



UNIVERSIDAD
POLITÉCNICA
DE MADRID

PROCESO DE
COORDINACIÓN DE LAS
ENSEÑANZAS PR/CL/001



E.T.S. de Ingeniería y Sist. de
Telecom.

ANX-PR/CL/001-01

GUÍA DE APRENDIZAJE

ASIGNATURA

595040503 - Programación Multiparadigma Para Sistemas De Datos

PLAN DE ESTUDIOS

59SO - Grado En Ingeniería De Sonido E Imagen

CURSO ACADÉMICO Y SEMESTRE

2025/26 - Primer semestre

Índice

Guía de Aprendizaje

1. Datos descriptivos.....	1
2. Profesorado.....	1
3. Competencias y resultados de aprendizaje.....	2
4. Descripción de la asignatura y temario.....	3
5. Cronograma.....	6
6. Actividades y criterios de evaluación.....	9
7. Recursos didácticos.....	12
8. Otra información.....	13

1. Datos descriptivos

1.1. Datos de la asignatura

Nombre de la asignatura	595040503 - Programación Multiparadigma para Sistemas de Datos
No de créditos	6 ECTS
Carácter	Optativa
Curso	Cuarto curso
Semestre	Séptimo semestre
Período de impartición	Septiembre-Enero
Idioma de impartición	Castellano
Titulación	59SO - Grado en Ingeniería de Sonido e Imagen
Centro responsable de la titulación	59 - E.T.S. De Ingeniería Y Sist. De Telecom.
Curso académico	2025-26

2. Profesorado

2.1. Profesorado implicado en la docencia

Nombre	Despacho	Correo electrónico	Horario de tutorías *
David Jesus Meltzer Camino	A4403	david.meltzer@upm.es	Sin horario. Publicadas en la Web de la ETSIST. Acordar la tutoría por correo electrónico.

Carlos Carrillo Sanchez (Coordinador/a)	4401,4421	carlos.carrillo@upm.es	Sin horario. Publicadas en la Web de la ETSIST. Acordar la tutoría por correo electrónico.
--	-----------	------------------------	---

* Las horas de tutoría son orientativas y pueden sufrir modificaciones. Se deberá confirmar los horarios de tutorías con el profesorado.

3. Competencias y resultados de aprendizaje

3.1. Competencias

CE B2 - Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.

CE TEL07 - Conocimiento y utilización de los fundamentos de la programación en redes, sistemas y servicios de telecomunicación.

CE TL07 - Capacidad de programación de servicios y aplicaciones telemáticas, en red y distribuidas.

CG 13 - Habilidades de aprendizaje con un alto grado de autonomía.

3.2. Resultados del aprendizaje

RA1255 - Saber utilizar las herramientas estándar de desarrollo de aplicaciones para un sistema operativo de propósito general

RA1014 - Familiarizarse con el manejo básico de herramientas para desarrollar programas: editor, compilador, enlazador y depurador, dentro de un entorno integrado de desarrollo.

RA1083 - Ser capaz de programar, en un lenguaje de alto nivel, aplicaciones de complejidad media de acuerdo a las reglas de la programación estructurada

RA1160 - Ser capaz de programar, en un lenguaje de alto nivel, aplicaciones de complejidad media de acuerdo a las reglas de la programación estructurada.

RA1208 - Saber programar en un lenguaje multiplataforma de código abierto

RA1209 - Saber programar en un lenguaje de programación de alto nivel

RA1210 - Saber programar en un lenguaje de programación interpretado

RA1248 - Ser capaz de programar, en un lenguaje de alto nivel, aplicaciones de complejidad media de acuerdo a las reglas de la programación estructurada

4. Descripción de la asignatura y temario

4.1. Descripción de la asignatura

La programación es una herramienta básica para cualquier graduado en ingeniería. En concreto, en Ingeniería y Sistemas de Datos, los graduados deben ser capaces de superar retos tecnológicos en el contexto de sistemas que generan, analizan e intercambian cantidades masivas de datos.

La asignatura de Programación representa el primer contacto que tienen los estudiantes con esta disciplina que desarrollarán a lo largo de la titulación. En ella, se hace una introducción a la programación multiparadigma, y a los elementos del lenguaje de programación Python en particular. Python ha evolucionado hasta convertirse en uno de los lenguajes de programación más utilizados en ciencia e ingeniería de datos, por su flexibilidad, simplicidad y adecuación al procesamiento de grandes cantidades de datos. A la vez, se capacita al alumno en el uso de herramientas de desarrollo y depuración así como en el diseño y desarrollo incremental de proyectos.

