancia, habilidades comunicativas, conocimientos empresariales y del sector, gestión de proyectos, etc.). Desde el punto de vista tecnológico, el estudiantado trabajará en varias plataformas (escritorio, web, móvil y/o sistemas embebidos) y dominará una gran variedad de lenguajes de programación, herramientas y servicios. Además, estará preparado para aprender nuevas tecnologías con el fin de adaptarse a los avances futuros.

Este grado interuniversitario es el sucesor del grado de Técnicas de Aplicaciones de Software (BTSD, por sus siglas en inglés), un título oficial de 180 ECTS ofrecido por la UOC desde 2021 y que quedará extinguido con la aparición del nuevo grado interuniversitario. Aunque el BTSD se impartía en inglés y estaba dirigido a un público internacional, este nuevo título amplía su alcance transfronterizo, incide en la calidad del software mediante pruebas e incorpora la experiencia de la OUNL y la UNED.

1.1.2. Metas y objetivos del grado

Los graduados y graduadas en Desarrollo y Pruebas de Software estarán en disposición de crear software útil, eficiente y de alta calidad en cualquier campo de aplicación. Para ello, deberán ser capaces de analizar problemas con un nivel adecuado de abstracción, identificar los aspectos relevantes del problema, evaluar las alternativas posibles teniendo en cuenta el estado actual de la técnica y seleccionar e implementar la solución más adecuada. Además, también deberán tener en cuenta la implementación, la administración y el mantenimiento de los sistemas que desarrollen.

Para alcanzar este objetivo, el estudiantado aprenderá los principios del desarrollo de software (programación, algoritmos, gestión de datos, ingeniería de software y usabilidad) y los fundamentos de las tecnologías de la información (estructura de ordenadores, sistemas operativos, redes y nube). Trabajará con diversas plataformas (escritorio, web, móvil y/o sistemas embebidos) utilizando una variedad de lenguajes de programación, herramientas y servicios. Asimismo, como el desarrollo de software es un campo en rápida evolución, los graduados y graduadas deberán ser capaces de adaptarse a las tecnologías emergentes y a los entornos cambiantes.

Además, aplicarán técnicas de gestión de proyectos y las mejores prácticas de desarrollo de software para garantizar la calidad y la puntualidad en la entrega del producto de software. Dado que el desarrollo de software es un trabajo en equipo, también deberán ser capaces de comunicarse y colaborar eficazmente dentro de un equipo y con los grupos de interés externos.

Por último, debido al potencial impacto del software en la sociedad, los graduados y graduadas también aprenderán a tener en cuenta en su actividad profesional aspectos como la sostenibilidad, la inclusividad, la perspectiva de género o la accesibilidad.

1.1.3. Perfiles de los graduados y graduadas

Este grado proporciona las habilidades y los conocimientos necesarios para trabajar como **desarrollador de software**. Este título abarca una amplia gama de puestos, entre los que se incluyen programadores, ingenieros de software, analistas, desarrolladores web, desarrolladores de aplicaciones, controladores de calidad (QA) y probadores de software.

El programa también proporciona las habilidades necesarias para **administrar** los sistemas en los que se implementarán las aplicaciones de software. Por consiguiente, estos estudios preparan al alumnado para trabajar en puestos como administrador de sistemas o administrador de redes.

A la hora de diseñar el plan de estudios, hemos tenido en cuenta los derechos fundamentales y la igualdad de oportunidades entre

hombres y mujeres, los principios de igualdad de oportunidades y accesibilidad universal para las personas con diversidad funcional, los valores de una cultura democrática de paz y los principios de sostenibilidad, de acuerdo con la legislación vigente y las directrices para la introducción de la sostenibilidad en el plan de estudios elaboradas por la CRUE.

1.1.4. Estrategias docentes específicas

Una de las principales preocupaciones de este grado es la enseñanza de técnicas, metodologías, herramientas y mejores prácticas para garantizar la *calidad del software*. Los errores de software (los llamados *bugs*) pueden tener consecuencias devastadoras: pérdidas económicas, vulnerabilidades de seguridad que pueden ser explotadas por terceros malintencionados o, en sistemas esenciales para la seguridad, la pérdida de vidas humanas. El estudiantado debe ser consciente de esta realidad, así como de las técnicas utilizadas para evitar, detectar y corregir los errores en el software.

Con este fin, nos basamos en la experiencia del proyecto europeo Erasmus+ QPED (2020-2023), dirigido por la OUNL (Países Bajos) y con la participación de la UOC (España), la Universidad Philipps de Marburgo (Alemania), la Universidad Tecnológica de Eindhoven (TUE, Países Bajos) y Quarterfall (Países Bajos). Este proyecto se centra en proporcionar recursos y herramientas para mejorar la educación en programación e introduce el concepto de *pruebas de software* desde las primeras asignaturas. Además, el proyecto tiene como objetivo fomentar la calidad del software más allá de su correcto funcionamiento teniendo en cuenta conceptos como la *legibilidad* o la *solidez del código*.

Un aspecto importante de este enfoque cualitativo en la enseñanza de la programación es el uso de herramientas de evaluación automática. Estas herramientas permiten una evaluación formativa continua, ya que proporcionan un retorno frecuente, informativo y oportuno sobre las tareas de programación. Este retorno automático sirve de complemento útil al *feedback* cualitativo más exhaustivo proporcionado por el profesorado de las asignaturas.

1.2. INFORMACIÓN BÁSICA

Nombre completo del programa:	Grado de Desarrollo y Pruebas de Software
Plazas:	450
Nivel de MEC:	6 (grado de primer ciclo)
Títulos obtenidos:	Grado
Número de créditos ECTS:	180 ECTS (3 años)
Campo(s) de estudio ISCED:	0613 Desarrollo y análisis de software y aplicaciones

Lista de instituciones que imparten el programa:

Nombre de la institución	Institución de educación superior	Institución que expide el título	Función en el consorcio
Universitat Oberta de Catalunya (UOC)	sí	sí	coordinador
Universidad Abierta de los Países Bajos (OUNL)	sí	sí	miembro
Universidad Nacional de Educación a Distancia (UNED)	sí	no	miembro

Estado de acreditación por institución:

Nombre de la institución	Agencia externa de garantía de la calidad relevante	Enfoque europeo permitido	Información adicional
Universitat Oberta de Catalunya (UOC)	Agencia para la Calidad del Sistema Universitario de Cataluña (AQU Catalunya, España)	sí	Sistema de garantía interna de la calidad Reputación: universidad pionera en la enseñanza en línea
Universidad Abierta de los Países Bajos (OUNL)	Nederlands Vlaamse Accreditatie Organisatie (NVAO)	sí	
Universidad Nacional de Educación a Distancia (UNED)	Agencia Nacional de Evaluación de la Calidad y Acreditación (ANECA, España)	sí	

1.3. DISEÑO E IMPARTICIÓN CONJUNTOS

Este grado aprovecha la amplia experiencia de la UOC, la OUNL y la UNED en la oferta de titulaciones de TIC en modalidad virtual.

- La UOC ofrece grados de TIC desde 1997. Actualmente, sus titulaciones de grado cuentan con más de 6.400 estudiantes, 6.400 graduados y 600 docentes. Estas titulaciones han recibido acreditaciones positivas de organismos españoles de garantía de la calidad, así como el sello de calidad Euro-Inf de EQANIE.
- La OUNL ofrece dos programas de grado (180 ECTS), de Informática y Ciencias de la Información, y cuatro programas de máster: Inteligencia Artificial (60 ECTS), que comenzó en 2022-2023; Ingeniería de Software (60 ECTS); Gestión de Procesos de Negocio y TI (60 ECTS), e Informática (120 ECTS). Las cifras reales se facilitarán más adelante. En el curso

académico 2022-2023, 826 estudiantes cursaron el programa de grado de Informática y 255, el programa de grado de Ciencias de la Información. El número de títulos de grado concedidos desde 2014-2015 es de 68 de Informática y 8 de Ciencias de la Información. El número de estudiantes matriculados en el curso académico 2022-2023 en los cuatro programas de máster mencionados fue de 687, y desde 2014-2015 se han expedido 942 títulos de máster.

- La UNED se fundó en 1972 con el objetivo de impartir enseñanzas a distancia en español en todo el mundo a través de la radio, la televisión y la comunicación postal. En la actualidad, la UNED aprovecha las innovaciones técnicas para formar a los casi 4.000 estudiantes de todo el mundo matriculados en sus dos titulaciones de grado de TIC durante el curso académico 2021-2022: Informática y Tecnologías de la Información. Estas titulaciones las imparten 90 docentes con la ayuda de 1.000 tutores (que orientan sobre los contenidos y evalúan los trabajos de las asignaturas) distribuidos en centros asociados situados en 75 ciudades de 17 países, donde el estudiantado puede realizar los exámenes. Los estudios están acreditados por la ANECA, la agencia oficial española. El número de estudiantes de la UNED que han finalizado estudios de grado de TIC supera los 65.000.

Este título se ha diseñado en el marco de la OpenEU Erasmus+ European University, una alianza de universidades europeas de aprendizaje virtual que inició su actividad el 1 de diciembre de 2024. El consorcio OpenEU está coordinado por la UOC y cuenta con la participación de otras universidades, entre las que se encuentran la OUNL y la UNED. En particular, las principales responsabilidades de las universidades en el marco del consorcio OpenEU son las siguientes:

- La UOC coordina el paquete de trabajo 1 sobre "Gobernanza y gestión para una universidad europea abierta, sólida y sostenible", y participa activamente en todos los demás paquetes de trabajo. Además, como coordinadora del

consorcio, durante los primeros años de la alianza, la UOC acogerá la secretaría de OpenEU.

- La OUNL coordina el paquete de trabajo sobre "Investigación e innovación conjuntas en educación digital" y participa activamente en los distintos paquetes de trabajo del consorcio. La OUNL también es miembro de todos los consejos de la alianza.
- La UNED coordina el paquete de trabajo de "Internacionalización" y codirige el de "Hacer realidad el aprendizaje permanente para la empleabilidad". También participa activamente en todos los paquetes de trabajo, además de ser miembro de todos los consejos de la alianza.

2. RESULTADOS DEL APRENDIZAJE

Los resultados del aprendizaje de este grado comprenden conocimientos, habilidades y competencias en el campo del **desarrollo y análisis de software y aplicaciones**, una subárea dentro de las TIC.

El nivel de estos resultados del aprendizaje se corresponde con el **nivel 6 de MEC** (primer ciclo de educación superior/grado) del FQ-EHEA.

