

**Técnicas de Programación**

Código: 43851  
Créditos ECTS: 6

| Titulación             | Tipo | Curso | Semestre |
|------------------------|------|-------|----------|
| 4315985 Geoinformación | OT   | 0     | 2        |

**Contacto**

Nombre: Joan Bartrina Rapesta

Correo electrónico: Joan.Bartrina@uab.cat

**Equipo docente**

Katerine Diaz Chito

**Uso de idiomas**

Lengua vehicular mayoritaria: español (spa)

**Prerequisitos**

No hay prerequisitos. En todo caso es aconsejable tener nociones básicas de tecnologías de la información.

**Objetivos y contextualización**

Las técnicas de programación son la área de la informática que se dedica al desarrollo de aplicaciones. Algunos de los temas que aborda esta área es la programación orientada a objetos, funciones, recursiva y programación de aplicaciones web entre otras. De todas formas, en esta asignatura nos centraremos en la programación orientada a objetos y la de aplicaciones web.

**Competencias**

- Aplicar metodologías y procedimientos de programación e implantación de aplicaciones geoespaciales para distintos tipos de plataformas (escritorio, web, móvil), utilizando distintos paradigmas y entornos de programación.
- Desarrollar ideas imaginativas, creativas e innovadoras en proyectos de sistemas, servicios, productos o aplicaciones de información geoespacial.
- Poseer y comprender conocimientos que aporten una base u oportunidad de ser originales en el desarrollo y/o aplicación de ideas, a menudo en un contexto de investigación.
- Que los estudiantes posean las habilidades de aprendizaje que les permitan continuar estudiando de un modo que habrá de ser en gran medida autodirigido o autónomo.

**Resultados de aprendizaje**

1. Conocer y aplicar los diferentes paradigmas y entornos de programación.
2. Conocer y programar aplicaciones utilizando entornos de desarrollo integrado.
3. Desarrollar ideas imaginativas, creativas e innovadoras en proyectos de sistemas, servicios, productos o aplicaciones de información geoespacial.
4. Poseer y comprender conocimientos que aporten una base u oportunidad de ser originales en el desarrollo y/o aplicación de ideas, a menudo en un contexto de investigación.

5. Que los estudiantes posean las habilidades de aprendizaje que les permitan continuar estudiando de un modo que habrá de ser en gran medida autodirigido o autónomo.

## Contenido

### Programación orientada a objetos

1. Introducción a la programación orientada a objetos (OOP).
  - Conceptos de objeto, clase y método.
  - Introducción a la programación en Java.
  - Introducción al entorno de desarrollo *Eclipse*.
  - Nuestro primer programa.
3. Tipos de datos.
  - Tipos de datos.
  - Declaración de variables.
  - Operadores aritméticos, de comparación, lógicos.
  - Palabras reservadas en Java.
5. Manipulación de objetos.
  - Ventajas de usar objetos.
  - Introducción a los métodos.
  - Lectura de datos desde teclado.
7. Estructuras de control condicionales.
8. Estructuras de control iterativas.
9. *Arrays*.
10. Pilares de la programación orientada a objetos.
  - Encapsulamiento.
  - Herencia.
  - Polimorfismo. Clases y métodos abstractos.
12. Paquetes en Java.

### Programación web

1. Introducción a la programación web y a HTML.
  - Arquitectura cliente-servidor.
  - Elementos básicos del lenguaje HTML.
  - HTML5 vs HTML4.
3. *Cascading Style Sheets (CSS)*.
  - Como insertar CSS.
  - Tipos de selectores.
  - Modelo de cajas.
  - Posicionamiento.
  - Colores en HTML.
5. JavaScript.
  - Programación orientada a objetos.
  - Como insertar JavaScript.
  - Tipos de datos.
  - Variables.
  - Operadores.
  - Sentencias de control.
  - Objetos predefinidos.
  - Funciones.
  - Browser Object Model*.
  - Document Object Model*.
  - JavaScript Object Notation (JSON)*.
7. *Hypertext PreProcessor (PHP)*.
  - Introducción.
  - Sintaxis básica.
  - Tipos de datos.

- Variables.
- Operadores.
- Sentencias de control.
- Procesamiento de formularios.
- Acceso a bases de datos.
- Cookies.
- Asynchronous JavaScript and XML (AJAX).
- 9. Paradigma Modelo-Vista-Controlador.
  - Introducción.
  - Flujo de datos.

## Metodología

El módulo se desarrolla mediante tres grupos de actividades:

**Actividades dirigidas:** Consisten en clases teórico-prácticas en aulas informáticas e incluyen la resolución de casos mediante ejercicios prácticos guiados, aplicando como metodología principal el aprendizaje basado en problemas. Las clases constituyen el hilo conductor del módulo. Su función es sistematizar los contenidos, presentar estados de la cuestión de las materias, aportar métodos y técnicas para la resolución de tareas y recapitular los conocimientos objeto de aprendizaje. Asimismo, generan y organizan las necesidades de trabajo autónomo del alumno para ampliar contenidos básicos o desarrollar contenidos complementarios.