La asignatura tiene 6 créditos ECTS oficiales. Esto se traduce en 162 horas de trabajo total, concentradas en unas 14 semanas (1 ECTS = 27 horas de trabajo, divididas en 10 horas de clase y 17 de trabajo individual del alumno). Este trabajo incluye la asistencia activa a las clases presenciales de grupo y de laboratorio, el estudio, realización de ejercicios y pruebas de autoevaluación, resolución de actividades y proyectos de laboratorio, y realización de las pruebas de evaluación progresiva. La superación de la asignatura se realizará a través de esta evaluación progresiva.

La asignatura se imparte en modo b-learning, es decir, combinando la enseñanza presencial y la no presencial, para lo cual se utilizará el entorno virtual de aprendizaje Moodle.

4.2. Temario de la asignatura

1. Introducción a la programación
 - 1.1. Introducción a la asignatura
 - 1.2. ¿Qué es la programación y para qué sirve? Paradigmas. Definiciones y terminología
 - 1.3. Introducción al lenguaje Python: Visión global y herramientas
 - 1.4. La vida de un programa
2. Variables, Objetos, tipos y expresiones
 - 2.1. Variables y objetos. Tipado dinámico
 - 2.2. Objetos enteros y strings. Operadores
 - 2.3. Legibilidad del código
 - 2.4. Expresiones y sentencias
3. Estructuras de datos I. Listas y tuplas
 - 3.1. Objetos mutables e inmutables
 - 3.2. Definición y uso de listas
 - 3.3. Definición y uso de tuplas
 - 3.4. Recorrido de listas y tuplas con bucles for
4. Estructuras de control
 - 4.1. Booleans y operadores lógicos
 - 4.2. Expresión condicional. Sentencias if/elif/else
 - 4.3. Bucles for y while
 - 4.4. Interacción con el usuario usando bucles while
5. Funciones
 - 5.1. Definición y uso de funciones
 - 5.2. Definición y uso de argumentos
 - 5.3. Resultado de una función
 - 5.4. Módulos
6. Estructuras de datos II. Diccionarios
 - 6.1. Definición y uso de diccionarios

6.2. Recorrido de diccionarios

6.3. Diccionarios anidados

7. Clases

7.1. Definición de clases y uso de objetos

7.2. Herencia y polimorfismo

7.3. Interacción entre clases

8. Ficheros

8.1. Lectura y escritura en disco

8.2. Persistencia de objetos: pickle

8.3. Ficheros de texto estructurados: cvs y json

8.4. Excepciones

9. Ingeniería de software. Verificación avanzada de código

9.1. Pruebas unitarias: unittest

9.2. Depuración avanzada de código

9.3. Gestión de proyectos y paquetes

9.4. Interactive computing con IPython

5. Cronograma

5.1. Cronograma de la asignatura *

Sem	Actividad tipo 1	Actividad tipo 2	Tele-enseñanza	Actividades de evaluación
1	Tema 1 Duración: 02:00 LM: Actividad del tipo Lección Magistral Tema 2 Duración: 01:00 LM: Actividad del tipo Lección Magistral Tema 2 Duración: 01:00 PR: Actividad del tipo Clase de Problemas			
2	Tema 3 Duración: 01:00 LM: Actividad del tipo Lección Magistral Tema 3 Duración: 03:00 PR: Actividad del tipo Clase de Problemas			
3	Tema 4 Duración: 01:00 LM: Actividad del tipo Lección Magistral Tema 4 Duración: 03:00 PR: Actividad del tipo Clase de Problemas			
4	Tema 5 Duración: 01:00 LM: Actividad del tipo Lección Magistral Tema 5 Duración: 03:00 PR: Actividad del tipo Clase de Problemas			
5	Tema 6 Duración: 01:00 LM: Actividad del tipo Lección Magistral Tema 6 Duración: 01:00 PR: Actividad del tipo Clase de Problemas	Proyecto Fase I Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
6	Tema 7 Duración: 01:00 LM: Actividad del tipo Lección Magistral Tema 7 Duración: 01:00 PR: Actividad del tipo Clase de Problemas	Proyecto Fase I Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		

