

INTRODUCCIÓN:

1. Información General del Proyecto

- **Nombre del Proyecto:** "PeluDog Manager"
- **Objetivo:** Desarrollar una página web funcional que permita gestionar las citas de clientes, los servicios de peluquería canina, la administración de los pagos y la presentación de la peluquería.

TECNOLOGÍAS EMPLEADAS:

- **PHP/JavaScript/HTML/CSS:** Todo el código del proyecto esta basado en estos cuatro lenguajes, PHP se ha usado más para el back mientras que los demás forman parte del front.
- **GitHub:** Se ha usado para ir subiendo las modificaciones a sus respectivas ramas para así llevar una mejor organización y poder trabajar en equipo sin solaparse y llevar un registro de todas las modificaciones.
- **ChatGpt/Copilot:** Utilizamos la IA para la creación de algunas de las imágenes usadas en la página web ya que estas no disponen de derechos de imagen.
- **SQL:** Lenguaje usado para la comunicación con la base de datos
- **XAMPP:** Utilizado para gestionar la base de datos a través de MariaDB y proporcionar el PHP
- **VisualStudio:** Es el editor de texto usado para desarrollar todo el proyecto.
- **Bootstrap:** Se ha utilizado esta librería de CSS.

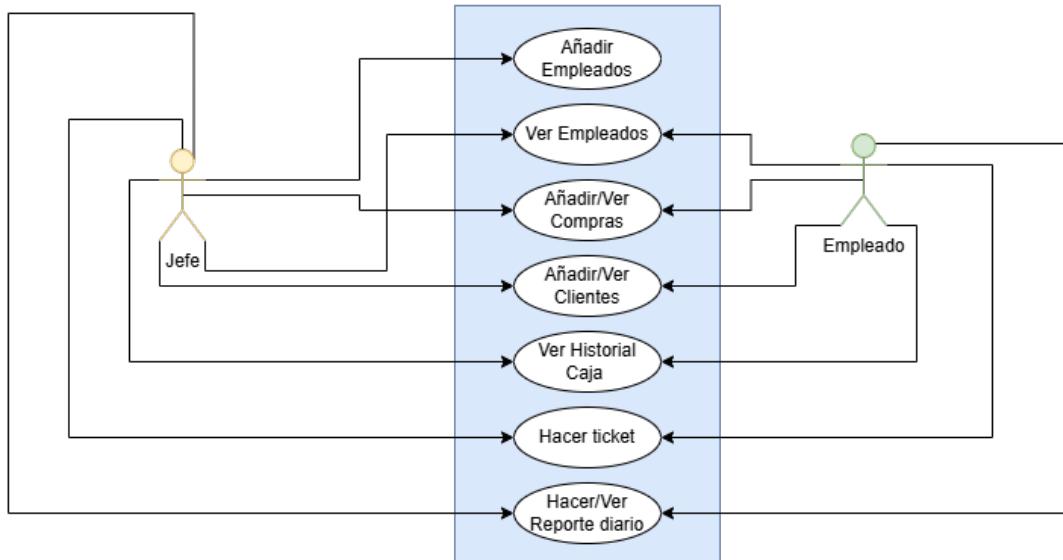
Decidí emplear estas tecnologías debido a por un lado, la gran experiencia que tengo con ellas y por otro lado a que las veía las más apropiadas para el tipo de página que iba a desarrollar.

ARQUITECTURA DEL PROYECTO:

En cuanto a la arquitectura de este proyecto me decante por el conocido MVC (Modelo-Vista-Controlador), debido principalmente a su gran capacidad para organizar el proyecto de manera clara y eficiente. Este tipo de arquitectura clara y sencilla facilita mucho el mantenimiento y la reutilización de código. También cabe destacar que un motivo bastante importante es que muchos frameworks lo soportan lo que me facilitaría en muy gran medida la implementación de estos.

