# Informe de Artefactos de Código

## Introducción

El presente informe documenta los artefactos de código del software Pet Services, una aplicación diseñada para la gestión de tiendas veterinarias. Se describen los módulos de la aplicación web desarrollada, la estructura del código, los lenguajes y frameworks utilizados, así como las configuraciones necesarias para su correcto funcionamiento. Además, se evalúan las posibilidades de adaptación para versiones de escritorio y móvil.

El código desarrollado está estructurado en componentes frontend y backend, con una base de datos MySQL administrada mediante phpMyAdmin. Se utilizaron tecnologías web como HTML, CSS, JavaScript (con Node.js y Express en el backend) y herramientas de control de versiones como GitHub.

## 1. Estructura de Artefactos de Código

### 1.1 Aplicación Web

La aplicación Pet Services está diseñada como una plataforma web accesible desde navegadores, desarrollada con las siguientes tecnologías:

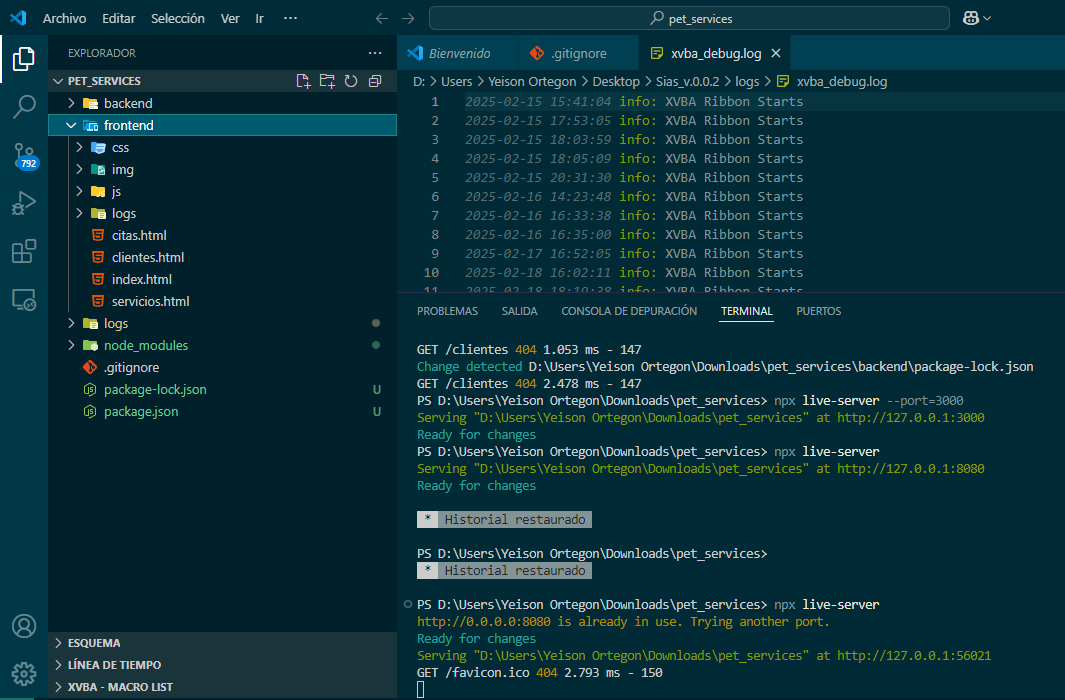
**Frontend:**

HTML5, CSS3, JavaScript

Frameworks y Librerías: Bootstrap, jQuery

**Estructura:**

index.html (Página de inicio)



clientes.html (Gestión de clientes)

servicios.html (Gestión de servicios)

citas.html (Gestión de citas)

Carpeta css/ para los estilos

Carpeta js/ para la funcionalidad

**Backend:**

Lenguaje y Framework: Node.js con Express

Base de Datos: MySQL, administrado con phpMyAdmin

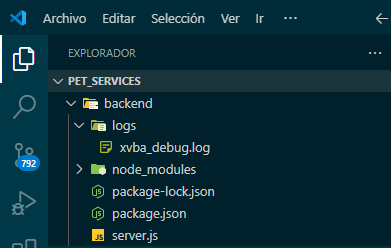
Estructura del Backend:

server.js (Configuración del servidor)

Carpeta routes/ (Rutas para API REST)

Carpeta controllers/ (Lógica de negocio)

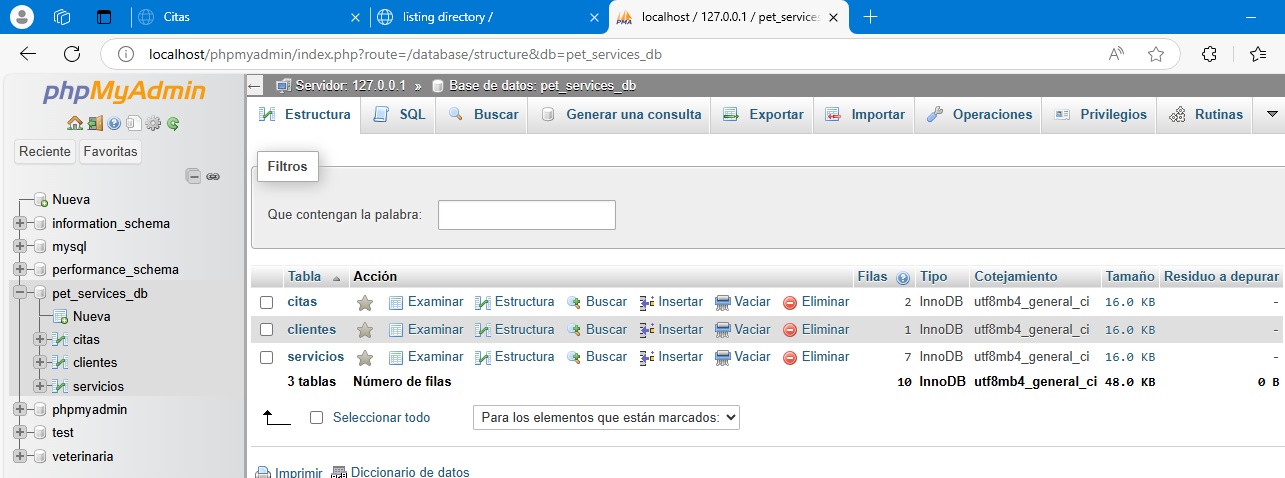
Carpeta models/ (Definición de la base de datos)