A lo largo de este grado interuniversitario, la consecución de estos resultados del aprendizaje se demuestra a través de distintos medios y estrategias de evaluación (véase la sección 5.2 sobre evaluación).

2.1. CONOCIMIENTOS

Al finalizar el grado, el estudiantado será capaz de:

- **K1**. Reconocer los principios científicos y matemáticos que constituyen la base del proceso de desarrollo de software.
- **K2**. Enumerar y describir los elementos estructurales y los principios operativos de los ordenadores y las redes, junto con sus características relevantes desde el punto de vista del software.
- **K3**. Identificar las características de los diferentes tipos de organizaciones y el papel que desempeñan en ellas las TIC.
- **K4**. Enumerar y describir las etapas del ciclo de vida del desarrollo de software, incluyendo los objetivos, las tareas y los retos que conlleva cada etapa.
- **K5**. Describir los lenguajes de programación, las herramientas y las tecnologías utilizados en el proceso de desarrollo de software, junto con los escenarios en los que son más adecuados.

2.2. HABILIDADES

Al finalizar el grado, el estudiantado será capaz de:

- **S1**. Diseñar, desarrollar y probar aplicaciones de software para diferentes plataformas y ámbitos de aplicación empleando las mejores prácticas de ingeniería de software.
- **S2**. Verificar, validar y medir la calidad del software durante el proceso de desarrollo según diferentes criterios, como la corrección, el rendimiento, la usabilidad o la seguridad.
- **S3**. Diseñar, configurar y gestionar redes y sistemas informáticos teniendo en cuenta los recursos disponibles y garantizando su disponibilidad y seguridad.
- **S4**. Aplicar técnicas para el procesamiento, el almacenamiento y la gestión de datos, así como el acceso a dichos datos, teniendo en cuenta el rendimiento y la disponibilidad.
- **S5**. Diseñar la experiencia de usuario de una aplicación de software teniendo en cuenta los objetivos, las necesidades y las expectativas de los usuarios.
- **S6**. Analizar los problemas de desarrollo con un nivel de abstracción adecuado para seleccionar la alternativa tecnológica más apropiada (lenguaje de programación, algoritmo, herramienta o componente).
- **S7**. Preparar propuestas para proyectos de desarrollo teniendo en cuenta los recursos, las alternativas disponibles, las condiciones del mercado y el estado actual de la ciencia.
- **S8**. Expresar ideas de forma clara y correcta en inglés, tanto oralmente como por escrito, demostrando el dominio de los términos y expresiones específicos del campo de desarrollo de software, tanto en contextos académicos como profesionales.

2.3. COMPETENCIAS

Al finalizar el grado, el estudiantado será capaz de:

- **C1**. Colaborar eficazmente tanto en entornos académicos como en equipos u organizaciones de desarrollo de software.
- **C2**. Planificar y llevar a cabo proyectos de desarrollo de software, presentar los resultados de forma adecuada al público destinatario (tanto en contextos académicos como profesionales) y defender las conclusiones ante un grupo de expertos.
- **C3**. Adaptarse a las nuevas tecnologías de desarrollo de software y a los cambios en las necesidades del mercado, el estado actual de la ciencia y la sociedad.
- **C4**. Aplicar las TIC en el ámbito académico y profesional.
- **C5**. Desempeñar la práctica académica y profesional de forma honesta, ética, sostenible y socialmente responsable, siendo consciente de la diversidad y los derechos humanos y respetándolos.

2.4. PROFESIONES REGULADAS

No hay profesiones reguladas relacionadas con este título.

3. PROGRAMA DE ESTUDIOS

3.1. PLAN DE ESTUDIOS

La tabla siguiente resume la organización temporal del plan de estudios del grado de Desarrollo y Pruebas de Software:

	Semestre 1	Semestre 2
Año 1	ECTS: 30 Asignaturas: • Trabajo en equipo en línea y habilidades académicas (formación básica, S7) • Ingeniería de software (formación básica, S3) • Fundamentos de programación (formación básica, S2) • Fundamentos de computadores (formación básica, S6) • Fundamentos de matemáticas I (formación básica, S1)	ECTS: 30 Asignaturas: • Fundamentos de los sistemas de información (formación básica, S7) • Lenguajes y estándares web (obligatoria, S4) • Prácticas de programación (formación básica, S2) • Estructura de computadores (obligatoria, S6) • Fundamentos de matemáticas II (formación básica, S1)

	Semestre 3	Semestre 4
Año 2	ECTS: 30 Asignaturas: • Programación web (obligatoria, S4) • Programación orientada a objetos (obligatoria, S2) • Patrones de diseño de software (obligatoria, S3) • Sistemas operativos (obligatoria, S6) • Factores humanos en sistemas informáticos (obligatoria, S3)	ECTS: 30 Asignaturas: • Redes y aplicaciones de Internet (obligatoria, S6) • Introducción a las bases de datos (obligatoria, S5) • Estructuras de datos (obligatoria, S2) • Arquitectura de software (obligatoria, S3) • Administración de redes y sistemas (obligatoria, S6)

	Semestre 5	Semestre 6

Año 3	ECTS: 30 Asignaturas: ● Diseño de bases de datos (obligatoria, S5) ● Pruebas de software (obligatoria, S3) ● Computación en la nube (obligatoria, S6) ● Asignatura optativa I (optativa) ● Asignatura optativa II (optativa)	ECTS: 30 Asignaturas: ● Desarrollo de aplicaciones móviles (obligatoria, S8) ● Asignatura optativa III (optativa) ● Asignatura optativa IV (optativa) ● Trabajo final (obligatoria, S9)

Las asignaturas de este grado se organizan conceptualmente en las siguientes 9 materias: (S1) fundamentos de matemáticas, (S2) programación, (S3) ingeniería de software, (S4) desarrollo web, (S5) bases de datos, (S6) organización y administración de computadores, (S7) habilidades empresariales y blandas, (S8) ámbitos de aplicación y (S9) proyectos. A continuación se describen las materias en detalle. La tabla anterior incluye a qué materia pertenece cada asignatura.

Materia 1. Fundamentos de matemáticas	
Número de créditos ECTS	12
Tipo	Formación básica
Calendario	Semestres 1 y 2
Modalidad	Virtual
Resultados del aprendizaje del grado relacionados con esta materia	• K1. Reconocer los principios científicos y matemáticos que constituyen la base del proceso de desarrollo de software. • S6. Analizar los problemas de desarrollo con un nivel de abstracción adecuado para seleccionar la alternativa tecnológica más apropiada (lenguaje de programación, algoritmo, herramienta o componente). • S8. Expresar ideas de forma clara y correcta en inglés, tanto oralmente como por escrito, demostrando dominio de los términos y expresiones específicos del campo de desarrollo de software, tanto en contextos académicos como profesionales.
Resultados del aprendizaje	Después de cursar esta materia, el estudiantado será capaz de: • Reconocer, explicar y utilizar los conceptos matemáticos y la notación utilizados para expresar conocimientos sobre un problema concreto. • Explicar las diferencias y la relación entre los distintos tipos de números (es decir, naturales, enteros, reales y complejos). • Resolver sistemas de ecuaciones lineales. • Aplicar conceptos de álgebra lineal (base, transformaciones lineales, diagonalización, etc.) para resolver problemas complejos. • Comprobar propiedades y realizar operaciones básicas en conjuntos. • Manipular expresiones booleanas y explicar el significado de una tabla de verdad. • Formalizar enunciados complejos en lenguaje natural utilizando una notación lógica. • Dado un conjunto de enunciados lógicos, realizar o validar un razonamiento para llegar a conclusiones sólidas.

Asignaturas	Todas las asignaturas son de 6 ECTS y se imparten en inglés. ● Fundamentos de matemáticas I (formación básica: matemáticas, semestre 1) ● Fundamentos de matemáticas II (formación básica: matemáticas, semestre 2)
Contenido	**Fundamentos de matemáticas I** ● Números. ● Herramientas algebraicas y geométricas. ● Sistemas de ecuaciones. ● Aplicaciones lineales. ● Transformaciones geométricas. **Fundamentos de matemáticas II** ● Teoría básica de conjuntos: operaciones con conjuntos, relaciones, funciones. ● Tablas de verdad y álgebra de Boole. ● Lenguaje y semántica de la lógica proposicional. ● Lenguaje y semántica de la lógica de primer orden. ● Formalización de expresiones del lenguaje natural. ● Demostración de teoremas.

Materia 2. Programación	
Número de créditos ECTS	30
Tipo	Mixta (6 créditos de formación básica, 18 obligatorios y 6 optativos)
Calendario	Semestres 1, 2, 3, 4 y 5/6
Modalidad	Virtual
Resultados del aprendizaje del grado relacionados con esta materia	● K5. Describir los lenguajes de programación, las herramientas y las tecnologías utilizados en el proceso de desarrollo de software, junto con los escenarios en los que son más adecuados. ● S1. Diseñar, desarrollar y probar aplicaciones de software para diferentes plataformas y ámbitos de aplicación empleando las mejores prácticas de ingeniería de software. ● S2. Verificar, validar y medir la calidad del software durante el proceso de desarrollo según diferentes criterios, como la corrección, el rendimiento, la usabilidad o la seguridad. ● S4. Aplicar técnicas para el procesamiento, el almacenamiento y la gestión de datos, así como el acceso a dichos datos, teniendo en cuenta el rendimiento y la disponibilidad. ● S6. Analizar los problemas de desarrollo con un nivel de abstracción adecuado para seleccionar la alternativa tecnológica más apropiada (lenguaje de programación, algoritmo, herramienta o componente). ● S8. Expresar ideas de forma clara y correcta en inglés, tanto oralmente como por escrito, demostrando dominio de los términos y

	expresiones específicos del campo de desarrollo de software, tanto en contextos académicos como profesionales. ● C3. Adaptarse a las nuevas tecnologías de desarrollo de software y a los cambios en las necesidades del mercado y la sociedad.
Resultados del aprendizaje	Después de cursar esta materia, el estudiantado será capaz de: ● Reconocer y explicar distintos paradigmas de programación (imperativo, orientado a objetos, funcional), entendiendo sus diferencias y puntos fuertes. ● Proponer algoritmos para resolver un problema dado. ● Escribir algoritmos en un lenguaje de programación dado aprovechando sus características. ● Utilizar las herramientas más adecuadas para crear, ejecutar, mantener y analizar programas informáticos (editores, IDE, compiladores, intérpretes, depuradores, etc.). ● Detectar, comprender y corregir errores en programas informáticos. ● Medir la complejidad temporal y espacial de algoritmos alternativos para un problema concreto con el fin de seleccionar la mejor alternativa. ● Describir y utilizar los algoritmos y las estructuras de datos más conocidos para resolver problemas fundamentales (ordenación, búsqueda, etc.). ● Aplicar las mejores prácticas al escribir código informático (convenciones de nomenclatura, comentarios, modularidad y organización del código, etc.).
Asignaturas	Todas las asignaturas son de 6 ECTS y se imparten en inglés. ● Fundamentos de programación (formación básica: computación, semestre 1) ● Prácticas de programación (formación básica, semestre 2) ● Programación orientada a objetos (obligatoria, semestre 3) ● Estructuras de datos (obligatoria, semestre 4)
Contenido	**Fundamentos de programación** ● Fundamentos de los algoritmos: tipos de datos, declaraciones, expresiones, estructuras algorítmicas (asignación, sentencias condicionales, bucles, etc.), procedimientos y funciones. ● Secuencias: algoritmos de búsqueda/recorrido. ● Tipos de datos estructurados: tablas y tuplas. ● Diseño descendente. ● Herramientas de desarrollo: IDE, compilador, intérprete. ● Lenguajes de programación. ● Convenciones de codificación (códigos, tabulaciones, nomenclatura, mejores prácticas). **Prácticas de programación** ● Historia de los lenguajes de programación. ● Diseño descendente. ● Estructuras básicas de datos: listas, pilas, colas. ● Complejidad algorítmica.

