



# SISTEMA DE BIBLIOTECA CIF

## Manual Completo del Sistema

---



### ÍNDICE

1. Descripción del Proyecto
  2. Requisitos del Sistema
  3. Instalación en Otro Equipo
  4. Estructura del Proyecto
  5. Código Fuente Completo
  6. Manual de Usuario
  7. Comandos Útiles
  8. Solución de Problemas
- 



### DESCRIPCIÓN DEL PROYECTO

Sistema de Biblioteca CIF es una aplicación web desarrollada en **PHP nativo** con **Programación Orientada a Objetos (POO)** que permite gestionar libros, usuarios y préstamos de una biblioteca institucional.

#### Tecnologías:

- PHP 7.4+
- MySQL/MariaDB
- Bootstrap 5
- HTML5/CSS3
- JavaScript

#### Características:

- Gestión de libros (CRUD completo)
- Gestión de usuarios (estudiantes, docentes, administrativos)
- Sistema de préstamos y devoluciones
- Control de disponibilidad automático
- Búsqueda y filtros

- POO con clases y métodos
  - Interfaz responsive
- 



## REQUISITOS DEL SISTEMA

- Laragon (incluye Apache, PHP, MySQL)
- PHP  $\geq 7.4$
- MySQL  $\geq 5.7$  o MariaDB  $\geq 10.3$
- Navegador web moderno

### Extensiones PHP:

- mysqli, PDO, mbstring, json
- 



## INSTALACIÓN EN OTRO EQUIPO

### Opción 1: Instalación Desde Cero

#### Paso 1: Instalar Laragon

1. Descarga desde: <https://laragon.org/download/>
2. Instala (siguiente, siguiente, finalizar)
3. Inicia Laragon
4. Clic en "Start All"

#### Paso 2: Crear el Proyecto

```
bash
```

```
# Abre Laragon → Menu → www → Open
```

```
# Crea carpeta: biblioteca-cif
```

#### Paso 3: Crear Base de Datos

1. Abre HeidiSQL desde Laragon
2. Crea nueva base de datos:

```
sql
```

```
CREATE DATABASE biblioteca_cif CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
```

3. Ejecuta el script `database.sql`:

```
sql

USE biblioteca_cif;

-- Copiar y pegar contenido de database.sql
```

## Paso 4: Copiar Archivos

Estructura a crear:

```
biblioteca-cif/
├── index.php
├── test-libro.php
├── database.sql
├── config/
│   └── conexion.php
├── modelos/
│   ├── Libro.php
│   ├── Usuario.php
│   └── Prestamo.php
├── controladores/
│   ├── LibroController.php
│   ├── UsuarioController.php
│   └── PrestamoController.php
└── vistas/
    ├── form-Libro.php
    ├── listar-libros.php
    ├── editar-libro.php
    ├── form-Usuario.php
    ├── listar-usuarios.php
    ├── form-Prestamo.php
    └── listar-prestamos.php
```

## Paso 5: Configurar Conexión

Edita `config/conexion.php`:

```
php
```

```
private $host = "localhost";  
private $usuario = "root";  
private $password = "";  
private $baseDatos = "biblioteca_cif";
```

## Paso 6: Acceder

Abre: <http://localhost/biblioteca-cif>

---

## Opción 2: Desde ZIP

**Paso 1:** Copia la carpeta a `C:\laragon\www\`

**Paso 2:** Crea la base de datos

1. Abre HeidiSQL
2. Crea database: `biblioteca_cif`
3. Importa `database.sql`

**Paso 3:** Verifica `config/conexion.php`

**Paso 4:** Accede a <http://localhost/biblioteca-cif>

---

## CÓDIGO FUENTE COMPLETO

### Script de Base de Datos

```
sql
```

-- Crear base de datos

```
CREATE DATABASE IF NOT EXISTS biblioteca_cif
CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;

USE biblioteca_cif;
```

-- Tabla de libros

```
CREATE TABLE IF NOT EXISTS libros (
  id INT AUTO_INCREMENT PRIMARY KEY,
  titulo VARCHAR(255) NOT NULL,
  autor VARCHAR(255) NOT NULL,
  categoria VARCHAR(100),
  anio INT,
  disponible BOOLEAN DEFAULT TRUE,
  fecha_registro TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  INDEX idx_titulo (titulo),
  INDEX idx_autor (autor),
  INDEX idx_disponible (disponible)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

-- Tabla de usuarios

```
CREATE TABLE IF NOT EXISTS usuarios (
  id INT AUTO_INCREMENT PRIMARY KEY,
  nombre VARCHAR(255) NOT NULL,
  documento VARCHAR(50) NOT NULL UNIQUE,
  tipo ENUM('estudiante', 'docente', 'administrativo') NOT NULL,
  fecha_registro TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  INDEX idx_documento (documento),
  INDEX idx_tipo (tipo)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

