

# Supuesto Didáctico

1.º DAM – Módulo: Programación

Estructuras dinámicas – “Gestor musical”

## ◆ 2.1 ¿Qué supuesto queremos trabajar?

💡 Desarrollo de una aplicación basada en:

- Listas dinámicas enlazadas
- Implementación y gestión de nodos en Java
- Visualización de la evolución estructural en tiempo de ejecución

🎯 El objetivo es que el alumnado entienda cómo funcionan las estructuras enlazadas modificables.

## 2.2 Contextualización del alumnado

- Alumnado de 1.º DAM
- Nivel intermedio: ya han visto arrays, clases y métodos
- Familiarizados con programación estructurada y orientada a objetos
- Grupo heterogéneo en ritmo y experiencia previa

## 2.3 Conocimientos previos requeridos

- Uso de estructuras de control ( `if` , `for` , `while` )
- Clases y objetos en Java
- Métodos con paso de parámetros y retorno
- Manipulación de arrays y estructuras básicas

## 2.4 Objetivos de aprendizaje

- Comprender el uso de estructuras dinámicas (listas enlazadas)
- Implementar nodos y enlaces con referencias en Java
- Diseñar programas estructurados y reutilizables
- Visualizar cómo evoluciona la estructura en tiempo real
- Aplicar programación orientada a objetos en una situación funcional

## 2.5 Metodología






- ◆ Enfoque: **Aprendizaje basado en tareas**
- ◆ Trabajo individual con apoyo continuo
- ◆ Desarrollo progresivo de funcionalidades
- ◆ Observación y reflexión sobre el comportamiento dinámico del programa
- ◆ Evaluación formativa + sumativa

## 2.6 Material didáctico (DUA)

Aplicación del **Diseño Universal para el Aprendizaje**:

- Esquema inicial con nodos + referencias visuales
- Código base comentado
- Guía paso a paso con checklist de objetivos
- Alternativas de visualización (consola, dibujos, ASCII)
- Entrega flexible: funcional o con explicación visual

## 2.7 Secuencia de acciones formativas

1.  Introducción y motivación
  - ¿Cómo funcionan las estructuras vivas?
  - Comparación con listas estáticas
2.  Exposición teórica guiada
  - Nodos, referencias, enlaces, operaciones básicas
3.  Análisis del código base
  - Clase Nodo, clase Lista
4.  Prácticas guiadas
  - Insertar, eliminar, recorrer
5.  Desarrollo individual de la actividad principal



## 2.8 Actividad principal: “Gestor musical”

El alumnado desarrolla un programa en Java para:

- Añadir canciones con título, artista, duración y género
- Agrupar las canciones en **listas enlazadas por género**
- Permitir:
  - Buscar canciones
  - Eliminar canciones
  - Listar canciones por género
- Visualizar la estructura tras cada operación

 Simula una colección musical como estructura viva







## 2.9 Evaluación: instrumentos y criterios

### Criterios

- Correcta implementación de listas enlazadas
- Modularidad y diseño POO
- Capacidad para explicar la estructura
- Claridad y funcionalidad del código

### Instrumentos

-  Rúbrica detallada por funcionalidad
-  Revisión individual
-  Posible defensa del código
-  Observación del desarrollo durante la práctica



## 2.10 Inclusión y atención a la diversidad

- Código base con partes opcionales
- Acompañamiento individual según ritmo
- Visualización alternativa de la estructura
- Explicaciones adaptadas al nivel previo
- Entrega del proyecto con consola o PDF descriptivo

## 2.11 Actividades de ampliación

Para alumnado avanzado:

- Añadir ordenación por duración o título
- Implementar estadísticas (canciones por género, tiempo total)
- Crear una interfaz gráfica simple (Swing o JavaFX)
- Cargar/guardar canciones desde fichero
- Añadir un sistema de reproducción aleatoria

 ¡Convierte tu estructura dinámica en un reproductor real!