	• Algoritmos fundamentales: ordenación (ordenación por selección, ordenación por inserción, ordenación por burbuja, ordenación rápida, ordenación por fusión). • Herramientas de desarrollo: IDE, compilador, intérprete, depurador. • La gestión de la memoria. • Entrada/salida. • Implementación de algoritmos complejos. **Programación orientada a objetos** • Diseño de software: criterios de calidad, modularidad. • Paradigma orientado a objetos: clases y objetos, herencia, instanciación. • Diagramas de clases UML y su implementación en el lenguaje Java. • Lenguaje de programación Java: fundamentos, tipos de datos, operadores, matrices, sentencias, métodos, visibilidad. **Estructuras de datos** • Tipos de datos abstractos (TDA): ventajas, especificación. • TDA secuenciales: pilas, colas, listas. • Árboles: tipos (binarios frente a n-arios, ordenados, de búsqueda), recorridos de árboles. • Colas de prioridad: montículos, algoritmo de ordenación por montículos. • Diccionarios: funciones hash, tablas hash y árboles AVL. • Grafos: recorridos de grafos. • Diseño de estructuras de datos: criterios para seleccionar el TDA más adecuado. • Uso y especificación de bibliotecas de TDA. • Implementación de TDA en Java.

Materia 3. Ingeniería de software	
Número de créditos ECTS	30
Tipo	Mixta (6 créditos de formación básica y 24 obligatorios)
Calendario	Semestres 1, 3, 4 y 5
Modalidad	Virtual
Resultados del aprendizaje del grado relacionados con esta materia	• K4. Enumerar y describir las etapas del ciclo de vida del desarrollo de software, incluyendo los objetivos, las tareas y los retos que conlleva cada etapa. • K5. Describir los lenguajes de programación, las herramientas y las tecnologías utilizados en el proceso de desarrollo de software, junto con los escenarios en los que son más adecuados. • S1. Diseñar, desarrollar y probar aplicaciones de software para diferentes plataformas y ámbitos de aplicación empleando las mejores prácticas de ingeniería de software.

	• S2. Verificar, validar y medir la calidad del software durante el proceso de desarrollo según diferentes criterios, como la corrección, el rendimiento, la usabilidad o la seguridad. • S5. Diseñar la experiencia de usuario de una aplicación de software teniendo en cuenta los objetivos, las necesidades y las expectativas de los usuarios. • S6. Analizar los problemas de desarrollo con un nivel de abstracción adecuado para seleccionar la alternativa tecnológica más apropiada (lenguaje de programación, algoritmo, herramienta o componente). • S7. Preparar propuestas para proyectos de desarrollo teniendo en cuenta los recursos, las alternativas disponibles y las condiciones del mercado. • S8. Expresar ideas de forma clara y correcta en inglés, tanto oralmente como por escrito, demostrando dominio de los términos y expresiones específicos del campo de desarrollo de software, tanto en contextos académicos como profesionales. • C2. Planificar y llevar a cabo proyectos de desarrollo de software, presentar los resultados de forma adecuada al público destinatario y defender las conclusiones ante un grupo de expertos. • C3. Adaptarse a las nuevas tecnologías de desarrollo de software y a los cambios en las necesidades del mercado y la sociedad. • C5. Desempeñar la práctica académica y profesional de forma honesta, ética, sostenible y socialmente responsable, siendo consciente de los derechos humanos, la perspectiva de género y la diversidad y respetándolos.
Resultados del aprendizaje	Después de cursar esta materia, el estudiantado será capaz de: • Enumerar y describir las etapas del ciclo de vida del desarrollo de software, incluyendo los objetivos, las tareas y los retos que conlleva cada etapa. • Diseñar los componentes estáticos y dinámicos de un sistema de software utilizando los diagramas UML estructurales y de comportamiento adecuados. • Aplicar metodologías ágiles de gestión de proyectos en proyectos de desarrollo de software. • Enumerar, describir y aplicar (cuando sea posible) patrones de diseño de software en el diseño de sistemas de software. • Diseñar la experiencia de usuario de una aplicación de software teniendo en cuenta los objetivos, las necesidades y las expectativas de los usuarios. • Evaluar la experiencia del usuario de un sistema de software determinado. • Diseñar e implementar la arquitectura de sistemas distribuidos complejos. • Definir diferentes tipos de pruebas y estrategias de prueba para comprobar el correcto funcionamiento de un sistema de software.
Asignaturas	Todas las asignaturas son de 6 ECTS y se imparten en inglés. • Ingeniería de software (formación básica, semestre 1) • Factores humanos en sistemas informáticos (obligatoria, semestre 3) • Patrones de diseño de software (obligatoria, semestre 3) • Arquitectura de software (obligatoria, semestre 4)

	• Pruebas de software (obligatoria, semestre 5)
Contenido	**Ingeniería de software** • Introducción a la ingeniería de software orientada a objetos (ciclo de vida del software, herramientas CASE, estándares OMG y UML). • Paradigma orientado a objetos: clases y objetos, abstracción y clasificación, herencia y polimorfismo. • UML: diagramas estáticos y dinámicos. • Ingeniería de requisitos: recopilación y documentación de los requisitos del software. • Diseño orientado a objetos. • Aplicación a casos de uso en el desarrollo de software. • Modelado en UML (introducción a las herramientas CASE y mejores prácticas). **Factores humanos en sistemas informáticos** • Introducción a la interacción humano-computadora. • Accesibilidad y usabilidad. • Obtención de los requisitos del usuario. • Metáforas, estilos y paradigmas. • Prototipado. • Evaluación. **Patrones de diseño de software** • Patrones: concepto y tipos. • Diseño de software utilizando patrones. • Catálogo de patrones. • Casos prácticos de aplicación de patrones de diseño de software. **Arquitectura de software** • Arquitecturas de software distribuido. • Arquitecturas de microservicios: concepto, diseño, implementación, conexión de microservicios. • Calidad en el software distribuido. • DevOps, entrega continua/integración continua (CD/CI) y desarrollo ágil. **Pruebas de software** • Descripción general de las pruebas. • Modelos para pruebas. • Diseño de casos de prueba. • Criterios de cobertura. • Calidad de los conjuntos de pruebas.

Materia 4. Desarrollo web

Número de créditos ECTS	12
Tipo	Obligatoria
Calendario	Semestres 2 y 3
Modalidad	Virtual
Resultados del aprendizaje del grado relacionados con esta materia	• K5. Describir los lenguajes de programación, las herramientas y las tecnologías utilizados en el proceso de desarrollo de software, junto con los escenarios en los que son más adecuados. • S1. Diseñar, desarrollar y probar aplicaciones de software para diferentes plataformas y ámbitos de aplicación empleando las mejores prácticas de ingeniería de software. • S2. Verificar, validar y medir la calidad del software durante el proceso de desarrollo según diferentes criterios, como la corrección, el rendimiento, la usabilidad o la seguridad. • S5. Diseñar la experiencia de usuario de una aplicación de software teniendo en cuenta los objetivos, las necesidades y las expectativas de los usuarios. • S6. Analizar los problemas de desarrollo con un nivel de abstracción adecuado para seleccionar la alternativa tecnológica más apropiada (lenguaje de programación, algoritmo, herramienta o componente). • S8. Expresar ideas de forma clara y correcta en inglés, tanto oralmente como por escrito, demostrando dominio de los términos y expresiones específicos del campo de desarrollo de software, tanto en contextos académicos como profesionales.
Resultados del aprendizaje	Después de cursar esta materia, el estudiantado será capaz de: • Enumerar y explicar los diferentes estándares, lenguajes y tecnologías implicados en el desarrollo de una aplicación web. • Diseñar y construir interfaces para aplicaciones web utilizando HTML y CSS. • Diseñar sitios web complejos, incluyendo tablas o formularios, utilizando HTML y CSS. • Validar el HTML/CSS y comprobar la accesibilidad de un sitio web. • Desarrollar aplicaciones del lado del cliente utilizando JavaScript.
Asignaturas	Todas las asignaturas son de 6 ECTS y se imparten en inglés. • Lenguajes y estándares web (obligatoria, semestre 2) • Programación web (obligatoria, semestre 3)

Contenido	**Lenguajes y estándares web** ● Introducción y conceptos básicos. ● Fundamentos de HTML y CSS. ● Listas, imágenes, enlaces, div, span y el modelo de caja en CSS. ● Validación y accesibilidad. ● Tablas. ● Formularios. ● Diseño. **Programación web** ● Programación del lado del cliente: lenguajes de scripting en navegadores. ● Introducción a la programación orientada a objetos. ● Orientación a objetos en JavaScript. ● Introducción al DOM. ● Manipulación del DOM con JavaScript.

Materia 5. Bases de datos	
Número de créditos ECTS	12
Tipo	Obligatoria
Calendario	Semestres 4 y 5
Modalidad	Virtual
Resultados del aprendizaje del grado relacionados con esta materia	● K5. Describir los lenguajes de programación, las herramientas y las tecnologías utilizados en el proceso de desarrollo de software, junto con los escenarios en los que son más adecuados. ● S4. Aplicar técnicas para el procesamiento, el almacenamiento y la gestión de datos, así como el acceso a dichos datos, teniendo en cuenta el rendimiento y la disponibilidad. ● S6. Analizar los problemas de desarrollo con un nivel de abstracción adecuado para seleccionar la alternativa tecnológica más apropiada (lenguaje de programación, algoritmo, herramienta o componente). ● S8. Expresar ideas de forma clara y correcta en inglés, tanto oralmente como por escrito, demostrando dominio de los términos y expresiones específicos del campo de desarrollo de software, tanto en contextos académicos como profesionales.

