

# AEV3 - Aplicación de Gestión de Biblioteca con Symfony

## Objetivos

- Demostrar el conocimiento adquirido durante el tema 7 sobre el Framework Symfony.
- Crear una aplicación web funcional con un caso de uso real.
- Aplicar los conceptos de controladores, rutas, entidades, relaciones y formularios.

## Recursos generales

- Presentaciones y vídeos del Tema 7.
  - Documentación oficial de Symfony.
  - Códigos de ejemplo vistos en clase y en las actividades previas (AP7-1 a AP7-5).
  - **No se puede utilizar ninguna IA. Si se detecta que se ha utilizado para generar el código, la evaluable se marcará como insuficiente.**
- 

## Contexto

Eres el desarrollador de una pequeña biblioteca de barrio que necesita digitalizar su gestión. Actualmente llevan todo en papel y quieren una aplicación web sencilla para:

- Gestionar el catálogo de libros
  - Registrar los socios de la biblioteca
  - Controlar los préstamos de libros
- 

## Requisitos Técnicos

- **Versión de Symfony:** 7.4
  - **Nombre del proyecto:** AEV3
  - **Nombre de la base de datos:** biblioteca
- 

## Modelo de Datos

Debes crear las siguientes entidades con sus relaciones:

**Entidad Book (Libro)**

Campo	Tipo	Restricciones
id	integer	PK, autoincremental
title	string(255)	Obligatorio
isbn	string(13)	Obligatorio, único
publishedYear	integer	Opcional
available	boolean	Por defecto: true

**Entidad Member (Socio)**

Campo	Tipo	Restricciones
id	integer	PK, autoincremental
name	string(100)	Obligatorio
surname	string(150)	Obligatorio
email	string(180)	Obligatorio, único
phone	string(15)	Opcional
registeredAt	datetime	Obligatorio, por defecto fecha actual

**Entidad Loan (Préstamo)**

Campo	Tipo	Restricciones
id	integer	PK, autoincremental
loanDate	datetime	Obligatorio, por defecto fecha actual
returnDate	datetime	Opcional (null si no se ha devuelto)

**Relaciones**

- **Loan - Book:** Muchos préstamos pueden ser del mismo libro (en diferentes momentos). Un préstamo es de un libro.

- **Loan - Member:** Muchos préstamos pueden ser del mismo socio. Un préstamo pertenece a un socio.
- 

## Tareas a Realizar

### Ejercicio 0 - Configuración Inicial (Obligatorio)

1. Inicia el contenedor ejecutando el archivo `docker-compose.yml` con clic derecho → Run.
2. Crea una nueva conexión a la base de datos (MariaDB) en PhpStorm usando el puerto `3330`.
3. Crear un nuevo proyecto Symfony llamado `AEV3`.
4. Configurar la conexión a la base de datos en el archivo `.env`.
5. Crear la base de datos mediante el comando de Symfony.
6. Verificar que el proyecto arranca correctamente en `http://localhost:8000`.

### Ejercicio 1 - Página Principal

1. Crear un `MainController` con una ruta raíz `/`.
2. La página principal debe mostrar:
  - Nombre de la aplicación: “Biblioteca Municipal”
  - Una imagen representativa de la biblioteca (añadir una imagen en la carpeta `public/images/`).
  - Un menú de navegación con enlaces a: Libros, Socios y Préstamos.
  - Este menú debe aparecer en **todas las páginas** (usar herencia de plantillas Twig).

### Ejercicio 2 - CRUD de Libros y Socios

1. Crear la entidad `Book` con los campos especificados.
2. Crear la entidad `Member` con los campos especificados.
3. Generar el CRUD completo.
4. **Personalización obligatoria:**
  - Las etiquetas de los formularios deben estar en castellano.
  - Añadir validación en Book: el ISBN debe tener exactamente 13 caracteres.
  - Añadir validación en Member: el email debe tener formato válido.
  - El campo `registeredAt` de Member debe establecerse automáticamente al crear (no editable en formulario).

## Ejercicio 3 - CRUD de Préstamos

1. Crear la entidad `Loan` con los campos especificados y las relaciones.
2. Generar el CRUD completo.
3. **Personalización obligatoria:**
  - En el formulario, el libro y el socio deben seleccionarse de desplegables.
  - Las etiquetas del formulario deben estar en castellano.
  - El campo `loanDate` debe establecerse automáticamente al crear (no editable).
  - El campo `returnDate` solo debe aparecer en la edición (para registrar devoluciones).
  - En el listado, mostrar el título del libro y el nombre del socio.