ANALISIS DE REQUISITOS:

DIAGRAMA DE CASOS DE USO:



REQUISITOS:

Requisitos funcionales:

Funcionalidades de los Usuarios:

1. Registro de Usuarios:
 - a. Los clientes pueden crear una cuenta con sus datos personales (nombre, teléfono, dirección de correo, peluquería canina)
2. Gestión de Citas:
 - a. Capacidad de ver todas las citas programadas y poder gestionarlas
 - b. Visualizar el calendario de reservas de la peluquería (futuro).
3. Gestión de empleados:
 - a. Acceso a los empleados de tu peluquería
 - b. El jefe o administrador de esa peluquería tendrá la capacidad para añadir, modificar o eliminar empleados
4. Caja e inventario:
 - a. Posibilidad de hacer la caja de ese día
 - b. Sacar un ticket instantáneo para entregar al cliente
 - c. Posibilidad de controlar el inventario/stock de los productos disponibles en la peluquería

Funcionalidades adicionales:

- 1.Posibilidad de hacer una consulta en caso de que no tengas contratada la web para solicitar un presupuesto

Requisitos no funcionales

- 1.Diseño responsive:

- a.La página web se verá bien en cualquier dispositivo

- 2.Accesibilidad:

- a.La página debe cumplir con las normas básicas de accesibilidad web (WCAG 2.1)

- 3.Seguridad:

- a.Protección de datos sensibles para los usuarios como pueden ser las contraseñas

- 4.Velocidad:

- a.La página web debe ser rápida y dinámica intentando tener un tiempo de carga de menos de 3 segundos

- 5.Escalabilidad:

- a.La página web debe poder ampliarse en un futuro permitiendo un número amplio tanto de clientes como empleados para las peluquerías

Requisitos técnicos:

- 1.Plataforma:

- a.Desarrollo en una plataforma web estándar (HTML, CSS, JavaScript)
 - b.Backend en PHP y la base de datos en MySQL

- 2.Mantenimiento y actualizaciones:

- a.La web debe permitir actualizaciones.

Requisitos de Implementación:

- 1.Plazo de Entrega:

- a.La página web debe estar lista para su lanzamiento en dos meses

- 2.Pruebas:

- a.La página debe ser sometida a pruebas de usabilidad, pruebas de carga y pruebas de seguridad antes de su lanzamiento.

CICLO DE VIDA :

Para este proyecto, he seleccionado el **Ciclo de Vida Ágil**, ya que es el más adecuado para proyectos de desarrollo web debido a su flexibilidad, capacidad de adaptación y a cambios. En ciclo, el trabajo se divide en ciclos cortos llamados **sprints**, que permiten entregas parciales y

feedback constante del cliente, lo cual es ideal para un proyecto como este en el cual la idea principal es adaptarse al cliente.

Sprint 0: Planificación y Definición de Requisitos Iniciales

Objetivo: Establecer los objetivos, los requisitos iniciales y los recursos para el proyecto.

Es el inicio del proyecto y en esta fase no se desarrolla código, pero se establece la base del trabajo.

Contenido:

- Reuniones con los dueños de las peluquerías interesadas para entender las necesidades.
- Definir los requisitos iniciales del sitio web (funcionalidades esenciales como reservas, pagos, servicios, etc.).
- Crear una lista de funcionalidades que vamos a añadir.
- Establecer el equipo de trabajo y asignar roles.

Duración estimada: 1 semana.

Sprint 1 : Diseño y Prototipo Inicial

Objetivo: Crear un diseño visual básico y un prototipo funcional de la página web para que el cliente le de el visto bueno.

Se enfoca en crear el diseño básico y comenzar con la estructura visual de la página.

Contenido:

- Creación de prototipos interactivos para las páginas principales del sitio (página de inicio, servicios, reserva de citas, contacto, etc.).
- Presentación de las primeras ideas visuales al cliente .
- Validación de la estructura del sitio y los elementos de diseño (colores, tipografía, imágenes, logotipos).

Duración estimada: 2 semanas.

Sprint 2: Desarrollo del Backend y Frontend Básico

Objetivo: Comenzar con el desarrollo de la funcionalidad básica de la página web, incluyendo el backend (gestión de clientes, gestión de compras , base de datos) y el frontend (interfaz de usuario inicial).

Este sprint se centra en la creación de las funcionalidades fundamentales del sitio. Se dejará espacio para la corrección de errores y ajustes que vayan surgiendo.

Contenido:

- Desarrollo de la estructura básica del sitio web: creación de la base de datos y desarrollo de las primeras funcionalidades del backend (gestión de clientes, gestión de compras , base de datos).
- Desarrollo de la interfaz del usuario para las páginas principales del sitio.
- Implementación de un sistema de caja básico

Duración estimada: 3 semanas.

Sprint 3: Implementación y Revisión de Funcionalidades

Objetivo: Ampliar las funcionalidades del sitio web, realizar pruebas y ajustes según el feedback recibido.