Resultados del aprendizaje	Después de cursar esta materia, el estudiantado será capaz de: ● Explicar la función y los tipos de sistemas de gestión de bases de datos (por ejemplo, NoSQL). ● Explicar los fundamentos del modelo relacional y las operaciones en álgebra relacional. ● Escribir sentencias SQL complejas para consultar una base de datos, crear tablas, insertar/eliminar/actualizar datos o definir procedimientos almacenados. ● Definir transacciones para agrupar sentencias SQL relacionadas de forma atómica. ● Ejecutar sentencias SQL desde código informático (por ejemplo, JDBC). ● Proponer el diseño lógico de una base de datos teniendo en cuenta cuestiones de calidad (es decir, normalización). ● Configurar el control de acceso y el diseño físico de una base de datos para garantizar su seguridad y eficiencia. ● Expresar y comprobar las restricciones de integridad utilizando construcciones SQL y disparadores.
Asignaturas	Todas las asignaturas son de 6 ECTS y se imparten en inglés. ● Introducción a las bases de datos (obligatoria, semestre 4) ● Diseño de bases de datos (obligatoria, semestre 5)
Contenido	**Introducción a las bases de datos** ● Introducción a los datos, las bases de datos y los sistemas de gestión de bases de datos. ● Modelo relacional y álgebra relacional. ● Lenguaje SQL (DML): crear tabla, insertar, eliminar, actualizar, seleccionar, procedimientos almacenados. ● Transacciones. ● Acceso a una base de datos desde un programa (JDBC). **Diseño de bases de datos** ● Diseño lógico de bases de datos (transformación del modelo conceptual al modelo lógico: estructuras de datos, restricciones de integridad y operaciones). ● Reconsideración del modelo lógico y conceptual (normalización, desnormalización). ● Especificación e implementación del control de acceso en bases de datos. ● Diseño físico de una base de datos (por ejemplo, índices, etc.). ● Especificación e implementación de restricciones de integridad en una base de datos (disparadores). ● Procesamiento de consultas y comandos SQL.

Materia 6. Organización y administración de computadores	
Número de créditos ECTS	42

Tipo	Mixta (6 créditos de formación básica, 30 obligatorios y 6 optativos)
Calendario	Semestres 1, 2, 3, 4, 5 y 6
Modalidad	Virtual
Resultados del aprendizaje del grado relacionados con esta materia	• K1. Reconocer los principios científicos y matemáticos que constituyen la base del proceso de desarrollo de software. • K2. Enumerar y describir los elementos estructurales y los principios operativos de los ordenadores y las redes, junto con sus características relevantes desde el punto de vista del software. • S3. Diseñar, configurar y gestionar redes y sistemas informáticos teniendo en cuenta los recursos disponibles y garantizando su disponibilidad y seguridad. • S6. Analizar los problemas de desarrollo con un nivel de abstracción adecuado para seleccionar la alternativa tecnológica más apropiada (lenguaje de programación, algoritmo, herramienta o componente). • S8. Expresar ideas de forma clara y correcta en inglés, tanto oralmente como por escrito, demostrando dominio de los términos y expresiones específicos del campo de desarrollo de software, tanto en contextos académicos como profesionales.
Resultados del aprendizaje	Después de cursar esta materia, el estudiantado será capaz de: • Codificar números enteros y decimales en formato binario y realizar operaciones con ellos. • Diseñar circuitos digitales utilizando puertas lógicas y bloques combinacionales. • Enumerar los componentes básicos del hardware de un ordenador (por ejemplo, procesador, memoria, dispositivos de E/S), explicar sus características relevantes desde el punto de vista de un programador y manejarlos utilizando el lenguaje ensamblador. • Explicar los componentes básicos de los sistemas operativos y acceder a ellos desde un programa informático. • Explicar la arquitectura y los diferentes tipos de redes informáticas. • Diseñar e implementar aplicaciones y sistemas que se comunican mediante redes informáticas. • Configurar sistemas y redes informáticas de acuerdo con las necesidades y normativas de los usuarios y la organización, garantizando su seguridad, eficiencia, fiabilidad y disponibilidad. • Seleccionar, configurar y gestionar servicios en la nube.
Asignaturas	Todas las asignaturas son de 6 ECTS y se imparten en inglés. • Fundamentos de computadores (formación básica: computación, semestre 1) • Estructura de computadores (obligatoria, semestre 2) • Sistemas operativos (obligatoria, semestre 3) • Redes y aplicaciones de Internet (obligatoria, semestre 4) • Administración de redes y sistemas (obligatoria, semestre 4) • Computación en la nube (obligatoria, semestre 5) • Seguridad en redes de computadores (optativa, semestre 5/6) • Sistemas empotrados (optativa, semestre 5/6)

Contenido	**Fundamentos de computadores** ● Conceptos básicos de un computador: introducción a los computadores, estructura jerárquica de un computador, arquitectura Von Newmann. ● Representación digital de la información: cambio de base, codificación binaria de miembros naturales y enteros, operaciones, representación en punto flotante: precisión, rango. ● Circuitos combinacionales: tablas de verdad, diseño de circuitos a dos niveles; método de Karnaugh; bloques combinacionales básicos (multiplexor, codificador, sumador). ● Circuitos secuenciales: D flip-flop, registro, modelo de Moore. ● Estructura básica de un computador: descripción de la arquitectura CISCA, introducción al lenguaje ensamblador utilizando el conjunto de instrucciones CISCA. **Estructura de computadores** ● Repertorio de instrucciones y modos de direccionamiento. ● Sistemas de E/S (programada, con interrupciones, con acceso directo a memoria); comparativas y evaluación del rendimiento. ● Jerarquía de memorias: tipo de memorias, características de las memorias, acceso a los datos de la memoria, concepto de *jerarquía de memorias*, análisis de su efectividad. ● Programación en ensamblador: control de flujo de un programa en ensamblador, programación de rutinas de atención de periféricos, programación de subrutinas y paso de parámetros, programación de los dispositivos de E/S. **Sistemas operativos** ● Introducción a los sistemas operativos ● El sistema operativo como máquina virtual. ● La gestión de la memoria. ● Dispositivos de entrada/salida. ● Sistema de ficheros. ● La concurrencia y la comunicación. ● El estado actual de los sistemas operativos. ● Introducción al sistema operativo GNU/Linux y a su intérprete de comandos mediante programación *shell script.* ● Llamadas a sistema en lenguaje C. **Redes y aplicaciones de Internet** ● Conceptos básicos de redes de comunicaciones e Internet. ● Aplicaciones y seguridad en Internet. ● Sistemas basados en web. ● Seguridad en la web. ● Arquitecturas de Internet: TCP/IP. **Administración de redes y sistemas** ● Objetivos y responsabilidades de un administrador de sistemas.

	• La comunicación entre diferentes sistemas operativos. • Conceptos y herramientas relacionadas con la administración de redes y sistemas. • Necesidades genéricas de los usuarios. • Integración del software en una organización. • Estrategias para maximizar la seguridad de un sistema informático. • Protección de datos: aspectos legales de los usuarios y los administradores de sistemas. • Planificación del sistema informático de una organización. • Gestión de un departamento de TI. • Planes de contingencia. • Leyes y normativas relacionadas con la seguridad. **Seguridad en redes de computadores** • Seguridad y riesgos en redes informáticas, ataques informáticos y defensa. • *Firewalls* y sistemas de detección de intrusiones. • Seguridad en la capa de comunicación (redes inalámbricas, IPsec, TLS) y en la capa de aplicación (SSH, VPN, correo electrónico seguro). **Computación en la nube** • Introducción a la computación en la nube, modelos IaaS/PaaS/SaaS. • Virtualización (hipervisores). • Arquitecturas y tipos de servicios en la nube. • Proveedores de nube pública y servicios relacionados (por ejemplo, Amazon AWS y Microsoft Azure). • DevOps. **Sistemas empotrados** • Identificación y estudio de sistemas empotrados (dispositivos médicos, sistemas autónomos, etc.). • Microcontroladores y periféricos básicos. • Periféricos avanzados. • Sistemas operativos en tiempo real.

Materia 7. Habilidades empresariales y blandas	
Número de créditos ECTS	18
Tipo	Mixta (12 créditos de formación básica y 6 optativos)
Calendario	Semestres 1, 2 y 5/6
Modalidad	Virtual

Resultados del aprendizaje del grado relacionados con esta materia	• K3. Identificar las características de los diferentes tipos de organizaciones y el papel que desempeñan en ellas las TIC. • K5. Describir los lenguajes de programación, las herramientas y las tecnologías utilizados en el proceso de desarrollo de software, junto con los escenarios en los que son más adecuados. • S6. Analizar los problemas de desarrollo con un nivel de abstracción adecuado para seleccionar la alternativa tecnológica más apropiada (lenguaje de programación, algoritmo, herramienta o componente). • S7. Preparar propuestas para proyectos de desarrollo teniendo en cuenta los recursos, las alternativas disponibles y las condiciones del mercado. • S8. Expresar ideas de forma clara y correcta en inglés, tanto oralmente como por escrito, demostrando dominio de los términos y expresiones específicos del campo de desarrollo de software, tanto en contextos académicos como profesionales. • C1. Colaborar eficazmente en equipos u organizaciones de desarrollo de software. • C2. Planificar y llevar a cabo proyectos de desarrollo de software, presentar los resultados de forma adecuada al público destinatario y defender las conclusiones ante un grupo de expertos. • C3. Adaptarse a las nuevas tecnologías de desarrollo de software y a los cambios en las necesidades del mercado y la sociedad. • C4. Aplicar las TIC en el ámbito académico y profesional. • C5. Desempeñar la práctica académica y profesional de forma honesta, ética, sostenible y socialmente responsable, siendo consciente de los derechos humanos, la perspectiva de género y la diversidad y respetándolos.
Resultados del aprendizaje	Después de cursar esta materia, el estudiantado será capaz de: • Seleccionar y utilizar herramientas y servicios para crear, editar y compartir diferentes tipos de contenidos (documentos, diagramas, hojas de cálculo, presentaciones, vídeos, audio, etc.). • Aplicar herramientas de inteligencia artificial generativa en entornos académicos y profesionales teniendo en cuenta sus ventajas, limitaciones e implicaciones éticas. • Colaborar eficazmente con un equipo de compañeros en un entorno en línea utilizando las herramientas y estrategias más adecuadas. • Gestionar proyectos de desarrollo de software de manera remota. • Comprender los objetivos y las características de los diferentes tipos de documentos, así como los aspectos que hay que tener en cuenta al redactarlos (contexto, objetivo, formato, estructura, vocabulario, sintaxis, puntuación, etc.). • Comunicarse de forma eficaz en entornos académicos y profesionales. • Identificar los diferentes tipos de sistemas de información, sus características definitorias y su función en las organizaciones.
Asignaturas	Todas las asignaturas son de 6 ECTS y se imparten en inglés. • Trabajo en equipo en línea y habilidades académicas (formación básica: computación, semestre 1) • Fundamentos de los sistemas de información (formación básica: comercial, semestre 2) • Habilidades comunicativas (optativa, semestre 5/6)

