


Parte Didáctica





Módulo: Programación – 1.º DAM

Actividad: “Diseña tu algoritmo estrella”

2.1 ¿Qué supuesto queremos trabajar?

 Diseñar, implementar y presentar un algoritmo eficiente usando estructuras de control y funciones, aplicando principios de programación estructurada, dentro de una experiencia gamificada.

2.2 Contextualización del alumnado

-  Alumnado de 1.º DAM en un punto intermedio del módulo.
-  Nivel: medio, con bases en condicionales, bucles y funciones.
-  Necesidad: consolidar lógica algorítmica, depuración y diseño claro.
-  Se potencia el trabajo colaborativo, la reflexión crítica y la exposición técnica.






2.3 Conocimientos previos requeridos

- Uso de `if`, `switch`, `for`, `while`
- Creación y uso de funciones
- Entrada y salida estándar
- Diseño modular y básico de algoritmos
- Manejo básico de Eclipse o IntelliJ

2.4 Objetivos de aprendizaje

- ✓ Dominar el uso de estructuras básicas de control
- ✓ Diseñar soluciones modulares y escalables
- ✓ Aplicar principios de cohesión y bajo acoplamiento
- ✓ Evaluar la eficiencia y calidad de un algoritmo
- ✓ Argumentar y defender decisiones técnicas en equipo

2.5 Metodología

-  Metodología activa, basada en retos
-  Trabajo por equipos
-  Elementos gamificados: "casting de funciones"
-  Evaluación formativa, revisión entre pares
-  Desarrollo incremental (de diseño a código y prueba)

2.6 Material didáctico (DUA)

- Guía de fases con orientaciones técnicas
- Entorno de desarrollo libre: Eclipse / IntelliJ
- Plantilla de justificación técnica por función
- Posibilidad de usar papel o medios digitales
- Apoyos visuales y rúbrica accesible para todos los niveles

2.7 Secuencia de acciones formativas

1. Introducción al reto y reparto de problemas (turnos, reservas, accesos...)
2. Diseño algorítmico modular en papel o pseudocódigo
3. Codificación en Java con enfoque estructurado
4. Preparación de defensa técnica: entrevista a funciones
5. Presentación en formato casting técnico
6. Prueba del programa con casos reales

2.8 Actividad principal

“Diseña tu algoritmo estrella: el reality show de la eficiencia”

- ◆ Fase 1: Se asigna un reto cotidiano por grupo
- ◆ Fase 2: Diseño del algoritmo con estructuras + funciones
- ◆ Fase 3: Codificación en Java con buenas prácticas estructuradas
- ◆ Fase 4:

 *Entrevista técnica a las funciones*

Cada grupo defiende sus funciones clave:

- ¿Qué hace?
- ¿Por qué está bien diseñada?
- ¿Es eficiente? (iterativa vs. recursiva)
- ¿Qué la diferencia?

2.9 Evaluación: instrumentos y criterios



Criterios:

- Estructuración y lógica del algoritmo
- Uso adecuado de funciones y control de flujo
- Calidad del código y modularidad
- Capacidad de análisis y argumentación
- Funcionamiento real del programa



Instrumentos:

- Rúbrica técnica y de presentación
- Observación directa
- Evaluación entre pares
- Revisión de entregables y código fuente

2.10 Inclusión y atención a la diversidad

- ✓ Código base opcional con funciones incompletas
- ✓ Uso de lenguaje claro y visual
- ✓ Roles adaptados dentro del grupo (portavoz, programador, planificador...)
- ✓ Alternativas escritas para quienes tengan dificultades orales
- ✓ Apoyo personalizado durante las fases de codificación

2.11 Actividades de ampliación

 Para alumnado avanzado:

- Comparar eficiencia de versiones recursivas vs iterativas
- Añadir pruebas automatizadas (JUnit básico)
- Visualizar el flujo con diagramas Nassi-Shneiderman
- Añadir persistencia de datos (guardar en archivo)

Conclusión

 Una propuesta activa y transversal para:

- Aplicar fundamentos de programación estructurada
- Fomentar el pensamiento algorítmico crítico
- Valorar la calidad técnica con creatividad y análisis

 ¡La lógica al servicio de la comunicación técnica!