En este sprint se realizan los ajustes necesarios según el feedback del cliente, incluyendo la corrección de errores y la mejora de la experiencia de usuario.

Contenido:

- Añadir más funcionalidades (por ejemplo, gestión de historial de citas, creación de perfil de empleado, creación del perfil de cliente).
- Realizar pruebas de integración entre frontend y backend.
- Revisión continua con el cliente para validar avances.

Duración estimada: 3 semanas.

Sprint 4: Pruebas de Usabilidad y Seguridad

Objetivo: Realizar pruebas de funcionalidad, seguridad, y usabilidad para garantizar que el sitio sea estable, seguro y fácil de usar.

Esta fase es esencial para garantizar que el sitio esté libre de errores y sea seguro antes del lanzamiento.

Contenido:

- Pruebas de usabilidad: verificación de la navegación y experiencia del usuario.
- Pruebas de seguridad: asegurar que las contraseñas u otros datos sensibles y la base de datos estén protegidos.
- Pruebas de carga para asegurar que el sitio pueda manejar múltiples empleados y clientes.
- Validación final con el cliente sobre el funcionamiento general.

Duración estimada: 2 semanas.

Sprint 5: Lanzamiento y Despliegue

Objetivo: Lanzar el sitio web en producción y realizar ajustes post-lanzamiento según sea necesario.

En esta fase, el sitio web se hace disponible para los clientes, pero se debe seguir monitoreando de cerca por si surgen problemas.

Contenido:

- Despliegue del sitio web en el servidor/equipo local de cada cliente.
- Asegurar que todas las funcionalidades estén operativas y que el sitio sea perfectamente accesible
- Capacitación al equipo administrativo sobre cómo gestionar el sistema (reservas, caja, clientes).
- Monitoreo de la web durante el lanzamiento para detectar posibles errores y poder resolverlos lo antes posible.

Duración estimada: 1 semana.

Sprint 6: Mantenimiento y Mejoras Continuas

Objetivo: Realizar actualizaciones periódicas, corregir errores detectados post-lanzamiento y agregar nuevas funcionalidades o mejoras.

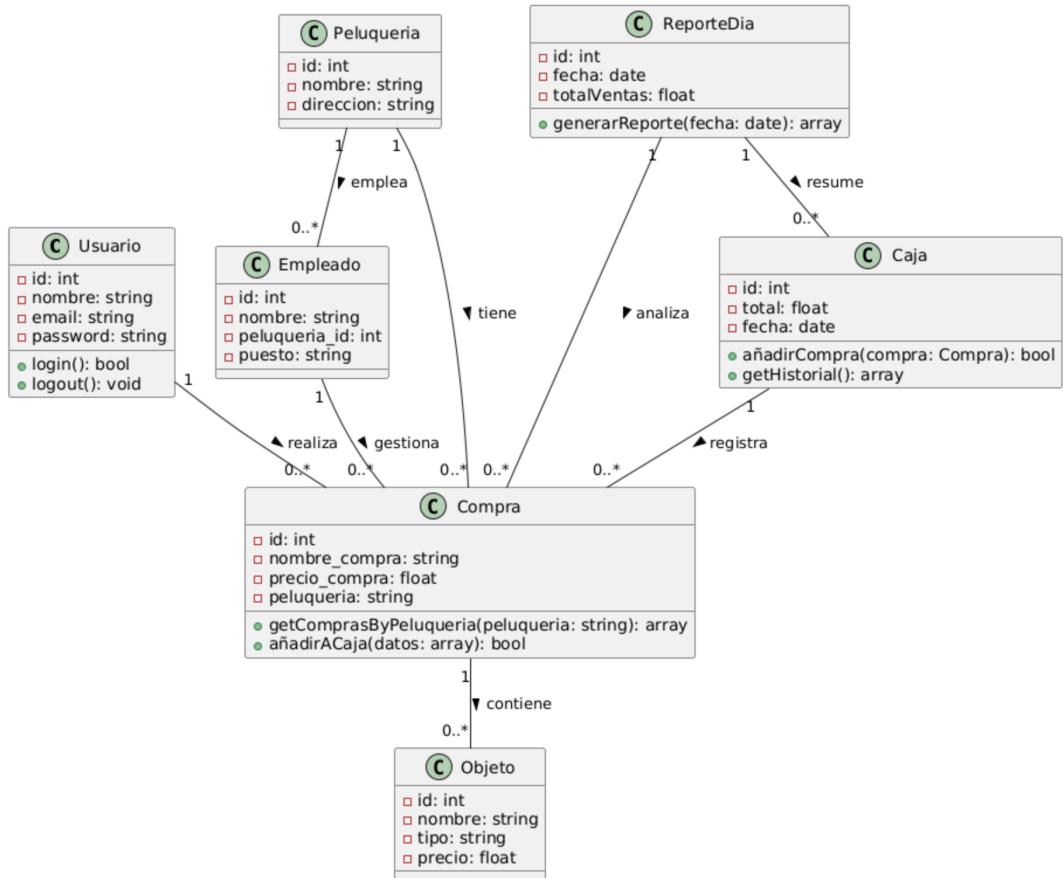
El mantenimiento y la mejora continua son parte integral del ciclo ágil. Se seguirá trabajando en el proyecto con ciclos de sprints posteriores.