Contenido	**Trabajo en equipo en línea y habilidades académicas** ● Navegación, comunicación y colaboración en línea. ● Preparación y presentación de documentos a distancia. ● Uso eficaz y ético de las herramientas de IA generativa. ● Herramientas y técnicas para el desarrollo de software a distancia. ● Planificación y gestión de proyectos de TI a distancia. ● Principios de gestión de proyectos desde PMI. **Competencias comunicativas** ● Conceptos básicos de la teoría de la comunicación oral, aplicación de los conceptos en conversaciones comerciales. ● Redacción de textos técnicos y cómo abordar el proceso de forma metódica. ● Presentaciones profesionales: estructura, formulación, exposición. ● Reflexión, análisis y comentarios sobre los puntos mencionados. **Fundamentos de los sistemas de información** ● Qué son los sistemas de información. ● Estrategia organizativa, ventaja competitiva y sistemas de información. ● Ética y privacidad. ● Seguridad de la información. ● Gestión de datos y conocimientos. ● eBusiness y eCommerce. ● Computación social. ● Sistemas de información dentro de la organización. ● Gestión de las relaciones con los clientes y gestión de la cadena de suministro. ● Análisis empresarial. ● Adquisición de sistemas de información y aplicaciones. ● Inteligencia artificial.

Materia 8. Ámbitos de aplicación	
Número de créditos ECTS	18
Tipo	Mixta (6 créditos obligatorios y 12 optativos)
Calendario	Semestres 5 y 6
Modalidad	Virtual
Resultados del aprendizaje del grado relacionados	● K5. Describir los lenguajes de programación, las herramientas y las tecnologías utilizados en el proceso de desarrollo de software, junto con los escenarios en los que son más adecuados. ● S1. Diseñar, desarrollar y probar aplicaciones de software para diferentes plataformas y ámbitos de aplicación empleando las mejores prácticas de ingeniería de software.

con esta materia	• S4. Aplicar técnicas para el procesamiento, el almacenamiento y la gestión de datos, así como el acceso a dichos datos, teniendo en cuenta el rendimiento y la disponibilidad. • S6. Analizar los problemas de desarrollo con un nivel de abstracción adecuado para seleccionar la alternativa tecnológica más apropiada (lenguaje de programación, algoritmo, herramienta o componente). • C3. Adaptarse a las nuevas tecnologías de desarrollo de software y a los cambios en las necesidades del mercado y la sociedad. • C5. Desempeñar la práctica académica y profesional de forma honesta, ética, sostenible y socialmente responsable, siendo consciente de los derechos humanos, la perspectiva de género y la diversidad y respetándolos.
Resultados del aprendizaje	Después de cursar esta materia, el estudiantado será capaz de: • Comprender el contexto específico, los objetivos y los retos de un ámbito de aplicación determinado e identificar los métodos, los algoritmos, las tecnologías y las herramientas más adecuados. • Diseñar, desarrollar, implementar y probar aplicaciones móviles. • Comprender los retos que plantea la creación de sistemas distribuidos y los diferentes algoritmos y sistemas que pueden utilizarse para abordarlos. • Comprender el concepto de *cadena de bloques* y *contrato inteligente*, así como sus posibles aplicaciones y limitaciones. • Utilizar los lenguajes, las bibliotecas y las herramientas existentes para desarrollar aplicaciones de cadena de bloques.
Asignaturas	Todas las asignaturas son de 6 ECTS y se imparten en inglés. • Desarrollo de aplicaciones móviles (obligatoria, semestre 6) • Sistemas distribuidos (optativa, semestre 5/6) • Tecnología *blockchain* y contratos inteligentes (optativa, semestre 5/6)
Contenido	**Desarrollo de aplicaciones móviles** • Evaluación de nuevas técnicas, marcos y bibliotecas para el desarrollo de aplicaciones móviles. • Un concepto moderno del desarrollo de aplicaciones móviles. • La garantía de la calidad de las aplicaciones móviles en la práctica. **Sistemas distribuidos** • Perspectiva algorítmica de los sistemas distribuidos: instantáneas, ondas, detección de interbloqueos, terminación, elección. • Algoritmos distribuidos probabilísticos. • Implementación de algoritmos. • Corrección y complejidad de los algoritmos distribuidos. **Tecnología *blockchain* y contratos inteligentes** • Introducción a las cadenas de bloques (principios de funcionamiento, tipos, protocolos de consenso, hashes y criptografía, carteras, transacciones, ventajas, retos y aplicaciones). • Contratos inteligentes (introducción a los contratos inteligentes, estándares, programación en Solidity, implementación de un contrato inteligente, oráculos).

	● Desarrollo *front-end* (introducción al desarrollo *front-end* de aplicaciones, bibliotecas JavaScript para interactuar con contratos inteligentes, programación en JavaScript). ● Almacenamiento descentralizado y permanente (introducción al problema, IPFS / Pinata, otras soluciones).

Materia 9. Proyectos	
Número de créditos ECTS	24
Tipo	Mixta (12 créditos obligatorios y 12 optativos)
Calendario	Semestres 5 y 6
Modalidad	Virtual
Resultados del aprendizaje del grado relacionados con esta materia	Esta materia participa en la consecución de todos los resultados del aprendizaje identificados para el grado.
Resultados del aprendizaje	Después de cursar esta materia, el estudiantado será capaz de: ● Definir, planificar y ejecutar un proyecto complejo de desarrollo de software sobre un tema concreto. ● Aplicar, adaptar y combinar todos los resultados del aprendizaje adquiridos durante el grado utilizando criterios profesionales, creativos e innovadores. ● Investigar un problema o ámbito específico e identificar y citar las fuentes de documentación e información pertinentes. ● Trabajar de forma autónoma como desarrollador de software estableciendo objetivos, definiendo prioridades y creando un calendario con hitos. ● Mostrar iniciativa y proactividad para introducir innovaciones, identificar problemas, buscar información y proponer soluciones creativas. ● Tener en cuenta los aspectos de calidad a la hora de preparar entregables y artefactos (software, documentos, presentaciones, etc.).

	• Adaptarse a nuevos escenarios y problemas inesperados proponiendo las mejores soluciones posibles dependiendo de la situación. • Comunicar y defender eficazmente las decisiones tomadas y los resultados del proyecto ante un público técnico, tanto de forma oral como escrita.
Asignaturas	Todas las asignaturas son de 6 ECTS y se imparten en inglés. • Trabajo final (obligatorio, semestre 6) • Prácticas (optativas, semestre 5)
Contenido	**Trabajo final** • El estudiantado deberá desarrollar un proyecto completo en el que aplique todos los conocimientos, habilidades y competencias adquiridos durante el grado. **Prácticas** • El contenido de las prácticas dependerá de la organización y el proyecto concretos en los que se realicen.

La correspondencia entre los resultados del aprendizaje del grado y los resultados del aprendizaje de las asignaturas se puede consultar aquí.

3.2. CRÉDITOS

La carga lectiva del grado de Desarrollo y Pruebas de Software se mide mediante créditos del sistema europeo de transferencia de créditos (ECTS).

El grado comprende 180 créditos ECTS en total organizados en asignaturas de 6 créditos ECTS, a excepción de Trabajo final y Prácticas (incluida como asignatura optativa), que son de 12 créditos ECTS para reflejar su mayor alcance.

Los créditos del grado se distribuyen entre los tipos de asignaturas de la siguiente manera:

Tipo de asignaturas	Créditos ECTS

Formación básica	48
Obligatorias	96
Optativas	24
Prácticas obligatorias	0
Trabajo final	12
Total	**180**

3.3. CARGA LECTIVA

3.3.1. Carga lectiva del grado y las asignaturas

El grado de Desarrollo y Pruebas de Software tiene una carga lectiva de **180 créditos** ECTS. En este grado, 1 crédito ECTS corresponde a **25 horas** de carga lectiva. Esta carga lectiva incluye todas las actividades que deben realizarse en la asignatura, como la lectura de recursos didácticos, la instalación y el uso de herramientas de software, la participación en los foros del aula o la resolución de las actividades de EC y EF. El diseño de cada asignatura tiene en cuenta esta carga lectiva a la hora de definir los resultados del aprendizaje, elegir los recursos didácticos o diseñar las actividades de evaluación.

Todas las asignaturas del grado son de **6 créditos ECTS** (150 horas de carga lectiva), a excepción de las asignaturas Prácticas y Trabajo final, que son de **12 créditos ECTS** cada una (300 horas de carga lectiva). Teniendo en cuenta un semestre típico, con 15 semanas de EC y de 2 a 3 semanas de EF, esto corresponde aproximadamente a una carga lectiva de **entre 8 y 9 horas semanales** por asignatura (partiendo de un curso ideal en el que la carga lectiva se distribuye de manera uniforme a lo largo del semestre).

3.3.2. Duración del grado

El grado de Desarrollo y Pruebas de Software está diseñado para ser completado en un período de entre **3 años en el caso de estudiantes a tiempo completo y 6 años en el caso de estudiantes a tiempo parcial**.

Este grado es un programa de educación a distancia que se imparte a través de un campus virtual. La flexibilidad y la asincronía de este modelo de aprendizaje lo hacen especialmente adecuado para estudiantes a tiempo parcial. Según los datos históricos de titulaciones similares de la UOC, la OUNL y la UNED, se espera que el alumnado a tiempo parcial constituya la gran mayoría de su población estudiantil y que el alumnado a tiempo completo sea una minoría.

