

## Documentación del Proyecto

### 1. Descripción del Proyecto

Este proyecto es un sistema de gestión de ofertas desarrollado con **PHP** en el backend y **Vue.js** en el frontend. Permite la creación, visualización y descarga de ofertas en formato Excel.

### 2. Arquitectura y Tecnologías Utilizadas

#### Tecnologías

- **Backend:** PHP (Versión 7 o superior)
- **Base de Datos:** MySQL
- **Frontend:** HTML, CSS, JavaScript, Vue.js
- **Servidor Web:** Apache o cualquier servidor compatible con PHP

#### Arquitectura

El sistema sigue una arquitectura cliente-servidor:

- **Frontend:** Se encarga de la interfaz de usuario con Vue.js.
- **Backend:** Provee endpoints API con PHP para gestionar las ofertas.
- **Base de datos:** Almacena las ofertas y sus detalles.

### 3. Estructura del Código

/config # Contiene los archivos de configuración y APIs

└─ database.php # Archivo de conexión a la base de datos

└─ api.php # API para la gestión de ofertas

└─ exportar\_excel.php # API para exportar ofertas en Excel

/frontend # Contiene el código del frontend

└─ index.html # Página principal (Lista de Ofertas)

└─ crear.html # Página para crear una oferta

└─ descargar.html # Página para descargar las ofertas

└─ js/main.js # Archivo Vue.js para manejar la lógica del frontend

/bd      # Contiene el script de la base de datos

README.md    # Documentación principal del proyecto

#### **4. Endpoints de la API**

##### **Obtener todas las ofertas**

**GET** /config/api.php

##### **Respuesta:**

```
[  
  {  
    "id": 1,  
    "objeto": "Oferta 1",  
    "descripcion": "Descripción de la oferta",  
    "moneda": "USD",  
    "presupuesto": 1000,  
    "fecha_inicio": "2024-04-01",  
    "fecha_cierre": "2024-04-10",  
    "estado": "Activo"  
  }  
]
```

##### **Crear una nueva oferta**

**POST** /config/api.php

##### **Cuerpo de la solicitud:**

```
{  
  "objeto": "Nueva oferta",  
  "descripcion": "Descripción de la nueva oferta",  
  "moneda": "USD",
```

```
"presupuesto": 2000,  
"actividad": "Construcción",  
"fecha_inicio": "2024-04-05",  
"fecha_cierre": "2024-04-20"  
}
```

**Respuesta:**

```
{  
  "message": "Oferta creada",  
  "id": 2  
}
```

**5. Buenas Prácticas Implementadas**

- **Uso de Vue.js en el frontend**, separando la lógica del cliente y el servidor.
- **Código modular**, con archivos separados para cada función.
- **Convenciones de nomenclatura y estructura de carpetas bien definida**