Contenido:

- Soporte técnico para resolver cualquier incidencia post-lanzamiento.
- Actualización de contenido (por ejemplo, nuevos servicios o promociones).
- Recopilación de feedback de usuarios y clientes para posibles mejoras..

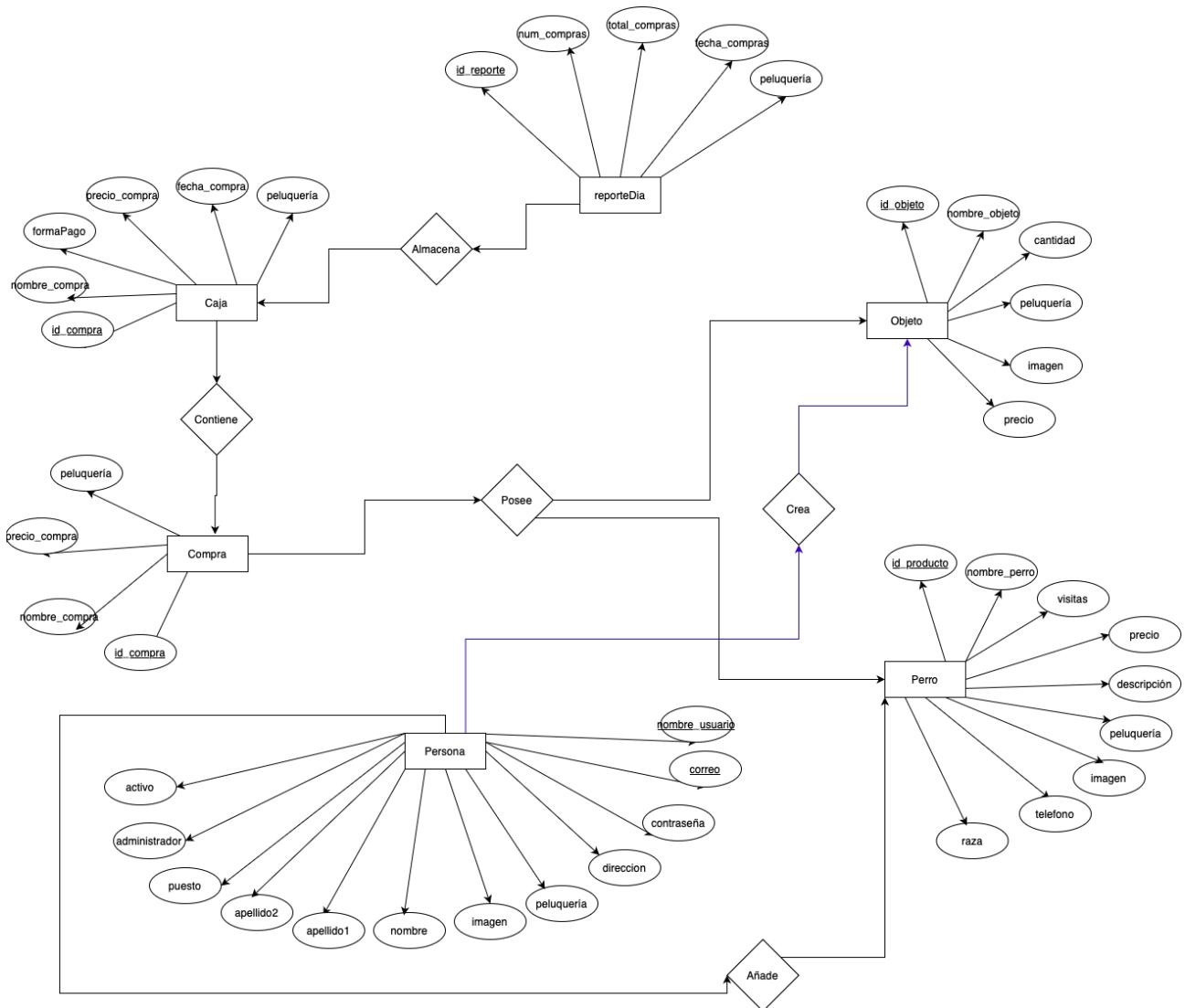
Duración estimada: 3-6 meses (dependiendo de la evolución del proyecto).