## Ejercicio 4 - Funcionalidades

1. **Estadísticas en página principal:**
  - Mostrar en la página de inicio: total de libros, libros disponibles, total de socios y préstamos activos.
2. **Detalle de socio con sus préstamos ( /member/{id} ):**
  - En la vista de detalle del socio, mostrar un listado de todos sus préstamos (activos e históricos).
3. **Contador de préstamos por socio:**
  - En el listado de socios, mostrar cuántos préstamos tiene cada socio y cuántos están activos.
4. **Listado de préstamos activos ( /loan/active ):**
  - Mostrar solo los préstamos donde `returnDate` sea `null`.
  - Incluir botón “Marcar como devuelto” que establezca `returnDate` a la fecha actual.
5. **Disponibilidad del libro:**
  - Al crear un préstamo, el campo `available` del libro debe ponerse a `false`.
  - Al marcar como devuelto, el campo `available` debe ponerse a `true`.
  - En el formulario de préstamo, solo mostrar libros disponibles en el desplegable.

---

## Entregables

1. **Repositorio GitHub:**
  - Realizar **mínimo 4 commits** demostrando progreso adecuado.
  - Los commits deben tener mensajes descriptivos.

**2. Vídeo demostrativo** (máximo 10 minutos):

- Mostrar la aplicación funcionando.
- Demostrar el CRUD de al menos una entidad completa.
- Mostrar la funcionalidad de préstamos (crear préstamo y marcar devolución).
- Subir a ScreenPal/Youtube/OneDrive y compartir el enlace.

**3. Actualizar este README** con:

- Comandos ejecutados en cada ejercicio.
- Enlace al vídeo.
- Autorúbrica cumplimentada.

**4. Archivo SQL** con el backup de la base de datos (mínimo: 3 libros, 2 socios, 2 préstamos).

- Ejecutar desde la terminal (fuera del contenedor):

```
docker exec servidor_db_aev3 mariadb-dump -u root -p root biblioteca > biblioteca.sql
```

**5. Archivo comprimido para subir a FloridaOberta:**

- Todo el proyecto **sin la carpeta /vendor ni var**.
- Usar los siguientes comandos para comprimir:

```
# Mac/Linux
zip -r AEV3.zip AEV3 -x "AEV3/vendor/*" "AEV3/var/*"

# Windows
tar -a -c -f AEV3.zip --exclude=AEV3/vendor --exclude=AEV3/var AEV3
```

---

## Rúbrica de Evaluación

### 1. Insuficiente (< 5) - Nivel excluyente

**Si no se cumplen TODOS estos puntos, la nota máxima es 4:**

- Ejercicio 0 completado (proyecto arranca sin errores).
- Ejercicio 1 completado (página principal con menú de navegación e imagen).
- Al menos las 3 entidades creadas correctamente con sus campos.
- Entregar el vídeo.

- Realizar mínimo 5 commits.

## 2. Suficiente (5 - 5.9)

- Cumplir todos los puntos de Insuficiente.
- Ejercicio 2 completado (CRUD de Libros y Socios).
- Ejercicio 3 parcialmente (CRUD de Préstamos con relaciones).
- Validaciones funcionando (ISBN 13 caracteres, email válido).
- Fecha `registeredAt` se establece automáticamente.
- Todas las etiquetas de formularios en castellano.

## 3. Notable (6 - 8.9)

- Cumplir todos los puntos de Suficiente.
- Ejercicio 3 completo.
- Las fechas automáticas funcionan correctamente ( `loanDate` ).
- Los desplegables muestran información legible (título del libro, nombre del socio).
- Ejercicio 4 parcialmente completado:
  - Estadísticas en página principal.
  - Detalle de socio muestra sus préstamos (activos e históricos).
  - Contador de préstamos por socio en el listado.

## 4. Excelente (9 - 10)

- Cumplir todos los puntos de Notable.
- Ejercicio 4 completamente terminado:
  - Listado de préstamos activos funcionando.
  - Botón “Marcar como devuelto” funcionando.
  - Control de disponibilidad del libro implementado.
- Datos de prueba incluidos y coherentes.
- Código limpio y bien organizado.

## 5. Extras (suman o restan puntos)

### Aspectos que suman:

- Validaciones adicionales en formularios.
- Mensajes flash de confirmación en operaciones CRUD.

- Filtros de búsqueda en algún listado.

#### Aspectos que restan:

- Errores de ejecución no controlados.
- Código desordenado.
- Commits sin sentido o todos al final.
- Vídeo no se ve o no se oye correctamente.
- Falta documentación en el README.

---

## Comandos Ejecutados

(Documenta los comandos que has utilizado en cada ejercicio)

### Ejercicio 0

### Ejercicio 1

### Ejercicio 2

### Ejercicio 3

---

## Enlace al Vídeo

(Pega aquí el enlace de ScreenPal/YouTube/OneDrive/SharePoint)

---

## Autorúbrica

Marcar con una X dónde creas que te has quedado.

Nivel	Cumplido (X)	Observaciones
Insuficiente		
Suficiente		
Notable		
Excelente		