# PROPUESTA TÉCNICA DEL SOFTWARE "PET SERVICES"

## 1. INTRODUCCIÓN

El software **Pet Services** es una aplicación web diseñada para gestionar los servicios de una tienda veterinaria, permitiendo la administración de clientes, servicios y citas. Esta propuesta técnica describe las tecnologías utilizadas, la arquitectura del sistema y los componentes que conforman la aplicación, asegurando su funcionalidad, escalabilidad y facilidad de uso.

## 2. OBJETIVO DEL SOFTWARE

Desarrollar una plataforma eficiente para la gestión de clientes, servicios veterinarios y estéticos, y citas, proporcionando una interfaz amigable y un backend robusto que permita el manejo seguro de la información.

## 3. TECNOLOGÍAS UTILIZADAS

|  |  |
| --- | --- |
| ****Tecnología**** | ****Descripción y Uso**** |
| ****Frontend**** | HTML, CSS y JavaScript para la interfaz de usuario |
| ****Framework Backend**** | Node.js con Express.js para la gestión del servidor |
| ****Base de Datos**** | MySQL para almacenamiento estructurado de datos |
| ****Gestor de Base de Datos**** | phpMyAdmin para administración de MySQL |
| ****Control de Versiones**** | GitHub para gestión del código fuente y control de cambios |
| ****Hosting Frontend**** | GitHub Pages para el despliegue del frontend |
| ****Hosting Backend**** | Servidor con soporte para Node.js y MySQL |
| ****Entorno de Desarrollo**** | Visual Studio Code para programación y depuración |
| ****Servidor Local**** | XAMPP para pruebas en entorno local |

## 4. ARQUITECTURA DEL SISTEMA

La arquitectura del sistema sigue el modelo **Cliente-Servidor**, con una separación clara entre el frontend, el backend y la base de datos:

* **Frontend**: Interfaz de usuario responsiva desarrollada en HTML, CSS y JavaScript.
* **Backend**: API REST desarrollada en Node.js con Express.js.
* **Base de Datos**: MySQL con tablas normalizadas para clientes, servicios y citas.
* **Control de Acceso**: Middleware en Express.js para validaciones y gestión de datos.

## 5. COMPONENTES PRINCIPALES

### 5.1. Gestión de Clientes

* Registro de clientes con datos básicos (nombre, teléfono, mascota).
* Consulta y actualización de información.

### 5.2. Gestión de Servicios

* Registro de servicios veterinarios y estéticos.
* Listado y edición de servicios.

### 5.3. Gestión de Citas

* Agendamiento de citas según disponibilidad de servicio y mascota.
* Consulta y cancelación de citas.

## 6. SEGURIDAD Y VALIDACIONES

* **Autenticación y Autorización**: Uso de tokens JWT para la autenticación de usuarios (opcional en futuras versiones).
* **Protección contra Inyecciones SQL**: Uso de consultas preparadas en MySQL.
* **Validaciones en Backend y Frontend**: Sanitización de entradas para evitar ataques de seguridad.

## 7. PRUEBAS Y DEPURACIÓN

* **Pruebas Unitarias**: Validación de módulos individuales del backend.
* **Pruebas Funcionales**: Verificación de funcionalidades en el frontend.
* **Pruebas de Integración**: Evaluación del flujo de datos entre frontend, backend y base de datos.

## 8. PLAN DE IMPLEMENTACIÓN

1. **Despliegue en Servidor Local** para pruebas iniciales.
2. **Configuración del Hosting Backend** con Node.js y MySQL.
3. **Publicación del Frontend en GitHub Pages**.
4. **Pruebas en Producción** con datos de prueba.

## 9. CONCLUSIONES

El desarrollo de **Pet Services** se realizó con tecnologías modernas que garantizan un sistema estable, escalable y seguro. La separación de responsabilidades entre frontend, backend y base de datos permite una gestión eficiente del proyecto. Su implementación en GitHub Pages y un hosting para el backend facilita el despliegue y mantenimiento del software.