DISEÑO ESTÁTICO:



MODELO DE DATOS:

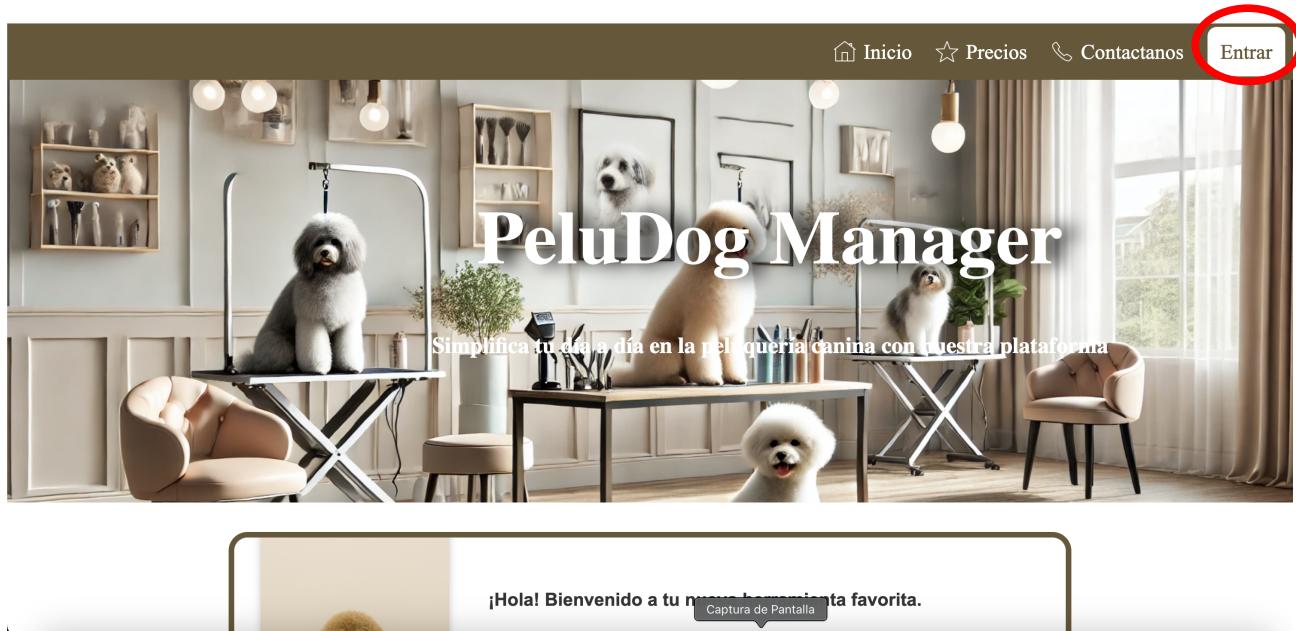
Modelo Entidad Relación:



MANUAL:

- **Importante:** La instalación la realizará nuestro técnico conectándose por remoto al sistema del cliente ya sea servidor o equipo.
- Una vez contratado el servicio de PeluDog Manager, el cliente(en este caso el jefe o administrador de la empresa) podrá acceder con la cuenta que le proporcionamos a través del botón ubicado en la parte superior derecha "ENTRAR".

