

Guía docente

820090 - PRE - Programación para Ingenieros

Última modificación: 14/06/2023

Unidad responsable: Escuela de Ingeniería de Barcelona Este
Unidad que imparte: 723 - CS - Departamento de Ciencias de la Computación.

Titulación: GRADO EN INGENIERÍA BIOMÉDICA (Plan 2009). (Asignatura optativa).
GRADO EN INGENIERÍA DE LA ENERGÍA (Plan 2009). (Asignatura optativa).
GRADO EN INGENIERÍA ELÉCTRICA (Plan 2009). (Asignatura optativa).
GRADO EN INGENIERÍA ELECTRÓNICA INDUSTRIAL Y AUTOMÁTICA (Plan 2009). (Asignatura optativa).
GRADO EN INGENIERÍA MECÁNICA (Plan 2009). (Asignatura optativa).
GRADO EN INGENIERÍA QUÍMICA (Plan 2009). (Asignatura optativa).
GRADO EN INGENIERÍA DE MATERIALES (Plan 2010). (Asignatura optativa).

Curso: 2023 **Créditos ECTS:** 6.0 **Idiomas:** Catalán, Castellano

PROFESORADO

Profesorado responsable: JAVIER FARRERES DE LA MORENA - FERRAN JUAN BARUEL

Otros: Primer quadrimestre:
JAVIER FARRERES DE LA MORENA - T11, T12
FERRAN JUAN BARUEL - T11, T12

CAPACIDADES PREVIAS

Éste es un segundo curso de programación. Preferiblemente es deseable que los alumnos ya tengan adquiridas las capacidades básicas de programación (hechas en Informática de Q1).

COMPETENCIAS DE LA TITULACIÓN A LAS QUE CONTRIBUYE LA ASIGNATURA

Específicas:

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

Transversales:

2. USO SOLVENTE DE LOS RECURSOS DE INFORMACIÓN - Nivel 3: Planificar y utilizar la información necesaria para un trabajo académico (por ejemplo, para el trabajo de fin de grado) a partir de una reflexión crítica sobre los recursos de información utilizados.

METODOLOGÍAS DOCENTES

Semanalmente habrá una sesión teórica y una sesión práctica en laboratorio con ordenadores. Paralelamente el alumno desarrollará un proyecto cuyo enunciado él mismo propondrá. Algunas sesiones de laboratorio durante el curso se dedicarán a la propuesta del alumno. Aprendizaje colaborativo. Aprendizaje basado en proyectos (PBL).

OBJETIVOS DE APRENDIZAJE DE LA ASIGNATURA

Objetivos de aprendizaje:

1. Analizar problemas complejos mediante mecanismos de reducción a problemas más pequeños.
2. Introducir al alumno en el uso de los tipos abstractos de datos
3. Introducir al alumno en la Programación Orientada a Objetos
4. Introducir al alumno en la Programación Orientada a Eventos y en el desarrollo de aplicaciones con interfície gráfica (GUI)



HORAS TOTALES DE DEDICACIÓN DEL ESTUDIANTADO

Tipo	Horas	Porcentaje
Horas grupo pequeño	30,0	20.00
Horas grupo grande	30,0	20.00
Horas aprendizaje autónomo	90,0	60.00

Dedicación total: 150 h

CONTENIDOS

Introducción

Descripción:

Se explican las tareas a realizar en la asignatura y el método de evaluación.

Objetivos específicos:

Recordatorio sobre la programación con acciones.

Actividades vinculadas:

Propuesta de proyecto por parte del alumno

Dedicación: 10h

Grupo grande/Teoría: 2h

Grupo pequeño/Laboratorio: 4h

Aprendizaje autónomo: 4h

Análisis y ciclo de vida

Descripción:

En este tema se da a los alumnos una visión general del proceso de desarrollo de una aplicación informática.

Objetivos específicos:

- Describir el ciclo de vida de una aplicación informática, y por extensión, de un proyecto de desarrollo informático
- Describir cómo realizar un análisis de requerimientos

Actividades vinculadas:

Análisis de requerimientos de la propuesta

Estado del arte en el diseño de aplicaciones

Dedicación: 6h

Clases teóricas: 2h

Aprendizaje autónomo: 4h

Diseño de aplicaciones

Descripción:

Se explican y se ponen en práctica diversas metodologías de diseño de aplicaciones.

Objetivos específicos:

Diseño descendente
Diseño modular
Diseño orientado a objetos
Diseño modular

Actividades vinculadas:

Diseño descendente de la propuesta (primer refinamiento)
Diseño completo de la propuesta
Diseño orientado a objetos de la propuesta

Dedicación: 36h

Grupo grande/Teoría: 14h
Grupo pequeño/Laboratorio: 2h
Aprendizaje autónomo: 20h

Tipos abstractos de datos

Descripción:

Se explica el concepto de Tipo Abstracto de Datos y se utilizan mediante llamadas a acciones predefinidas.

Objetivos específicos:

Describir y aprender a utilizar:
-ficheros
-punteros
-colas
-pilas
-listas

Actividades vinculadas:

Aplicación de Tipos Abstractos de Datos a la propuesta

Dedicación: 6h

Grupo grande/Teoría: 2h
Aprendizaje autónomo: 4h

Programación en entorno gráfico

Descripción:

Se explican las nociones necesarias para poder utilizar elementos visuales de programación y se desarrolla un proyecto.
Patrones de diseño Model View Presenter

Dedicación: 20h

Grupo pequeño/Laboratorio: 10h
Aprendizaje autónomo: 10h



Ampliación de programación

Descripción:

Módulos en Python
Ficheros
Errores i excepciones
Depuración i traza de código
Gestión de versiones con Git y Github

Dedicación: 8h

Grupo grande/Teoría: 4h
Grupo pequeño/Laboratorio: 4h

Desarrollo

Descripción:

Desarrollo de la propuesta por parte del alumno hasta llegar a obtener un programa completo y ejecutable

Actividades vinculadas:

Desarrollo de la propuesta

Dedicación: 64h

Grupo grande/Teoría: 10h
Grupo pequeño/Laboratorio: 10h
Aprendizaje autónomo: 44h

SISTEMA DE CALIFICACIÓN

La asignatura se evalúa en sucesivas entregas de un proyecto que el alumno desarrolla a lo largo del curso. Las distintas entregas y sus pesos son:

Diseño visual de la Aplicación 15%
Trabajo de Competencia 5%
Diseño Orientado a Objetos 20%
Portfolio 20%
Proyecto Final 40%

NORMAS PARA LA REALIZACIÓN DE LAS PRUEBAS.

No hay exámen final.

BIBLIOGRAFÍA

Básica:

- Summerfield, Mark. Rapid GUI programming with Python and Qt : the definitive guide to PyQt programming. Upper Saddle River, NJ: Prentice Hall, 2007. ISBN 9780132354189.
- Yourdon, Edward. Techniques of program structure and design. Englewood Cliffs, NJ: Prentice-Hall, 1975. ISBN 013901702X.
- Parnas, D. L. "On the criteria to be used in decomposing systems in modules". Communicatios of the ACM [en línea]. [Consulta: 22/04/2020]. Disponible a: <https://doi-org.recursos.biblioteca.upc.edu/10.1145/361598.361623>.
- Matthes, Eric. Python Crash Course : a hands-on, project-based introduction to programming [en línea]. San Francisco: No Starch Press, 2016 [Consulta: 09/06/2020]. Disponible a: <https://ebookcentral.proquest.com/lib/upcatalunya-ebooks/detail.action?docID=4503145>. ISBN 9781593277390.