Por consiguiente, con el fin adaptar el grado al estudiantado a tiempo parcial que pueda tener compromisos familiares o laborales, no hay un número mínimo de asignaturas en las que haya que matricularse por semestre. No obstante, los tutores recomiendan matricularse en **2 o 3 asignaturas por semestre en el caso del estudiantado a tiempo parcial** y en un máximo de **5 asignaturas por semestre en el caso del estudiantado a tiempo completo**. De este modo, se puede alcanzar un equilibrio adecuado entre la disponibilidad, la carga lectiva y el progreso en la titulación. El estudiantado que desee matricularse en más de 6 asignaturas en un semestre deberá solicitarlo formalmente justificando el motivo por el que necesita esa matriculación excepcional y demostrando que sus antecedentes y circunstancias personales contribuyen a sus posibilidades de éxito.

La tabla siguiente resume la carga lectiva ideal y el tiempo necesario para completar el grado para estos dos perfiles de estudiantes:

	Carga lectiva elegida por semestre		**Tiempo necesario para completar el grado**	
Perfil de estudiante	**Asignaturas**	**Créditos ECTS**	**Años**	**Semestres**
Estudiante a	2-3	12-18	6	12

tiempo parcial		(promedio 15)		
Estudiante a tiempo completo	5	30	3	6

Cabe tener en cuenta que la duración del grado puede ser menor para el estudiantado que convalide créditos por experiencia y estudios (véase la sección 4.2, "Reconocimientos").

4. ADMISIÓN Y RECONOCIMIENTOS

4.1. ADMISIÓN

4.1.1. Normativa y procedimientos generales de admisión

El grado interuniversitario se regirá por la normativa académica de la UOC para el acceso a los estudios. El título 1 de dicha normativa está dedicado a los requisitos y las normas de admisión. Se ofrecerá un apoyo específico a las personas con necesidades de accesibilidad (véase el compromiso de accesibilidad).

Por otra parte, la información general sobre el proceso de admisión y matriculación en un grado de la UOC se puede encontrar en la siguiente página web.

4.1.2. Criterios específicos de admisión a este grado

El inglés es la lengua de docencia de este grado. Por lo tanto, para ser admitidas en el grado, las personas candidatas deberán acreditar un nivel de competencia en inglés equivalente, como mínimo, al nivel B2 del Marco común europeo de referencia para las lenguas (MCER).

La competencia en inglés puede acreditarse de diferentes maneras:

- Mediante un certificado oficial que acredite este nivel de competencia lingüística en inglés.
- Habiendo superado los cursos de inglés correspondientes al nivel B2 en titulaciones universitarias o en centros de idiomas de universidades.
- Habiendo completado el bachillerato o una titulación universitaria en el extranjero en inglés.
- Habiendo completado una titulación universitaria centrada en el estudio de lenguas extranjeras, con al menos 30 créditos ECTS dedicados al inglés en los que se hayan ejercitado las

cuatro habilidades lingüísticas (expresión oral, comprensión oral, lectura y expresión escrita).

- Habiendo realizado una prueba oficial de inglés (como LinguaSkill) que evalúe las cuatro habilidades lingüísticas (expresión oral, comprensión oral, lectura y expresión escrita) y presentando un certificado de un nivel equivalente, como mínimo, al B2.

4.2. RECONOCIMIENTOS

4.2.1. Normativa y procedimientos generales de reconocimiento

El título 5 de la normativa académica de la UOC está dedicado al reconocimiento de titulaciones y conocimientos previos. Por otra parte, en el portal se ofrece información general sobre los procedimientos de reconocimiento de créditos.

4.2.2. Criterios específicos de reconocimiento para este grado

Las tablas siguientes contienen los criterios específicos de reconocimiento para estudiantes que acrediten formación profesional (primera tabla), experiencia profesional, títulos universitarios no oficiales (segunda tabla) o títulos extinguidos (última tabla).

Reconocimiento de la formación profesional	Máximo: 45 + 12 ECTS
El reconocimiento de créditos de la formación profesional no superará el 25 % del número total de créditos del título (180 ECTS). Las tablas de reconocimiento serán validadas por la Generalitat de Cataluña. *Además, de acuerdo con la normativa española de formación profesional (art. 130 del Real Decreto 659/2023, solo en español), se puede reconocer una cantidad adicional de hasta 12 créditos al estudiantado que acredite formación profesional especializada directamente relacionada con el grado. Un ejemplo de esta formación profesional especializada es el curso de especialización de Desarrollo de aplicaciones en lenguaje Python. La lista completa de cursos se publicará en las tablas de reconocimiento, que serán validadas por la Generalitat de Cataluña.*	

Reconocimiento de la experiencia profesional **Reconocimiento de títulos universitarios no**	Máximo: 24 ECTS

oficiales

El reconocimiento de créditos no podrá superar el 15 % del número total de créditos del grado (180 ECTS).

Reconocimiento de créditos de titulaciones extinguidas Máximo: 168 ECTS

El estudiantado del grado de Técnicas de Aplicaciones de Software, que queda extinguido por este grado interuniversitario, podrá convalidar asignaturas según la tabla siguiente.

Grado de Técnicas de Aplicaciones de Software (grado anterior)	Grado de Desarrollo y Pruebas de Software (grado nuevo)
Álgebra	Fundamentos de matemáticas I
Lógica	Fundamentos de matemáticas II
Trabajo en equipo en la red	Trabajo en equipo en línea y habilidades académicas
Fundamentos de programación	Fundamentos de programación
Prácticas de programación	Prácticas de programación
Programación web	Programación web
Ingeniería de software	Ingeniería de software
Estructura de computadores	Estructura de computadores
Emprendimiento	Emprendimiento
Programación orientada a objetos	Programación orientada a objetos
Estructuras de datos	Estructuras de datos
Lenguajes y estándares web	Lenguajes y estándares web
Patrones de diseño de software	Patrones de diseño de software
Arquitectura de software	Arquitectura de software
Interacción persona-ordenador	Interacción persona-ordenador
Introducción a las bases de datos	Introducción a las bases de datos
Diseño de bases de datos	Diseño de bases de datos
Sistemas operativos	Sistemas operativos
Redes y aplicaciones de Internet	Redes y aplicaciones de Internet

Administración de redes y sistemas	Administración de redes y sistemas
Computación en la nube	Computación en la nube
Desarrollo de aplicaciones móviles	Desarrollo de aplicaciones móviles
Gestión empresarial y de TI	Fundamentos de los sistemas de información
Competencias comunicativas para profesionales de las TIC	Competencias comunicativas
Sistemas distribuidos	Sistemas distribuidos
Seguridad en redes de computadores	Seguridad en redes de computadores
Comercio electrónico	Comercio electrónico
Fundamentos de computadores	Fundamentos de computadores
Sistemas empotrados	Sistemas empotrados
Prácticas	Prácticas

4.3. MOVILIDAD

La normativa académica de la UOC regula la movilidad mediante el artículo 31 y el capítulo 3 del título III. El estudiantado externo que solicite una plaza en la UOC puede encontrar toda la información pertinente en el portal de esta universidad. Asimismo, el estudiantado del programa puede encontrar información sobre movilidad también en el portal de la UOC. Además, hay guías disponibles para ayudar al estudiantado a tramitar todos los procedimientos relacionados con las becas Erasmus (por ejemplo, la presentación de bienvenida, la guía de grados y másteres y la guía de prácticas).

Este proyecto se enmarca en la iniciativa OpenEU Erasmus+ European University. OpenEU proporciona un vehículo digital para que el estudiantado y el personal confeccionen sus currículos, mejoren su empleabilidad y sus carreras profesionales, y contribuyan a la creación de un panorama de la educación superior más diversificado e inclusivo y de una base más sólida para el desarrollo regional en Europa.

OpenEU establecerá un enfoque común en toda la Alianza para garantizar una movilidad inclusiva, fluida y más sostenible para el estudiantado y el personal de OpenEU mediante distintos formatos de movilidad: virtual, híbrida, combinada y presencial. Se prevé que, en 2028, al menos el 50 % del estudiantado que se gradúe en una universidad OpenEU y al menos el 10 % del personal académico y de gestión a tiempo completo de OpenEU habrán participado como mínimo en un programa de movilidad.

5. APRENDIZAJE, ENSEÑANZA Y EVALUACIÓN

5.1. APRENDIZAJE Y ENSEÑANZA

5.2.1. Asignaturas de formación básica, obligatorias y optativas

La siguiente tabla muestra las asignaturas ordenadas por tipos.

Tipo de asignaturas	Nombre de la asignatura
Formación básica	• Fundamentos de los sistemas de información • Fundamentos de programación • Fundamentos de computadores • Fundamentos de matemáticas I • Fundamentos de matemáticas II • Trabajo en equipo en línea y habilidades académicas • Prácticas de programación • Ingeniería de software
Obligatorias	• Computación en la nube • Estructura de computadores • Estructuras de datos • Diseño de bases de datos • Factores humanos en sistemas informáticos • Introducción a las bases de datos • Desarrollo de aplicaciones móviles • Redes y aplicaciones de Internet • Administración de redes y sistemas • Programación orientada a objetos • Sistemas operativos • Arquitectura de software • Patrones de diseño de software • Pruebas de software • Programación web • Lenguajes y estándares web

Optativas	• Tecnología *blockchain* y contratos inteligentes • Competencias comunicativas • Sistemas distribuidos • Sistemas empotrados • Prácticas • Seguridad en redes de computadores

Actividades de aprendizaje en este grado

Las asignaturas pondrán en práctica diferentes actividades y métodos de aprendizaje a lo largo del grado. A continuación se describen brevemente los más representativos.

1. **Debate:** discusión entre compañeros sobre un tema seleccionado y siguiendo una estructura predeterminada. El objetivo es fomentar las habilidades comunicativas y la capacidad de razonar y argumentar para defender una postura.
2. **Búsqueda de información:** identificar fuentes de información relevantes (publicaciones, informes, documentación, bases de datos), elaborar consultas de búsqueda adecuadas y seleccionar, comprender y procesar la información, a ser posible recopilando información de varias fuentes.
3. **Estudio de casos:** partiendo de la descripción de un escenario complejo y realista en un contexto profesional, se invita a analizar un problema técnico con el fin de proponer una solución que se adapte a las características particulares del escenario estudiado.
4. **Exposición escrita:** creación de un documento textual (un informe, un sitio web, un documento de requisitos, un proyecto o documentación) que describa un problema, un proyecto o una solución técnica. El objetivo es fomentar las habilidades de comunicación y síntesis.
5. **Presentación oral:** exposición de ideas clave sobre un problema, proyecto o solución técnica teniendo en cuenta las limitaciones temporales y aprovechando los medios disponibles (por ejemplo, diapositivas).

