


## Parte Didáctica

Módulo: Programación – 1.º DAM

Proyecto: “La startup de los objetos”

## 2.1 ¿Qué supuesto queremos trabajar?

Simular un desarrollo profesional desde el diseño hasta la implementación de un sistema basado en objetos.

 Aplicación de POO: clases, relaciones, encapsulamiento, herencia, polimorfismo y principios de diseño.

## 2.2 Contextualización del alumnado

- Curso: 1.º DAM
- Edad media: 17–20 años
- Nivel: inicial–intermedio
- Grupos heterogéneos con experiencia previa desigual en programación estructurada.

## 2.3 Conocimientos previos requeridos

- Programación estructurada (variables, bucles, condicionales)
- Uso básico de funciones
- Nociones básicas de clases y objetos
- Compilación y ejecución en Java o Python

## 2.4 Objetivos de aprendizaje

- Aplicar el paradigma de programación orientada a objetos
- Identificar entidades y modelarlas como clases
- Entender y usar relaciones: herencia, composición, asociación
- Justificar decisiones de diseño con principios SOLID
- Mejorar el trabajo en equipo y la capacidad de presentación técnica

## 2.5 Metodología

- Enfoque **ABP** (Aprendizaje Basado en Proyectos)
- Trabajo cooperativo en grupos
- Evaluación formativa y sumativa
- Aprendizaje por descubrimiento y reflexión
- Tutorías individuales y dinámicas de clase invertida

## 2.6 Material didáctico (DUA)

- Lenguajes: Java o Python
- IDEs: IntelliJ, VSCode, Replit o BlueJ
- Plantillas de diseño UML básicas
- Recursos visuales y guías paso a paso
- Accesibilidad: opción de pseudocódigo / representación visual previa
- Feedback adaptado a necesidades individuales

## 2.7 Secuencia de acciones formativas

1. Introducción a la POO y repaso de fundamentos
2. Diseño del modelo de dominio (UML básico)
3. Implementación de clases, relaciones y principios POO
4. Aplicación progresiva de encapsulamiento y herencia
5. Desarrollo iterativo en equipo
6. Preparación de la presentación simulada
7. Exposición del proyecto (reunión técnica con feedback)



## 2.8 Actividad principal:

### Proyecto “La startup de los objetos”

- ◆ Cada grupo define un caso de uso (e-commerce, reservas, biblioteca, etc.)
- ◆ Identifican clases: Usuario, Producto, Pedido...
- ◆ Aplican relaciones:
  - Herencia
  - Composición
  - Asociación
- ◆ Implementación en **Java o Python**, con buenas prácticas:
  - Clases organizadas en paquetes o módulos
  - Encapsulamiento
  - Comentarios y legibilidad

## 2.9 Evaluación: instrumentos y criterios

### Instrumentos:

- Observación directa y rúbrica de proyecto
- Presentación oral
- Código entregado y documentación técnica






### Criterios:

- Diseño coherente y orientado a objetos
- Aplicación correcta de encapsulamiento y relaciones
- Justificación de decisiones de diseño
- Funcionalidad y claridad del código
- Calidad de la exposición

## 2.10 Inclusión y atención a la diversidad

- Plantillas guiadas para quienes necesiten más apoyo
- Elección de lenguaje (Java o Python) según perfil del grupo
- Refuerzo visual (UML, diagramas de flujo)
- Tutorías individualizadas
- Valoración del esfuerzo y mejora personal

## 2.11 Actividades de ampliación

-  Añadir capa de persistencia con una librería ORM
-  Incorporar pruebas unitarias con JUnit o PyTest
-  Crear una interfaz gráfica simple (Swing, Tkinter)
-  Documentar con README técnico y diagrama de clases
-  Simulación de validación de requisitos con feedback externo

## ¡Construye tu startup de objetos!

Programa como si diseñaras un producto real.

Presenta como si estuvieras ante inversores.

 De la idea → al código → al diseño profesional.