-- Tabla de préstamos

```
CREATE TABLE IF NOT EXISTS prestamos (
  id INT AUTO_INCREMENT PRIMARY KEY,
  libro_id INT NOT NULL,
  usuario_id INT NOT NULL,
  fecha_prestamo DATE NOT NULL,
  fecha_devolucion DATE,
  estado ENUM('activo', 'devuelto') DEFAULT 'activo',
  fecha_registro TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  FOREIGN KEY (libro_id) REFERENCES libros(id) ON DELETE RESTRICT,
  FOREIGN KEY (usuario_id) REFERENCES usuarios(id) ON DELETE RESTRICT,
  INDEX idx_estado (estado),
```

```
INDEX idx_fecha_prestamo (fecha_prestamo)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

*-- Datos de ejemplo*

```
INSERT INTO libros (titulo, autor, categoria, anio) VALUES
('Criminalística Forense', 'Dra. Ana Torres', 'Investigación', 2021),
('Introducción a la Programación', 'Carlos Mendoza', 'Informática', 2020),
('Base de Datos Relacionales', 'María González', 'Informática', 2019),
('Química Forense', 'Dr. Roberto Silva', 'Ciencias', 2022),
('Análisis de Evidencias', 'Lic. Patricia Ruiz', 'Investigación', 2023);
```

```
INSERT INTO usuarios (nombre, documento, tipo) VALUES
('Juan Pérez', '12345678', 'estudiante'),
('María García', '87654321', 'docente'),
('Carlos López', '11223344', 'administrativo');
```

## Clase Conexion (config/conexion.php)

php

```

<?php
class Conexion {
    private $host = "localhost";
    private $usuario = "root";
    private $password = "";
    private $baseDatos = "biblioteca_cif";
    private $conexion;

    public function conectar() {
        try {
            $this->conexion = new mysqli(
                $this->host,
                $this->usuario,
                $this->password,
                $this->baseDatos
            );

            if ($this->conexion->connect_error) {
                throw new Exception("Error de conexión: " . $this->conexion->connect_error);
            }

            $this->conexion->set_charset("utf8mb4");
            return $this->conexion;

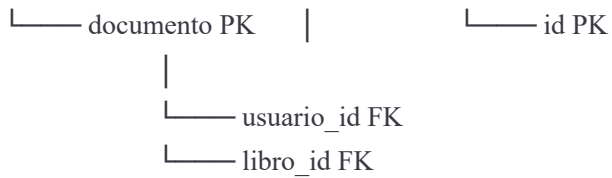
        } catch (Exception $e) {
            die("Error: " . $e->getMessage());
        }
    }

    public function cerrar() {
        if ($this->conexion) {
            $this->conexion->close();
        }
    }
}
?>

```

## Esquema de Relaciones

USUARIOS (1) — (N) PRESTAMOS (N) — (1) LIBROS  
 |                      |                      |  
 |                      |                      |



## MANUAL DE USUARIO

### Acceso al Sistema

1. Inicia Laragon → Start All
2. Abre navegador
3. Ve a: <http://localhost/biblioteca-cif>

### Página Principal

Verás 2 opciones:

- Registrar Nuevo Libro
- Ver Todos los Libros

### Módulo de Libros

#### Registrar Libro:

1. Clic en "Registrar Nuevo Libro"
2. Llena el formulario:
  - **Título** (obligatorio)
  - **Autor** (obligatorio)
  - **Categoría** (opcional)
  - **Año** (opcional)
3. Clic en "Guardar"

#### Ver Libros:

1. Clic en "Ver Todos los Libros"
2. Verás tabla con:
  - ID
  - Título
  - Autor



- Categoría
- Año
- Estado (Disponible/Prestado)
- Acciones (Editar, Eliminar)

### **Editar Libro:**

1. En lista de libros, clic en "Editar"
2. Modifica campos
3. Clic en "Guardar Cambios"

### **Eliminar Libro:**

1. En lista, clic en "Eliminar"
2. Confirma la acción

## **Módulo de Usuarios**

### **Registrar Usuario:**

1. Ir a "Registrar Usuario"
2. Llena:
  - **Nombre** (obligatorio)
  - **Documento** (obligatorio, único)
  - **Tipo** (estudiante/docente/administrativo)
3. Guardar

### **Ver Usuarios:**

Lista de todos los usuarios con:

- Nombre
- Documento
- Tipo
- Fecha de registro

## **Módulo de Préstamos**

### **Registrar Préstamo:**

1. Ir a "Nuevo Préstamo"
2. Seleccionar:
  - **Libro** (solo disponibles)
  - **Usuario**
  - **Fecha de préstamo**
3. Guardar

**Automático:** El libro cambia a "No disponible"

### **Registrar Devolución:**

1. En préstamos activos, clic en "Devolver"
2. Se registra fecha de devolución
3. **Automático:** Libro vuelve a "Disponible"

### **Ver Préstamos:**

Lista con:

- Título del libro
  - Usuario
  - Fecha de préstamo
  - Fecha de devolución
  - Estado (Activo/Devuelto)
- 

## **COMANDOS ÚTILES**

### **MySQL/MariaDB**

```
sql
```

```
-- Ver bases de datos
SHOW DATABASES;

-- Usar base de datos
USE biblioteca_cif;