6. **Resolución de problemas:** abordar retos utilizando varios conocimientos y habilidades.
7. **Preguntas teóricas:** responder a preguntas sobre hechos, principios y teorías en el campo del desarrollo de software.
8. **Ejercicios y actividades prácticas:** aplicar métodos, herramientas y técnicas conocidas para abordar una situación concreta.
9. **Programación:** desarrollar código informático para un método, aplicación o servicio que satisfaga un conjunto de requisitos concretos. Dependiendo de la tarea, este código se podrá crear desde cero o interactuando con código existente.
10. **Proyecto:** definir, planificar, ejecutar y finalizar un proyecto que aplique un conjunto amplio de habilidades.

Metodologías de aprendizaje del grado

Este grado tendrá en cuenta las siguientes metodologías de aprendizaje:

1. **Instrucción guiada mediante la lectura de documentación científica y técnica:** lectura de una selección cuidada de documentos científicos y técnicos para obtener información sobre un método, herramienta o tecnología concretos. La lectura va seguida de ejemplos y ejercicios para consolidar el aprendizaje.
2. **Aprendizaje mediante la práctica:** entrenamiento de una habilidad práctica mediante la resolución de tareas y situaciones realistas con el fin de aprender de los éxitos y los errores.
3. **Aprendizaje basado en casos:** la actividad de aprendizaje se centra en un estudio de caso similar a una situación real, lo que aporta motivación y elementos que deben tenerse en cuenta.
4. **Aprendizaje basado en problemas:** las actividades de aprendizaje se centran en un problema que debe ser resuelto. Este problema motiva el aprendizaje de las habilidades y competencias necesarias para resolverlo.

5. **Aprendizaje basado en proyectos:** las actividades de aprendizaje se centran en la planificación y ejecución de un proyecto complejo con objetivos concretos.
6. **Aprendizaje colaborativo:** resolución de actividades complejas como un esfuerzo de equipo que requiere la definición de objetivos, calendarios de trabajo, reparto de tareas, comunicación y coordinación.

Véase el anexo A.2.1., "Actividades, metodologías y métodos de evaluación" para la correspondencia entre actividades, metodologías y métodos de evaluación.

5.1.2. Prácticas obligatorias

El grado no incluye prácticas obligatorias. Se ofrecerá un apoyo específico al estudiantado que curse la asignatura optativa de prácticas. En particular, la universidad coordinadora prestará asistencia en la búsqueda de prácticas (por ejemplo, ayudando a definir el acuerdo de prácticas entre la empresa y cada estudiante) y en la supervisión (por ejemplo, siguiendo el proceso de aprendizaje y asegurándose de que las prácticas aborden las competencias requeridas por la asignatura). En el anexo se ofrece más información sobre la asignatura optativa Prácticas, de 12 créditos ECTS.

5.1.3. Trabajo final

El trabajo final consiste en la definición, planificación, ejecución y defensa de un proyecto de desarrollo de software.

Los trabajos finales se organizan en torno a **áreas temáticas** dedicadas a una tecnología o un ámbito de aplicación concretos. En cada área, un equipo de profesores y profesoras expertos actúan como **supervisores del proyecto** y se encargan de hacer un seguimiento de cada estudiante y de apoyarlo durante el semestre. Además, dependiendo del área, el estudiantado puede proponer un tema para su trabajo final (que deberá ser validado por la persona que supervise el proyecto) o recibir una lista cerrada de temas entre los que elegir. En ambos casos, el trabajo deberá estar orientado a

la práctica, aplicar los resultados del aprendizaje adquiridos durante el grado y tener un alcance que se corresponda con la carga lectiva.

La primera actividad del trabajo final es la elaboración de un **plan** en el que se describen objetivos, tareas, recursos, calendario y riesgos. Las actividades de evaluación segunda y tercera se corresponden con la **ejecución del trabajo**, en la que el estudiantado lleva a cabo y supervisa el plan propuesto. Durante este tiempo, el tutor o tutora del trabajo final proporciona orientación, apoyo y comentarios personalizados. El trabajo concluye con una **presentación y defensa** ante un tribunal de expertos, que se llevará a cabo de forma síncrona a través de una videoconferencia pública.

Metodologías de aprendizaje del trabajo final

- Aprendizaje mediante la práctica
- Aprendizaje basado en proyectos

Actividades de aprendizaje en el trabajo final

- Debate
- Búsqueda de información
- Exposición escrita
- Presentación oral
- Trabajo

(Estas actividades se han descrito en la sección 5.2.1)

5.2. EVALUACIÓN DEL ESTUDIANTADO

5.2.1. Evaluación de las asignaturas de formación básica, obligatorias y optativas

La siguiente tabla contiene los sistemas de evaluación que se utilizarán en este grado.

	Sistema de evaluación
1	Pruebas de evaluación continua
2	Actividades prácticas
3	Trabajo final

4	Pruebas de síntesis
5	Exámenes

El **modelo de evaluación** se centra en la EC y puede incluir una EF. La EC consiste en la realización y superación de las actividades/pruebas establecidas en el plan docente, mediante las que se garantiza progresivamente el logro de los resultados del aprendizaje del grado y los definidos para cada asignatura. En las asignaturas con EF, esta se realiza al finalizar el semestre. Hay dos tipos de pruebas finales: la PS, destinada a evaluar al estudiantado que ha superado la EC, con una duración máxima de 1 hora; y el EX, que evalúa el logro de los resultados del aprendizaje de la asignatura en su totalidad y tiene una duración máxima de 2 horas.

El modelo de evaluación de cada asignatura tiene en cuenta **aspectos pedagógicos** y, siempre que es posible, aprovecha métodos innovadores para facilitar y animar al estudiantado a alcanzar los resultados del aprendizaje de la asignatura. También se tienen en cuenta aspectos técnicos para garantizar un proceso de evaluación eficaz y riguroso.

Para garantizar unos criterios de evaluación comunes no solo en las asignaturas, sino también en todas las instituciones que participan en el grado, el programa utilizará rúbricas estandarizadas siempre que sea posible y una vez que hayan sido validadas por el profesorado. Además, todas las asignaturas se someterán a un sistema de calificación común, de acuerdo con el enfoque proporcionado en el marco del programa. Estas medidas tienen por objeto ofrecer un sistema de calificación unificado en el programa, independientemente de la institución que imparta la asignatura.

Para verificar la identidad y la autoría de las tareas presentadas por el estudiantado, la UOC aplica sistemas de reconocimiento de identidad. A tal fin, se podrá exigir al estudiantado que se identifique mediante un documento de identidad y se permitirá el uso de micrófonos, cámaras u otras herramientas para la evaluación. Para

detectar y prevenir el plagio, se utilizarán herramientas y estrategias específicas.

El artículo 113 de la normativa académica de la UOC define las situaciones que se consideran conductas irregulares, así como los procedimientos disciplinarios académicos que pueden derivarse de dichas conductas.

La UOC revisa sistemáticamente la eficacia de los mecanismos que garantizan el rigor de la evaluación y los resultados académicos con el fin de mejorar el proceso.

5.2.2. Evaluación de las prácticas obligatorias

El grado no incluye prácticas obligatorias. En el anexo A.2.2 se ofrece más información sobre la asignatura optativa Prácticas, de 12 créditos ECTS.

5.2.3. Evaluación del trabajo final

Los trabajos finales se evalúan mediante una defensa pública y síncrona ante un tribunal examinador. El título IV de la normativa académica de la UOC establece las normas que rigen los trabajos finales.

El tribunal examinador está formado por tres miembros:

- El profesor o profesora del curso que ha tutorizado al estudiante durante el semestre.
- El profesor o profesora coordinador del proyecto final.
- Otro experto, que puede ser docente de una de las instituciones o un miembro del personal docente asociado.

Durante el semestre, el plan docente define varios hitos en los que el estudiantado debe presentar entregables parciales a su supervisor del trabajo final. Estos hitos se utilizan para definir los objetivos y el calendario del trabajo final, comprobar su progreso y proporcionar primeros comentarios.

La entrega final incluye varios entregables:

- Un **documento** que describa los objetivos, la planificación, la ejecución y los resultados del trabajo final.
- Una **presentación** del trabajo en formato de vídeo que resuma los aspectos más relevantes del mismo.
- En la mayoría de los casos, también habrá un **producto** (una aplicación, una herramienta, un servicio) desarrollado como parte del trabajo final.

La defensa del trabajo final tiene lugar después de la entrega final, una vez que el tribunal examinador ha revisado los entregables. En una sesión pública síncrona, el tribunal examinador formula preguntas a cada estudiante sobre los objetivos, el desarrollo y los resultados del proyecto. Las respuestas del estudiantado pueden dar lugar a preguntas y aclaraciones adicionales.

La nota final tendrá en cuenta los siguientes elementos:

1. Rendimiento continuado durante el semestre
2. Documento del trabajo final
3. Presentación
4. Producto (si procede)
5. Respuestas durante la defensa

El supervisor del proyecto final evalúa el punto 1 y el tribunal evaluador evalúa los puntos del 2 al 5. Teniendo en cuenta esta información, el tribunal evaluador elabora un informe de evaluación que resume las conclusiones del tribunal y la calificación propuesta.

6. ACOMPAÑAMIENTO AL ESTUDIANTADO

El acompañamiento al estudiantado se llevará a cabo a través de la universidad coordinadora, lo que permitirá ofrecer un servicio de apoyo centralizado y coherente. Sin embargo, las otras universidades participantes también intervendrán en este proceso de apoyo siempre que el estudiantado necesite asistencia específica relacionada con las asignaturas que dichas universidades ofrecen. A través de los órganos de coordinación establecidos, se mantendrá un contacto regular para garantizar que las instituciones asociadas contribuyan en función de las necesidades de cada caso, lo que permitirá un seguimiento eficiente y una responsabilidad compartida en el apoyo al estudiantado de todo el programa interuniversitario.

En cuanto al acompañamiento ofrecido, la UOC pone a disposición de sus estudiantes **canales de apoyo en línea**, que se explican en la **Guía del estudiantado**. Nuestro objetivo es dar respuestas personalizadas a las dudas del estudiantado **de forma rápida, eficaz y con los más altos niveles de calidad**. La UOC dispone de varios **canales de apoyo en línea** que el estudiantado puede utilizar para enviar sus consultas cualquier día de la semana, **desde cualquier lugar y en cualquier momento**. El estudiantado puede encontrar apoyo a través del Campus Virtual (el Servicio de Ayuda), las cuentas de redes sociales (el Twitter de la UOC) y otros medios. Además de estos canales, ocasionalmente ofrecemos otras vías para ayudar al estudiantado a resolver rápidamente cualquier duda o

incidencia que pueda surgir, por ejemplo, al realizar los exámenes finales o defender los trabajos finales en línea.