7	<p>Tema 8 Duración: 01:00 LM: Actividad del tipo Lección Magistral</p> <p>Tema 8 Duración: 01:00 PR: Actividad del tipo Clase de Problemas</p>	<p>Proyecto Fase II Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio</p>		
8	<p>Tema 9 Duración: 01:00 LM: Actividad del tipo Lección Magistral</p> <p>Tema 9 Duración: 03:00 PR: Actividad del tipo Clase de Problemas</p>			<p>Examen 1er parcial EX: Técnica del tipo Examen Escrito Evaluación Progresiva Presencial Duración: 01:00</p>
9	<p>Tema 9 Duración: 01:00 LM: Actividad del tipo Lección Magistral</p> <p>Tema 9 Duración: 03:00 PR: Actividad del tipo Clase de Problemas</p>			
10	<p>Tema 9 Duración: 01:00 LM: Actividad del tipo Lección Magistral</p> <p>Tema 9 Duración: 03:00 PR: Actividad del tipo Clase de Problemas</p>			
11	<p>Tema 9 Duración: 01:00 LM: Actividad del tipo Lección Magistral</p> <p>Tema 9 Duración: 03:00 PR: Actividad del tipo Clase de Problemas</p>			
12	<p>Tema 9 Duración: 01:00 LM: Actividad del tipo Lección Magistral</p> <p>Tema 9 Duración: 03:00 PR: Actividad del tipo Clase de Problemas</p>			
13	<p>Tema 9 Duración: 01:00 LM: Actividad del tipo Lección Magistral</p> <p>Tema 9 Duración: 03:00 PR: Actividad del tipo Clase de Problemas</p>			
14				
15				
16				
17				<p>Examen 2º parcial EX: Técnica del tipo Examen Escrito Evaluación Progresiva Presencial Duración: 02:00</p> <p>Examen final EX: Técnica del tipo Examen Escrito Evaluación Global</p>

				Presencial Duración: 03:00 Prueba de laboratorio EP: Técnica del tipo Examen de Prácticas Evaluación Progresiva Presencial Duración: 02:00
--	--	--	--	---

Para el cálculo de los valores totales, se estima que por cada crédito ECTS el alumno dedicará dependiendo del plan de estudios, entre 26 y 27 horas de trabajo presencial y no presencial.

6. Actividades y criterios de evaluación

6.1. Actividades de evaluación de la asignatura

6.1.1. Evaluación (progresiva)

Sem.	Descripción	Modalidad	Tipo	Duración	Peso en la nota	Nota mínima	Competencias evaluadas
8	Examen 1er parcial	EX: Técnica del tipo Examen Escrito	Presencial	01:00	20%	/ 10	CG 13 CE TEL07 CE TL07 CE B2
17	Examen 2º parcial	EX: Técnica del tipo Examen Escrito	Presencial	02:00	60%	/ 10	CE TL07 CE B2 CG 13 CE TEL07
17	Prueba de laboratorio	EP: Técnica del tipo Examen de Prácticas	Presencial	02:00	20%	/ 10	CG 13 CE TEL07 CE TL07 CE B2

6.1.2. Prueba evaluación global

Sem	Descripción	Modalidad	Tipo	Duración	Peso en la nota	Nota mínima	Competencias evaluadas
17	Examen final	EX: Técnica del tipo Examen Escrito	Presencial	03:00	100%	/ 10	CG 13 CE TEL07 CE TL07 CE B2

6.1.3. Evaluación convocatoria extraordinaria

Descripción	Modalidad	Tipo	Duración	Peso en la nota	Nota mínima	Competencias evaluadas
Examen final	EX: Técnica del tipo Examen Escrito	Presencial	02:00	100%	/ 10	CE TL07 CE B2 CG 13 CE TEL07

6.2. Criterios de evaluación

La metodología utilizada es **evaluación progresiva**.

Durante el curso, el alumno deberá trabajar de forma continuada durante todo el cuatrimestre, asistiendo y participando en las clases teóricas y de laboratorio.

A lo largo del curso, se llevarán a cabo pequeñas actividades interactivas mediante herramientas como Kahoot o Wooclap, con el objetivo de fomentar la participación activa y el aprendizaje continuo.

La implicación del estudiante en estas dinámicas será valorada muy positivamente durante la evaluación progresiva.

Igualmente, deberá entregar en fecha todos los trabajos solicitados en los enunciados de los proyectos de laboratorio. Se realizarán tres pruebas de evaluación:

- Semana 8 Examen escrito. Primer parcial (20%). La fecha de realización de esta prueba estará indicada en el plan docente anual de la ETSIST
- Semana 17 Examen escrito Prueba de laboratorio (20%) La fecha de realización de esta prueba estará indicada en el plan docente anual de la ETSIST
- Semana 17 Examen escrito Segundo parcial (60%) (2 horas).

Sobre la calificación

- El alumno debe obtener al menos un 5.0 en la media ponderada de las calificaciones de las pruebas.
- El alumno se considera PRESENTADO cuando se presenta a alguna de las pruebas que componen la

evaluación progresiva.