**Actividades supervisadas:** Comprenden la realización de un proyecto de cuatrimestre, consistente en un caso de aplicación real, mediante horas de taller, trabajo autónomo y tutorías, que permite aplicar conjuntamente los conocimientos y habilidades técnicas de los contenidos de todos los módulos del cuatrimestre. El proyecto de cuatrimestre constituye para el alumno/a un hito y la demostración material de haber alcanzado los objetivos de todos los módulos del cuatrimestre y es la pieza fundamental de la evaluación, pues además del seguimiento continuado de su realización, deberá entregar una memoria de síntesis del mismo y exponerlo oralmente.

**Actividades autónomas:** El trabajo autónomo del alumno incluye el tiempo para estudiar materiales teóricos (artículos, manuales, informes de interés, etc.), buscar documentación y datos, realizar ejercicios de ampliación de contenidos complementarios del módulo y, en gran medida, llevar a cabo el desarrollo personal del proyecto de cuatrimestre.

## Actividades

| Título                    | Horas | ECTS | Resultados de aprendizaje |
|---------------------------|-------|------|---------------------------|
| <b>Tipo: Dirigidas</b>    |       |      |                           |
| Clases teórico/aplicadas  | 36    | 1,44 | 1, 2, 5                   |
| <b>Tipo: Supervisadas</b> |       |      |                           |
| Proyecto                  | 15    | 0,6  | 1, 2, 3, 5                |
| <b>Tipo: Autónomas</b>    |       |      |                           |
| Trabajo Personal          | 69    | 2,76 | 1, 2, 3, 5                |

## Evaluación

EVALUACIÓN CONTINUADA

a) Proceso y actividades de evaluación:

La evaluación del módulo se basa principalmente en la realización del proyecto de cuatrimestre, el cual es objeto de dos actividades de evaluación. Por una parte, la elaboración y entrega de la memoria de síntesis del proyecto y por otra la defensa oral del proyecto realizado. Dado el contenido altamente técnico del módulo, se atribuye un peso del 50% a la memoria del proyecto, ya que es el medio más adecuado para exponer los detalles técnicos con toda su complejidad, y un peso del 35% a la defensa oral. La evaluación se complementa con un 15% de realización de ejercicios prácticos, debido a que la mayor parte de ejecución práctica corresponde a la realización del proyecto.

Salvo que se indique lo contrario, todas las actividades de evaluación (memoria del proyecto de cuatrimestre, exposición oral del proyecto de cuatrimestre, ejercicios prácticos del módulo) son individuales.

Las horas atribuidas a cada actividad de evaluación incluyen el tiempo destinado a la elaboración de los medios materiales de evaluación de cada actividad (memoria, presentación, etc.).

#### **b) Programación de actividades de evaluación:**

Memoria del proyecto del 2º cuatrimestre: Elaboración a lo largo del cuatrimestre. Entrega al final del periodo de actividades del módulo, **el 12 de abril de 2019**.

Defensa oral del proyecto del 2º cuatrimestre: Elaboración a lo largo del cuatrimestre. Exposición oral o presentación comentada al final del periodo de actividades del módulo, **el 5 de abril de 2019**.

Ejercicios prácticos del módulo: Realización y entrega semanal o quincenal, a lo largo del cuatrimestre.

#### **c) Procedimiento de revisión de la evaluación:**

Una vez publicadas las notas, los alumnos dispondrán de una semana para efectuar la revisión solicitando cita con los profesores o profesoras correspondientes.

#### **d) Proceso de recuperación:**

Memoria del proyecto del 2º cuatrimestre: Recuperable en el plazo máximo de 2 semanas después de la fecha de entrega programada. La recuperación consistirá en una nueva entrega de toda la memoria en caso de evaluación negativa de la memoria.

Defensa oral del proyecto del 2º cuatrimestre: Recuperable en el plazo máximo de 1 semana después de la fecha de realización programada. La recuperación consistirá en efectuar de nuevo la defensa oral en caso de evaluación negativa de la primera defensa oral realizada.

Ejercicios prácticos del módulo: No recuperables.

Para participar en la recuperación el alumno/a deberá haber sido previamente evaluado en un conjunto de actividades cuyo peso equivalga por lo menos a dos tercios de la evaluación total del módulo. Por lo tanto, deberá haber sido evaluado necesariamente en la fecha programada de la memoria (50%) y de la defensa oral (35%) del proyecto de cuatrimestre, que, juntas, corresponden al 85% de la evaluación total del módulo.

Sólo podrá participar en el proceso de recuperación el alumno/a que, no habiendo superado la evaluación del módulo (calificación total mínima de 5,0), haya obtenido una calificación mínima total del módulo superior a 3,5.

### **Actividades de evaluación**

| Título                   | Peso | Horas | ECTS | Resultados de aprendizaje |
|--------------------------|------|-------|------|---------------------------|
| Defensa oral de trabajos | 35%  | 10,5  | 0,42 | 1, 2, 5, 4                |
| Entrega trabajo/informes | 50%  | 15    | 0,6  | 1, 2, 5                   |
| Prácticas                | 15%  | 4,5   | 0,18 | 1, 2, 3, 5                |

## Bibliografía

Durante el transcurso de la sesiones se iran aportando distintos recursos bibliográficos y se indicará com consultarlos de forma eficiente.

- Sikora, Zbigniew M, *Java: practical guide for programmers*, Amsterdam ; Boston : Morgan Kaufmann, 2003
- Terry Felke-Morris, *Web development and design foundations with HTML5*, Addison-Wesley, 2012
- Medinets, David, *PHP3 programing browser-based applications*, McGraw-Hill, cop. 2000