

**Actividad evaluativa - Eje 2**

**Presentado por:**

Forero Torres Josep Alejandro

Hernandez Ulloa Yisel Patricia

Martínez González Ximena

Miranda García Carlos Julián

Rueda Lievano Nicolas

**Fundación Universitaria Área Andina**  
**Facultad de Ciencias Básicas e Ingeniería**

**Docente**

Oscar Fabian Campos

Porras

Arquitectura de Software

Bogotá - Modalidad Virtual  
31 de agosto de 2025

## **Índice**

<b>Introducción</b>	<b>3</b>
<b>Objetivos</b>	<b>4</b>
Objetivo General	4
Objetivos Específicos.	4
<b>Desarrollo de la actividad</b>	<b>5</b>

## **Introducción**

Este proyecto se enfoca en el desarrollo de una aplicación de software que incorpora patrones de diseño y estilos arquitectónicos, como MVC, DTO y DAO, junto con Programación Orientada a Objetos. La aplicación se conecta a una base de datos y permite realizar operaciones CRUD. Se destaca la importancia de los patrones de diseño en la estructura y mantenimiento del código. La elección del lenguaje de programación es flexible. El proyecto busca demostrar cómo aplicar estos principios en la creación de aplicaciones eficientes y sostenibles.

## Objetivos

### Objetivo General

El desarrollo de la actividad consiste en implementar patrones de diseño para el desarrollo de aplicaciones de software.

### Objetivos Específicos.

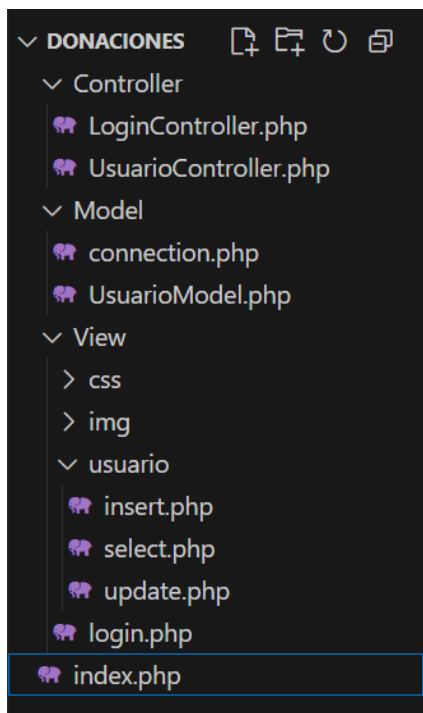
- Implementar una aplicación que utilice uno o varios patrones de diseño o estilos arquitectónicos, tales como **MVC, DTO o DAO**, con el fin de garantizar una estructura organizada y escalable.
- Establecer una **conexión a una base de datos** que permita el almacenamiento y la gestión eficiente de la información.
- Integrar los principios de la **Programación Orientada a Objetos (POO)** en el desarrollo de la aplicación, asegurando modularidad, reutilización y mantenibilidad del código.
- Desarrollar las operaciones básicas de **CRUD (Crear, Leer, Actualizar y Eliminar)** para la gestión de registros dentro de la aplicación.
- Construir la aplicación en un **lenguaje de programación adecuado** (PHP, JS, MySQL, entre otros) que facilite la implementación de los requerimientos establecidos.

## Desarrollo de la actividad

El desarrollo de la actividad consiste en implementar patrones de diseño para el desarrollo de aplicaciones de software, de acuerdo con las siguientes instrucciones.

**1. Construir una aplicación que implemente uno o varios patrones de diseño o estilos arquitectónicos, por ejemplo (MVC, DTO, DAO):**

Se eligió la arquitectura **MVC** (Modelo-Vista-Controlador), implementada utilizando el lenguaje de programación **PHP**.



## 2. La aplicación debe tener una conexión a una base de datos.

Se eligió como motor de base de datos **MySQL**:



```

1  <?php
2  class Connection {
3      private static $host = "localhost";
4      private static $user = "root";
5      private static $pass = "";
6      private static $db = "bd_donaciones";
7      private static $conn;
8
9      public static function getConnection(): mixed {
10         if (!isset(self::$conn)) {
11             self::$conn = new mysqli(hostname: self::$host, username: self::$user, password: self::$pass, database: self::$db);
12
13             if (self::$conn->connect_error) {
14                 die("Error en la conexión: " . self::$conn->connect_error);
15             }
16         }
17         return self::$conn;
18     }
19 }
20 ?>
    
```

## 3. Integrar en la aplicación la Programación Orientada a Objetos y debe permitir las operaciones básicas para el manejo de registros (CRUD).

A continuación, se adjunta el enlace al repositorio, donde es posible visualizar la arquitectura del proyecto junto con su respectivo código fuente, también adjuntamos un video de la demostración de la funcionalidad:

Repositorio alojado en GitHub: [Repositorio](#)

Video de la funcionalidad:  [CRUD.mkv](#)

## Conclusiones

Durante el desarrollo de esta aplicación de software, hemos vivido la relevancia de utilizar patrones de diseño y principios arquitectónicos robustos. La implementación del patron MVC ha sido clave para establecer un código modular, claro y fácilmente manejable. La Programación Orientada a Objetos (POO) ha servido como un pilar firme para la fabricación de componentes que se pueden reutilizar y una estructura que puede expandirse.

La vinculación con una base de datos y la realización de operaciones CRUD nos ha permitido entender la administración de datos de forma eficaz, lo cual es crucial en diversas aplicaciones. Además, hemos reconocido la versatilidad que proporciona la selección del lenguaje de programación, ajustándose a las necesidades particulares del proyecto.

## Bibliografía

PHP Group. (2024). PHP Manual. *Recuperado de* <https://www.php.net/>

Oracle. (2024). *Oracle Database Security Guide*. Oracle Help Center.  
<https://docs.oracle.com/en/database/>

Arquitectura de software. (2025). *Referente pensamiento eje 1*  
<https://areandina.instructure.com/courses/64475>

Arquitectura de software. (2025). *Referente pensamiento eje 2*  
<https://areandina.instructure.com/courses/64475>

Cómo hacer un CRUD con PHP y MySQL. (2022). *Video de youtube*  
<https://www.youtube.com/watch?v=sYaEoNy5OGs&t=1662s>

Pablo García. (2025). *Arquitectura MVC: El Patrón de Diseño para Aplicaciones Web*.  
Recuperado de [pablogarciajc.com](http://pablogarciajc.com).