

# Sistema de Gestión de Librerías y Empresa con JPA/Hibernate



## Índice

---

1. [Descripción y Objetivos](#)
2. [Arquitectura y Diseño](#)
3. [Modelo de Datos](#)
4. [Funcionalidades Implementadas](#)
5. [Decisiones Técnicas](#)
6. [Resultados y Evidencias](#)
7. [Conclusiones](#)

## Descripción y Objetivos

---

Este sistema implementa dos modelos de negocio utilizando JPA/Hibernate como tecnología de persistencia:

1. **Requerimiento 1:** Modelo de datos para gestionar una cadena de librerías, incluyendo autores, editoriales, libros y librerías, con relaciones bidireccionales.
2. **Requerimiento 2:** Modelo empresarial que demuestra los tres tipos de relaciones (1:1, 1:N, N:M) utilizando empleados, departamentos, direcciones y proyectos.

## Objetivos Principales

- Implementar las entidades JPA con sus relaciones bidireccionales
- Demostrar el funcionamiento de los tres tipos de relaciones
- Crear consultas que evidencien el funcionamiento correcto de las relaciones
- Proporcionar una interfaz de usuario mediante consola para interactuar con el sistema



## Requisitos del Sistema

---

- Java JDK 23
- MySQL 9.1.0
- Hibernate ORM 6.6.3.Final
- Maven
- Lombok



## Arquitectura y Diseño

---



### Estructura del Sistema

```
Usuario
  ↓
MainController
  ↓
CompanyController / BookstoreController
  ↓
Services (BookstoreService / CompanyService)
  ↓
DAO Layer (AutorDAO, LibroDAO, EmpleadoDAO, etc.)
  ↓
Entities (JPA Models)
  ↓
HibernateUtil
  ↓
MySQL DB
```



### Capas del Sistema

#### 1. Controladores:

- **MainController** : Coordina la navegación entre módulos
- **BookstoreController** y **CompanyController** : Gestionan menús específicos

#### 2. Servicios:

- **BookstoreService** y **CompanyService** : Implementan la lógica de negocio

- Orquestan las operaciones entre entidades y DAOs

### 3. DAOs:

- Clases DAO para cada entidad (AutorDAO, LibroDAO, EmpleadoDAO, etc.)
- Encapsulan las operaciones de persistencia y consultas

### 4. Entidades:

- Clases mapeadas con anotaciones JPA/Hibernate
- Incluyen métodos helper para mantener la consistencia bidireccional

### 5. Utilidades:

- `HibernateUtil` : Gestiona la sesión de Hibernate
- `Constants` : Almacena mensajes y valores constantes

## Modelo de Datos

---

## Requerimiento 1: Sistema de Librerías

### Entidades y Relaciones:

#### 1. Autor (1:N con Libro)

- Atributos: id, nombre, apellidos, fechaNacimiento
- Relación: Un autor escribe muchos libros

#### 2. Editorial (1:N con Libro)

- Atributos: id, nombre, dirección
- Relación: Una editorial publica muchos libros

#### 3. Libro (N:1 con Autor/Editorial, N:M con Librería)

- Atributos: id, título, precio
- Relaciones: Pertenece a un autor y editorial, está en varias librerías

#### 4. Librería (N:M con Libro)

- Atributos: id, nombre, nombreDueño, dirección

- Relación: Una librería tiene muchos libros

## Requerimiento 2: Sistema Empresarial

### Entidades y Relaciones:

#### 1. Empleado (1:1, N:1, N:M)

- Atributos: id, nombre, apellido, email, fechaContratacion, salario
- Relaciones:
  - Tiene una dirección (1:1)
  - Pertenece a un departamento (N:1)
  - Participa en múltiples proyectos (N:M)

#### 2. Dirección (1:1 con Empleado)

- Atributos: id, calle, ciudad, codigoPostal, país

#### 3. Departamento (1:N con Empleado)

- Atributos: id, nombre, descripción, ubicación

#### 4. Proyecto (N:M con Empleado)

- Atributos: id, nombre, descripción, fechaInicio, fechaFin, presupuesto

## Funcionalidades Implementadas

---

### Sistema de Librerías

- **Gestión completa** de autores, editoriales, libros y librerías
- **Consultas principales:**
  - Mostrar libros con su editorial y autor
  - Mostrar autores con sus libros asociados
  - Mostrar librerías con sus libros
  - Mostrar libros y las librerías donde están disponibles