-- Ver tablas
SHOW TABLES;

-- Ver estructura
DESCRIBE libros;

-- Ver datos
SELECT * FROM libros;
SELECT * FROM libros WHERE disponible = 1;
SELECT * FROM prestamos WHERE estado = 'activo';

-- Limpiar tabla (¡CUIDADO!)
TRUNCATE TABLE prestamos;
```

## Laragon

- **Iniciar:** Clic en "Start All"
- **Detener:** Clic en "Stop All"
- **Abrir www:** Menu → www → Open
- **HeidiSQL:** Menu → HeidiSQL
- **Logs:** Menu → Apache → httpd.conf



## SOLUCIÓN DE PROBLEMAS

### Error de Conexión

**Síntoma:** "Error de conexión"

**Solución:**

1. Verifica MySQL corriendo en Laragon
2. Revisa credenciales en `config/conexion.php`
3. Confirma que existe database `biblioteca_cif`

## Tabla no Existe

**Síntoma:** "Table doesn't exist"

**Solución:**

1. Abre HeidiSQL
2. Selecciona `biblioteca_cif`
3. Ejecuta `database.sql` nuevamente

## Préstamo no se Registra

**Causas:**

- Libro no disponible
- Usuario no existe
- Llaves foráneas incorrectas

**Solución:** Verifica en HeidiSQL:

```
sql

SELECT * FROM libros WHERE disponible = 1;
SELECT * FROM usuarios;
```

## Bootstrap no Carga

**Causa:** No hay internet (Bootstrap desde CDN)

**Solución:** Descarga Bootstrap localmente o verifica conexión

## Error Parse en PHP

**Causa:** Sintaxis incorrecta

**Solución:** Verifica:

- Etiquetas PHP completas `<?php ... ?>`
  - Punto y coma al final de líneas
  - Paréntesis balanceados
-

# CARACTERÍSTICAS POO

## Clase Libro

### Propiedades:

- id, titulo, autor, categoria, anio, disponible

### Métodos:

- `validar()` - Valida datos obligatorios
- `guardar()` - Inserta en DB
- `actualizar()` - Actualiza registro
- `eliminar()` - Borra libro
- `listarTodos()` - Lista todos
- `buscarPorId()` - Busca por ID
- `buscar()` - Búsqueda general
- `mostrar()` - Retorna string con info

## Clase Usuario

### Propiedades:

- id, nombre, documento, tipo

### Métodos:

- `validar()`, `guardar()`, `actualizar()`, `eliminar()`, `listarTodos()`, `buscarPorId()`

## Clase Prestamo

### Propiedades:

- id, libroId, usuarioId, fechaPrestamo, fechaDevolucion, estado

### Métodos:

- `registrar()` - Crea préstamo y marca libro no disponible
  - `devolver()` - Registra devolución y marca libro disponible
  - `listarTodos()` - Lista todos los préstamos
  - `listarActivos()` - Solo préstamos activos
-

## RESPONSIVE

Bootstrap 5 hace el sistema responsive:

- Desktop: Tablas completas
  - Tablet: Cards apiladas
  - Móvil: Vista simplificada
- 

## PERSONALIZACIÓN

### Cambiar Colores

Edita en archivos `.php`:

```
css

/* De azul a verde */
.navbar-dark.bg-primary {
    background: #28a745 !important;
}
```

### Agregar Categorías

Edita el SELECT en formulario:

```
html

<option value="Nueva Categoría">Nueva Categoría</option>
```

### Cambiar Items por Página

No aplica (muestra todos)

Para implementar paginación, agregar LIMIT en SQL:

```
php

$sql = "SELECT * FROM libros LIMIT 10 OFFSET 0";
```

---

## VALIDACIONES

### Libros:

- Título: Obligatorio
- Autor: Obligatorio
- Categoría: Opcional
- Año: Opcional

### Usuarios:

- Nombre: Obligatorio
- Documento: Obligatorio, único
- Tipo: Obligatorio (enum)

### Préstamos:

- Libro: Debe estar disponible
  - Usuario: Debe existir
  - Fecha: Obligatoria
- 

## SEGURIDAD

### Implementado:

- Prepared statements (previene SQL injection)
- Charset UTF-8
- Validación de datos

### Recomendado Agregar:

- Sistema de login
  - Roles y permisos
  - Sanitización de inputs
  - HTTPS en producción
  - Backup automático de DB
- 

## MEJORAS FUTURAS

## 1. Sistema de Login

- Autenticación
- Sesiones
- Roles (admin, bibliotecario)

## 2. Multas

- Cálculo por días de retraso
- Registro de pagos

## 3. Reservas

- Reservar libros prestados
- Notificaciones

## 4. Estadísticas

- Libros más prestados
- Usuarios más activos
- Gráficos

## 5. Búsqueda Avanzada

- Filtros múltiples
- Autocompletado

## 6. Código de Barras

- Escaneo rápido
- Impresión de etiquetas

---

### Sistema de Biblioteca CIF

**Versión:** 1.0

**Tecnología:** PHP Nativo + MySQL

**Fecha:** Octubre 2025