7. RECURSOS

7.1. PERSONAL

Como se puede observar en el anexo **A.1.10, "CV de los miembros del personal académico"**, el personal docente del grado cuenta con un total de 26 miembros. El 96 % de este personal docente posee un título de doctorado. Entre las contribuciones docentes más significativas destacan la amplia experiencia en educación tanto en línea como presencial, así como los enfoques innovadores reconocidos en entornos institucionales y relacionados con los temas desarrollados en el programa. En cuanto a la producción científica, los miembros del profesorado participan activamente en proyectos de investigación y publican regularmente en revistas indexadas, y varios de ellos cuentan con prestigiosas acreditaciones y premios. Esta combinación de excelencia docente y rigor investigador pone de relieve el elevado perfil del equipo académico que respalda el programa.

La universidad coordinadora ofrecerá sesiones de formación especiales para presentar y describir las instalaciones y los métodos de evaluación utilizados durante la licenciatura (véase la sección siguiente y la sección 5.2, respectivamente) a las otras universidades que participan en el programa. Estas sesiones se centrarán principalmente en el enfoque de la enseñanza y el aprendizaje, incluido el sistema de gestión del aprendizaje, con el fin de garantizar que el personal docente sea conocedor de los aspectos metodológicos y técnicos de sus asignaturas y que todas ellas estén debidamente coordinadas.

7.2. INSTALACIONES

La UOC ofrece una formación a distancia y utiliza las TIC a través de su Campus Virtual para proporcionar espacios, herramientas y recursos que permitan el buen desarrollo de la actividad formativa,

así como el acceso a la información y los trámites. En el Campus, el aula virtual permite acceder a las asignaturas matriculadas y consultar el plan de estudios correspondiente, los recursos, las actividades y el registro de resultados de evaluación. Además, facilita la relación entre el personal docente y el estudiantado. Cuenta con dos espacios básicos de comunicación: los anuncios del profesorado y los foros de debate.

La tipología de las aulas puede aplicarse a diversos fines:

- Asignaturas estándar: la actividad docente sigue un plan de aprendizaje común. La atención se lleva a cabo a través de buzones personales para cada estudiante, buzones de grupo y la dinamización del profesorado de la asignatura. La cifra de estudiantes por aula es de un máximo de 70.
- Asignaturas con dedicación especial: prevalecen los elementos de individualización para que cada estudiante o pequeños grupos de estudiantes sigan un itinerario de aprendizaje diferenciado. La cifra máxima propuesta de estudiantes por aula es de 50. En las asignaturas de Trabajo final se recomienda no superar los 40 estudiantes por aula.
- El trabajo final requiere una actividad de seguimiento y tutoría individualizada y personalizada. La cifra de estudiantes por aula en estas asignaturas es de 10 a 15 como máximo, siendo habitualmente inferior a 10.

Para más información, véase la Guía del estudiantado, que contiene información sobre servicios como:

- El modelo educativo y cómo se estudia en la UOC
- Biblioteca
- Servicios de apoyo en línea y ubicaciones de la UOC
- Acompañamiento y orientación de graduados
- Servicios para estudiantes con discapacidad
- Tutoría personalizada

8. TRANSPARENCIA Y DOCUMENTACIÓN

Las tres instituciones publicarán información sobre el programa interuniversitario en sus respectivos sitios web. Esta información incluirá: presentación y descripción del programa, plan de estudios, reconocimiento de titulaciones y experiencia previa, profesorado y personal administrativo, un estudio sobre el mercado laboral, requisitos de acceso, matrícula y precios, y formulario de contacto. Además, se incluirá información específica sobre las particularidades del programa, como las universidades participantes y las principales características del modelo educativo y de evaluación que se seguirá. A modo de ejemplo, véase la página Study at the UOC, que incluye información relevante sobre los diferentes programas ofrecidos en inglés por la UOC.

Además, como ya se ha indicado, en cada curso académico se publica una Guía del estudiantado con información destinada a explicar al estudiantado nuevo cómo matricularse y estudiar en la UOC, las principales características del funcionamiento y el modelo educativo de la universidad, así como los servicios que la UOC les ofrece antes, durante y después de finalizar sus estudios.

La información sobre la calidad de los programas también se ofrece en la sección *Calidad de las titulaciones* de la web de la universidad. Del mismo modo, existe información pública sobre los Estudios de Informática, Multimedia y Telecomunicación.

9. GARANTÍA DE LA CALIDAD

El sistema de garantía de la calidad aplicado para garantizar la calidad de este programa será el de la UOC. En particular, el programa implementará el sistema de garantía interna de la calidad proporcionado por la UOC, como universidad coordinadora, para garantizar la calidad del programa.

El sistema de garantía interna de la calidad sigue el manual correspondiente, elaborado en 2009 en el marco del programa AUDIT. Este manual, certificado por AQU Catalunya en 2009, actualizado en 2020 y certificado finalmente por AQU Catalunya en 2021, permite integrar todas las actividades destinadas a garantizar la calidad de las enseñanzas, ampliar los mecanismos de participación de los grupos de interés e incorporar la sistemática de revisión y mejora continua. El sistema de garantía interna de la calidad se ha diseñado siguiendo los estándares internacionales de calidad. Así, apuesta por un enfoque en los procesos en sentido amplio para dar cabida e integrar cualquier proceso que la UOC pueda identificar y desarrollar en el futuro.

Para más información sobre el sistema de garantía interna de la calidad de la UOC, consulte la sección *Sistema de garantía interna de la calidad*.

A. Anexos

A.1. ANEXOS OBLIGATORIOS

A.1.1. Documentos acreditativos de la personalidad jurídica de las instituciones asociadas

1) Documentos acreditativos de la personalidad jurídica de las instituciones asociadas:

UOC: Documento 1 y Documento 2

OUNL: Documento 1 y Documento 2

UNED: Documento 1 y Documento 2

A.1.2. Convenio de colaboración

2) Convenio de colaboración

3) Adenda sobre movilidad

4) Acuerdo específico sobre movilidad

A.1.3. Documentos acreditativos de la base jurídica de cada socio

5) Documentos acreditativos de la base jurídica de cada socio:

- *a) Participación en el programa interuniversitario*
- *b) Derechos de expedición del título de grado (interuniversitario) (si procede)*

La información se ha facilitado en la sección A.1.2.

A.1.4. Lista de resultados del aprendizaje previstos

6) Lista de resultados del aprendizaje previstos, incluyendo:

- *a) Matriz de alineación con el Marco de cualificaciones del espacio europeo de educación superior (FQ-EHEA)*
  - *i) Los resultados del aprendizaje del grado se pueden consultar aquí.*

ii) Los resultados del aprendizaje del grado desarrollados por cada asignatura se pueden consultar aquí.

iii) La correspondencia entre los resultados del aprendizaje del grado y los resultados del aprendizaje de las asignaturas se puede consultar aquí.

b) Matriz de alineación con el marco nacional de cualificaciones aplicable. La correspondencia detallada entre los resultados del aprendizaje del grado y los marcos de cualificaciones español y neerlandés se puede consultar aquí. Ambos marcos están alineados con el marco de cualificaciones europeo.

	Marco de cualificaciones español *MECU/MECES*	***Marco de cualificaciones neerlandés*** *NLQF*	***Marco europeo de cualificaciones*** *MEC*
Grado 180 ECTS	*Nivel MECU 6 (Marco español de cualificaciones)* *Nivel MECES 2* *(Marco español de cualificaciones para la educación superior)*	*Nivel 6*	*Nivel 6*

A.1.5. Programas de asignaturas de todos los socios

7) Programas de asignaturas de todos los socios

Programas

A.1.6. Estructura del plan de estudios

8) Estructura del plan de estudios

Encontrará los detalles en la sección 3 (programa de estudios)

A.1.7. Documentos oficiales que indican los requisitos de admisión y los procedimientos de selección

9) Documentos oficiales que indican los requisitos de admisión y los procedimientos de selección

Normativa académica de la UOC, capítulo II

A.1.8. Documentos oficiales que describen el procedimiento de reconocimiento de las cualificaciones

10) Documentos oficiales que describen el procedimiento de reconocimiento de las cualificaciones

Normativa académica de la UOC, título V

A.1.9. Normativa de evaluación del estudiantado

11) Normativa de evaluación del estudiantado

Normativa académica de la UOC, título VI

A.1.10. CV de los miembros del personal académico (todos los socios)

12) CV de los miembros del personal académico (todos los socios)

A.1.11. Documentos pertinentes que constituyen el sistema de garantía interna de la calidad

13) Documentos pertinentes que constituyen el sistema de garantía interna de la calidad

Sistema de garantía interna de la calidad
Sistema de EUC Informes (AQU Catalunya)
RUCT

A.1.12. Suplemento al título (modelo)

14) Suplemento al título (modelo)

Modelo

A.2. ANEXOS ADICIONALES

A.2.1. ACTIVIDADES, METODOLOGÍAS Y MÉTODOS DE EVALUACIÓN

La correspondencia entre las actividades, las metodologías y los métodos de evaluación se muestra en esta tabla.

A.2.2. PRÁCTICAS

El grado incluye la opción de realizar una asignatura de prácticas durante los estudios. Puede encontrar más información sobre los aspectos metodológicos y una lista de las principales empresas con convenios de colaboración educativa con la universidad coordinadora aquí.

Además, en el contexto de este grado internacional, se explorarán colaboraciones con una variedad más amplia de empresas e instituciones con convenios de colaboración con las universidades participantes en el grado interuniversitario.

ImpEA

Facilitating implementation of the European Approach for Quality Assurance of Joint Programmes

Erasmus+

Líder del proyecto

Socios del proyecto

Deusto

EUSKAL UNIBERTSITATE SISTEMAREN KALITATE AGENTZIA
AGENCIA DE CALIDAD DEL SISTEMA UNIVERSITARIO VASCO
UNIBASQ

Akkreditiert durch AQAS www.aqas.de

ADAM MICKIEWICZ UNIVERSITY POZNAŃ
UAM

CARL VON OSSIETZKY universität OLDENBURG

ENQA
EUROPEAN ASSOCIATION FOR QUALITY ASSURANCE IN HIGHER EDUCATION

WARSAW SCHOOL OF TOURISM AND HOSPITALITY MANAGEMENT

eca european consortium for accreditation