1. Descripción General

Este backend simula el servidor de una página web de consultoría y desarrollo de software. Se implementa una API RESTful en PHP puro, respetando buenas prácticas, con simulación de base de datos mediante arreglos.

Se atienden solicitudes HTTP (GET y POST) para entregar información de servicios, "nosotros", y recibir formularios de contacto.

2. Tecnologías Utilizadas

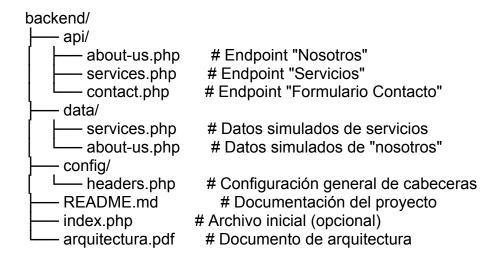
Lenguaje: PHP 8.x

• **Servidor local**: XAMPP (Apache)

Formato de datos: JSON

Simulación de base de datos: Arreglos PHP

3. Estructura de Carpetas y Archivos



4. Descripción de Cada Componente

4.1. config/headers.php

Configura las cabeceras HTTP para:

- Permitir CORS (origen cruzado)
- Definir que las respuestas sean JSON
- Permitir métodos GET, POST y OPTIONS

4.2. data/services.php

Simula los servicios ofrecidos como un arreglo de PHP. Cada servicio contiene:

- ID
- Nombre
- Descripción

4.3. data/about-us.php

Simula la información de la empresa:

- Misión
- Visión
- Valores

4.4. api/services.php

- Método permitido: GET
- Devuelve la lista de servicios.
- Simula una respuesta de base de datos.

4.5. api/about-us.php

• Método permitido: **GET**

Devuelve información de "Nosotros".

4.6. api/contact.php

- Método permitido: POST
- Recibe nombre, email y mensaje del formulario de contacto.
- Simula el guardado de datos (sin BD real).
- Valida que no falte ningún campo.

5. Flujo de Funcionamiento

- 1. El navegador (frontend) hace una solicitud HTTP (GET o POST).
- 2. Apache redirige la solicitud al archivo PHP correspondiente en /api/.
- 3. Se cargan las cabeceras configuradas.
- 4. Se accede a los datos simulados en /data/ si corresponde.
- 5. PHP procesa la solicitud y devuelve una respuesta JSON.
- 6. El navegador recibe la respuesta y la muestra en el frontend.

6. Endpoints Implementados

| Endpoint | Métod o | Descripción |
|-----------------------|------------|-----------------------------------|
| /api/service s.php | GET | Devuelve lista de servicios |
| /api/about-u s.php | GET | Devuelve misión, visión y valores |
| /api/contact .php | POST | Recibe formulario de contacto |

7. Seguridad y Validaciones

- CORS habilitado solo para propósito de pruebas.
- Validación de métodos HTTP en cada endpoint.
- Control de errores cuando faltan datos obligatorios.