- La detección de copia en cualquier prueba de evaluación supondrá un 0 en dicha prueba.
- Está prohibido el uso de cualquier dispositivo de comunicación en las pruebas o exámenes de evaluación, a no ser que el profesor de la clase o el encargado del examen dé una autorización explícita. Si se detecta el incumplimiento de esta norma, el alumno será inmediatamente expulsado del aula y su calificación en el examen será un 0.

Examen evaluación global

La convocatoria de evaluación global tendrá lugar en el mes de junio o julio, a propuesta de la Escuela. Podrán optar a ella los alumnos que no hayan aprobado la asignatura.

Examen extraordinario

La convocatoria extraordinaria tendrá lugar en el mes de junio o julio, a propuesta de la Escuela. Podrán optar a ella los alumnos que no hayan aprobado la asignatura.

Información sobre actuaciones en caso de copia o plagio:

Ante la comprobación fehaciente de copia en una prueba de evaluación, ésta se calificará con la puntuación de cero al estudiante o estudiantes implicados. Si la comprobación se produce durante el desarrollo de la prueba, ésta se podrá interrumpir inmediatamente para el estudiante o estudiantes implicados. El Tribunal de la asignatura o el Director del Departamento podrán elevar al Rector y al Comité de Ética de la UPM los hechos para que puedan tomarse, en su caso, las medidas disciplinarias correspondientes. (A.12)

Los derechos y deberes de los estudiantes universitarios están desarrollados en el Estatuto del Estudiante Universitario (RD 1791/2010 de 30 de diciembre) y en el artículo 13 del referido estatuto, en el punto d) especifica que es deber del estudiante universitario abstenerse de la utilización o cooperación en procedimientos fraudulentos en las pruebas de evaluación, en los trabajos que se realicen o en documentos oficiales de la universidad.

7. Recursos didácticos

7.1. Recursos didácticos de la asignatura

Nombre	Tipo	Observaciones
Python Crash Course	Bibliografía	Autor: Eric Matthes Python Crash Course, 2nd Edition. O'Reilly, 2019. ISBN: 9781593279288 Disponible online: HTTPS://WWW.OREILLY.COM/LIBRARY/VIEW/PYTHON-CRASH-COURSE/9781492071266/ (accesible desde la red de la UPM o usando la VPN-UPM).
Moodle	Recursos web	En la plataforma Moodle el alumno encontrará ejercicios propuestos y resueltos, enlaces interesantes y documentación de la asignatura con los contenidos específicos de la misma.
Learning Python, 5th Edition	Bibliografía	Autor: Mark Lutz Learning Python, 5th Edition O'Reilly Release Date: June 2013 ISBN: 9781449355739
Python Data Science Handbook: Tools and Techniques for Developers	Bibliografía	Autor: Jake VanderPlas Python Data Science Handbook: Tools and Techniques for Developers O'Reilly 25 marzo 2016 ISBN: 9781491912058

Head First Python	Bibliografía	<p>Autor: Paul Barry
 Head First Python, 2nd edition (16 diciembre 2016)
 Editor: O'Reilly
 ISBN: 9781491919538</p>
Python Programming: An Introduction to Computer Science	Bibliografía	<p>Autor: John Zelle
 Python Programming: An Introduction to Computer Science, 3rd Edition (8 de agosto de 2016)
 Franklin, Beedle & Associates Inc
 ISBN: 9781590282755</p>

8. Otra información

8.1. Otra información sobre la asignatura

La asignatura se imparte en modo presencial. Los alumnos cuentan con recursos Moodle en forma de ejercicios de autoevaluación (modo *quiz* y *CodeRunner*), ejercicios propuestos y resueltos, documentación de la asignatura preparada para cada clase, etc. Las clases se imparten de manera dinámica e interactiva. Los alumnos, desde el primer día, prueban y practican en el ordenador cada contenido explicado.

Esta asignatura, de carácter técnico, contribuye a alcanzar los Objetivos de Desarrollo Sostenible (ODS) (<https://sostenibilidad.upm.es/conoce-los-objetivos-de-desarrollo-sostenible/>); concretamente el "ODS9: Industria, innovación e infraestructura", ofreciendo una base de los conocimientos necesarios para desarrollar las diferentes metas de dicho objetivo, tales como: "9.1 Desarrollar infraestructuras fiables, sostenibles, resilientes y de calidad", o "9.5 Aumentar la investigación científica y mejorar la capacidad tecnológica industrial". Las tareas de la asignatura se orientan a la consecución del ODS9 a través del aprendizaje de los alumnos en las diferentes habilidades en el ámbito de los lenguajes de programación necesarias para alcanzar dichos objetivos.