**Base de Datos (MySQL)**

Tablas: clientes, servicios, citas

Relacionamiento entre tablas (JOIN para consultas)

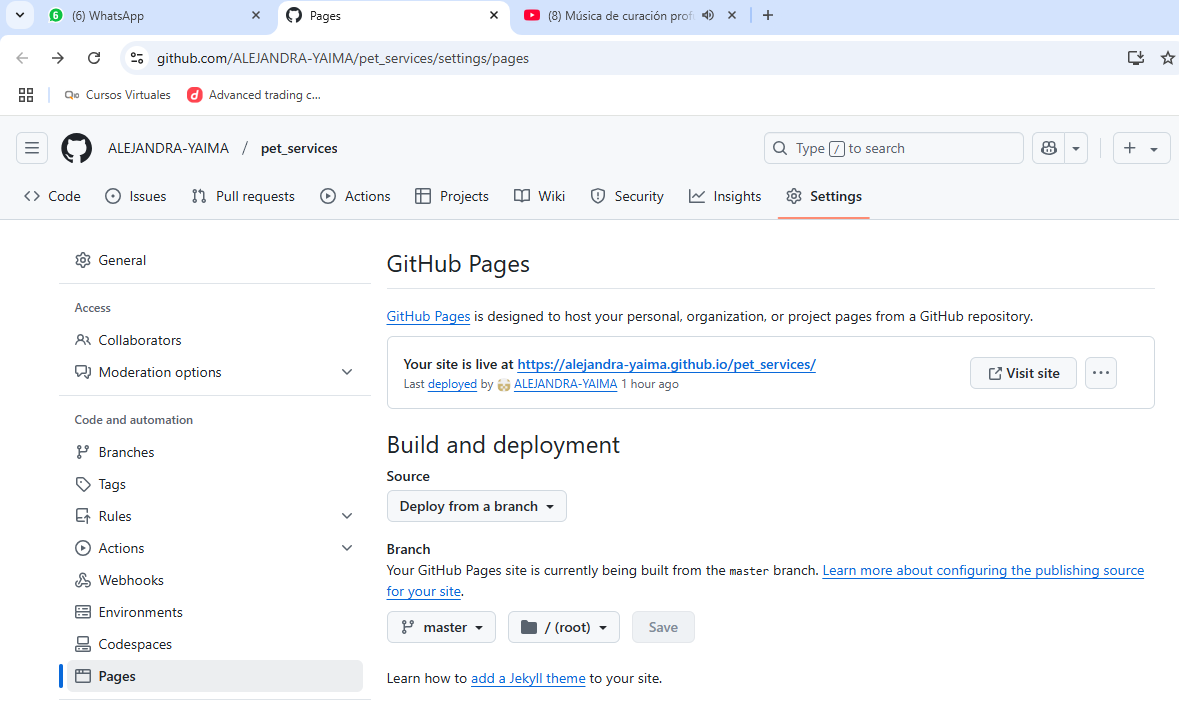


**Control de Versiones y Hosting**

Repositorio en GitHub

Despliegue del Frontend en GitHub Pages

Backend alojado en un servidor con Node.js



### 1.2 Aplicación Escritorio (Opcional)

Para adaptar la aplicación web a escritorio se podrían usar tecnologías como:

Electron.js: Para empaquetar la app web como un software ejecutable en Windows/Mac/Linux.

SQLite o MySQL como base de datos local.

En caso de ser necesaria una versión de escritorio, la estructura del código sería reutilizable con ajustes en la interfaz gráfica.

### 1.3 Aplicación Móvil (Opcional)

Para desarrollar una versión móvil, se podrían considerar las siguientes opciones:

React Native: Para crear una aplicación híbrida que funcione en Android e iOS.

Flutter: Para una mejor experiencia de usuario en dispositivos móviles.

PWA (Progressive Web App): Para que la versión web sea accesible como aplicación móvil.

La aplicación web actual podría adaptarse a móvil con un diseño responsive, y en el futuro integrarse con APIs para funciones avanzadas.

## 2. Configuración y Ejecución del Código

Para ejecutar el software, se deben seguir estos pasos:

**Frontend**

1. Clonar el repositorio de GitHub.

2. Abrir index.html en un navegador.

**Backend**

1. Instalar Node.js y MySQL.

2. Configurar la base de datos en config/db.js.

3. Ejecutar:

npm install

node server.js

## 3. Conclusiones

El código de Pet Services está estructurado para una aplicación web funcional, con posibilidad de adaptación a escritorio o móvil. El modularidad del código permite la escalabilidad del sistema y facilita futuras mejoras.