# Informe de Modelado del Sistema - Academia de Música

## 1. Portada

\*\*Proyecto:\*\* Sistema de Gestión para una Academia de Música  
\*\*Autor:\*\* [Tu Nombre]  
\*\*Fecha:\*\* [Fecha Actual]

## 2. Introducción

El presente informe tiene como objetivo documentar el proceso de modelado del sistema de gestión para una academia de música. Este sistema permitirá a los administradores gestionar usuarios, a los profesores registrar asistencias y planificar cursos, y a los estudiantes consultar sus horarios y verificar sus asistencias.  
  
El modelado del sistema se ha realizado utilizando el estándar \*\*UML (Unified Modeling Language)\*\*, el cual proporciona una representación visual clara y organizada de los componentes del software. Este informe incluye los principales artefactos generados durante la fase de análisis y diseño del proyecto.

## 3. Objetivos

### 3.1 Objetivo General

- Desarrollar un modelo conceptual claro y organizado del sistema de gestión de una academia de música mediante diagramas UML, con el fin de comprender su estructura y comportamiento antes de la implementación.

### 3.2 Objetivos Específicos

- Identificar y representar las clases, atributos, métodos y relaciones del sistema mediante un \*\*Diagrama de Clases\*\*.

- Organizar las dependencias y relaciones del sistema a través de un \*\*Diagrama de Paquetes\*\*.

- Visualizar el flujo de las principales actividades del sistema mediante un \*\*Diagrama de Actividades\*\*.

- Modelar la estructura de la base de datos a través de un \*\*Diagrama Entidad-Relación (ER)\*\*.

- Documentar los casos de uso y las historias de usuario siguiendo un estándar formal.

## 4. Desarrollo del Informe

### 4.1 Diagrama de Clases

El diagrama de clases refleja la estructura del sistema, mostrando las entidades principales y sus relaciones.  
  
\*\*Descripción:\*\*  
- La clase `Usuario` es una clase base de la cual heredan `Administrador`, `Profesor` y `Estudiante`.  
- Los `Profesores` pueden registrar asistencias y están asociados a uno o varios `Cursos`.  
- Los `Estudiantes` están inscritos en `Cursos` y tienen registros de asistencia.  
  
![Diagrama de Clases](inserta\_aquí\_la\_ruta\_de\_la\_imagen)

### 4.2 Diagrama de Paquetes

El diagrama de paquetes organiza las clases en módulos lógicos para mejorar la comprensión del sistema.  
  
\*\*Descripción:\*\*  
- \*\*Usuarios:\*\* Contiene las clases relacionadas con el manejo de los usuarios (Administrador, Profesor, Estudiante).  
- \*\*Cursos:\*\* Agrupa las clases `Curso` y `Horario`.  
- \*\*Asistencias:\*\* Gestiona el registro y seguimiento de las asistencias.  
- \*\*Autenticación:\*\* Maneja el control de acceso al sistema.  
  
![Diagrama de Paquetes](inserta\_aquí\_la\_ruta\_de\_la\_imagen)

### 4.3 Diagrama de Actividades

Este diagrama representa el flujo de trabajo del sistema, describiendo las acciones desde el inicio de sesión hasta la gestión de cursos y asistencia.  
  
\*\*Descripción:\*\*  
- El flujo inicia con la autenticación del usuario.  
- Los administradores gestionan usuarios, los profesores registran asistencia y los estudiantes consultan sus horarios.  
  
![Diagrama de Actividades](inserta\_aquí\_la\_ruta\_de\_la\_imagen)

### 4.4 Modelo de Base de Datos (ER)

El modelo Entidad-Relación describe las tablas, sus atributos y relaciones, que sirven como base para la implementación del sistema.  
  
\*\*Descripción:\*\*  
- Entidades principales: `Usuario`, `Curso`, `Horario`, `Asistencia`.  
- Relaciones clave: Un profesor imparte varios cursos; un estudiante está inscrito en uno o varios cursos.  
  
![Diagrama ER](inserta\_aquí\_la\_ruta\_de\_la\_imagen)

### 4.5 Documentos de Casos de Uso

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | Nombre del Caso de Uso | Actor | Descripción | Precondiciones | Flujo Principal | Postcondición |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| CU01 | Gestionar Usuarios | Administrador | Permite crear, editar y eliminar usuarios. | Usuario autenticado | 1. Iniciar sesión. | Usuario gestionado correctamente. |
| CU02 | Registrar Asistencia | Profesor | Registrar la asistencia de los estudiantes. | Profesor autenticado | 1. Seleccionar curso. 2. Registrar. | Asistencia almacenada. |
| CU03 | Consultar Horarios | Estudiante | Permite al estudiante revisar su horario. | Estudiante autenticado | 1. Acceder a 'Mis Horarios'. | Visualización del horario. |
| CU04 | Autenticación del Usuario | Usuario | Iniciar sesión en el sistema con credenciales. | Usuario registrado | 1. Introducir credenciales. | Acceso al sistema permitido. |

## 5. Conclusiones

- El modelado del sistema mediante UML ha permitido comprender de forma clara y detallada la estructura y funcionamiento del sistema de gestión de la academia de música.  
- La segmentación en paquetes facilita la organización modular del código, permitiendo una implementación más eficiente y mantenible.  
- Este informe proporciona una base sólida para la siguiente fase del proyecto: la implementación del sistema.

## 6. Anexos

Se adjuntan los códigos fuente de los diagramas en formato PlantUML para su edición y visualización en herramientas compatibles como \*\*VSCode con PlantUML\*\*.