## Sistema Empresarial

- **Relación Uno a Uno:** Empleado-Dirección
  - Asociación directa entre un empleado y su dirección
- **Relación Uno a Muchos:** Departamento-Empleados
  - Un departamento contiene múltiples empleados
- **Relación Muchos a Muchos:** Empleado-Proyecto
  - Empleados participan en múltiples proyectos
  - Proyectos tienen varios empleados asignados

## Decisiones Técnicas

---

### Patrones y Prácticas Clave

#### 1. Named Queries

- Consultas JPQL centralizadas en las entidades
- Consultas reutilizables y mejor organizadas

#### 2. Métodos Helper para Relaciones

- Métodos como `addLibro()` , `asignarProyecto()` , etc.
- Mantienen la consistencia bidireccional de las relaciones

#### 3. Gestión de Transacciones

- Control granular en los DAOs
- Soporte para transacciones anidadas

#### 4. Uso de Cascadas

- `CascadeType.ALL` para relaciones 1:1
- `CascadeType.PERSIST` y `MERGE` para otras relaciones
- Evita eliminaciones en cascada no deseadas

#### 5. Lazy Loading

- Optimización con FetchType.LAZY
- FETCH JOIN para consultas específicas

## 6. Lombok

- Reduce código boilerplate manteniendo claridad
- Anotaciones @Getter, @Setter, etc.

## 7. HibernateUtil

- Patrón Singleton para gestión de sesiones
- Gestión explícita de recursos y transacciones

# Resultados y Evidencias

---

## Menús del Sistema

La aplicación presenta una interfaz de consola con menús intuitivos:

```
===== SISTEMA DE GESTIÓN JPA =====  
1. Gestión de Librería (Requerimiento 1)  
2. Gestión de Empresa (Requerimiento 2)  
3. Salir del sistema
```

### Menú de Gestión de Librería

```
===== GESTIÓN DE LIBRERÍA =====  
1. Crear datos de ejemplo  
2. Mostrar libros con editorial y autor  
3. Mostrar autores con sus libros  
4. Mostrar librerías con sus libros  
5. Mostrar libros y sus librerías  
6. Eliminar todos los datos  
7. Volver al menú principal
```

### Menú de Gestión de Empresa

```
===== GESTIÓN DE EMPRESA =====  
1. Crear datos de ejemplo  
2. Mostrar relación Uno a Uno (Empleado-Dirección)
```

3. Mostrar relación Uno a Muchos (Departamento–Empleados)
4. Mostrar relación Muchos a Muchos (Empleado–Proyecto)
5. Eliminar todos los datos
6. Volver al menú principal

## Ejemplos de Consultas Ejecutadas

### Requerimiento 1: Librerías

#### 1. Libros con editorial y autor:

```
===== CONSULTA 1: TODOS LOS LIBROS CON SU EDITORIAL Y AUTOR =====  
Libro: La fiesta del chivo – Precio: 26.75€ – Editorial: Penguin  
Random House – Autor: Mario Vargas Llosa  
Libro: La ciudad y los perros – Precio: 24.50€ – Editorial: Penguin  
Random House – Autor: Mario Vargas Llosa  
Libro: El amor en los tiempos del cólera – Precio: 22.50€ –  
Editorial: Penguin Random House – Autor: Gabriel García Márquez  
Libro: Cien años de soledad – Precio: 25.99€ – Editorial: Penguin  
Random House – Autor: Gabriel García Márquez  
Libro: Eva Luna – Precio: 19.99€ – Editorial: Planeta – Autor:  
Isabel Allende  
Libro: La casa de los espíritus – Precio: 21.75€ – Editorial:  
Planeta – Autor: Isabel Allende  
Libro: El otoño del patriarca – Precio: 20.25€ – Editorial: Planeta  
– Autor: Gabriel García Márquez  
Libro: Crónica de una muerte anunciada – Precio: 18.99€ –  
Editorial: Planeta – Autor: Gabriel García Márquez
```

#### 2. Autores con sus libros:

```
===== CONSULTA 2: TODOS LOS AUTORES CON SUS LIBROS ASOCIADOS =====  
Autor: Gabriel García Márquez  
– Cien años de soledad (25.99€)  
– El amor en los tiempos del cólera (22.50€)  
– Crónica de una muerte anunciada (18.99€)  
– El otoño del patriarca (20.25€)  
  
Autor: Isabel Allende  
– La casa de los espíritus (21.75€)  
– Eva Luna (19.99€)  
  
Autor: Mario Vargas Llosa  
– La ciudad y los perros (24.50€)  
– La fiesta del chivo (26.75€)
```

### 3. Librerías con sus libros:

===== CONSULTA 3: TODAS LAS LIBRERÍAS CON SUS LIBROS ASOCIADOS =====

Librería: El Rincón Literario – Dueño: Ana Gómez – Dirección: Calle Mayor 12, Madrid

- La ciudad y los perros
- Cien años de soledad
- La casa de los espíritus
- Crónica de una muerte anunciada

Librería: Libros y Más – Dueño: Carlos Ruiz – Dirección: Avenida Libertad 45, Barcelona

- La fiesta del chivo
- El amor en los tiempos del cólera
- Eva Luna
- El otoño del patriarca

### Requerimiento 2: Empresa

#### 1. Relación Uno a Uno:

===== RELACIÓN UNO A UNO (Empleado–Dirección) =====

Empleado: Juan Pérez

Dirección: Calle Mayor 10, Madrid, 28001, España

Empleado: María López

Dirección: Avenida Diagonal 100, Barcelona, 08018, España

Empleado: Carlos Gómez

Dirección: Plaza Nueva 5, Sevilla, 41001, España

Empleado: Ana Martínez

Dirección: Gran Vía 30, Madrid, 28013, España

#### 2. Relación Uno a Muchos:

===== RELACIÓN UNO A MUCHOS (Departamento–Empleados) =====

Departamento: Tecnología – Departamento de Tecnología e Innovación

– Ubicación: Planta 3

Empleados:

- Juan Pérez – Contratado: 2020-01-15
- María López – Contratado: 2019-05-10

Departamento: RRHH – Recursos Humanos – Ubicación: Planta 2

Empleados:

- Carlos Gómez – Contratado: 2021-03-22



Departamento: Marketing – Departamento de Marketing y Ventas –  
Ubicación: Planta 1  
Empleados:  
– Ana Martínez – Contratado: 2018-11-05

### 3. Relación Muchos a Muchos:

```
===== RELACIÓN MUCHOS A MUCHOS (Proyecto-Empleados) =====  
--- Proyectos y sus Empleados: ---  
Proyecto: Web Corporativa – Rediseño de la web corporativa –  
Presupuesto: 25000.0€  
Empleados asignados:  
– Juan Pérez – juan.perez@empresa.com  
– Ana Martínez – ana.martinez@empresa.com  
  
Proyecto: App Móvil – Desarrollo de app para clientes –  
Presupuesto: 45000.0€  
Empleados asignados:  
– María López – maria.lopez@empresa.com  
– Juan Pérez – juan.perez@empresa.com
```

## Conclusiones

---

El proyecto implementa satisfactoriamente los dos requerimientos solicitados:

1. Se ha creado un modelo completo para la gestión de librerías con relaciones bidireccionales entre autores, editoriales, libros y librerías.
2. Se ha desarrollado un modelo empresarial que demuestra los tres tipos de relaciones: uno a uno (Empleado-Dirección), uno a muchos (Departamento-Empleados) y muchos a muchos (Empleado-Proyecto).

El uso de JPA/Hibernate ha permitido crear un mapeo objeto-relacional efectivo, demostrando las capacidades de esta tecnología para gestionar modelos de datos complejos con distintos tipos de relaciones.

## Autores

---

- **Oriol Macias** – Desarrollador