*Recomendamos cambiar la contraseña una vez iniciada sesión por primera vez



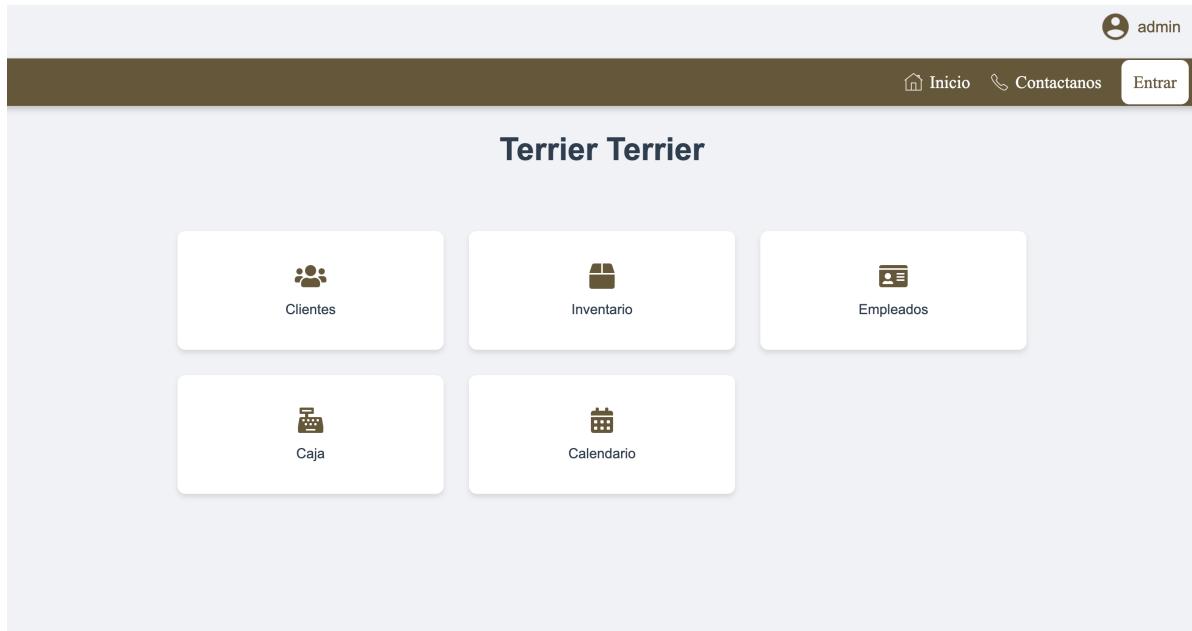
PeluDog Manager

Enviar

¿No tienes cuenta?

[Registrarse](#) [Volver al Inicio](#)

- Cuando iniciamos sesión accederemos a un menú con diferentes opciones para la gestión de la peluquería :



- En el apartado de "Clientes", nos aparecerán todos nuestros clientes, a su vez si clicamos sobre cualquiera de estos podremos ver información detallada de los mismos, pulsando el botón de editar podremos modificar la información de ese perro:

This screenshot displays a list of clients (dogs) with their details. Each entry consists of a small photo, the dog's name, its breed, the price for a service, and an 'editar' (edit) button.

Nombre	Raza	Precio	Opciones
Jacko	labrador	25.00€	
lake		35.00€	
Pepita29		20.00€	
pepito	labrador	10.00€	
dora	labrador	10.00€	
prueba29			
prueba30			
prueba30			
pepito			

At the bottom left is a 'Atrás' (Back) button, and at the bottom right is a '+' button for adding new clients. A watermark 'Captura de Pantalla' is visible at the bottom center.



Nombre:	Raza:
Jacko	labrador
Precio:	Telefono:
25.00€	640209584
Observaciones:	
Se porta bien, baño y cepillar bien	

(vista detalle al pulsar sobre cualquier perro)

- Si pulsamos en el botón "+" que aparece en la parte inferior derecha nos llevará a un formulario para poder añadir clientes:

Agregar Nuevo Perro

Nombre del perro
Precio
Descripción
Terrier Terrier
Visitas
Imagen (URL o nombre archivo)
Teléfono (9 dígitos)
Raza

   **Atrás**

- En la página “Ver inventario” podremos ver todos los objetos asociados a la peluquería, debajo de cada uno vemos un número, este número corresponde a la cantidad disponible que tenemos, el stock .

Junto con estos números tenemos un botón de + y otro - , los cuales nos sirven para poder añadir o en su defecto quitar cantidad.

The screenshot shows a user interface for managing inventory. At the top right, there is a user icon labeled "admin". Below the header, there are navigation links: "Inicio" (Home), "Contactanos" (Contact us), and "Entrar" (Enter). A search bar is located at the top center with the placeholder "Buscar un producto...".

Imagen	Nombre del objeto	Precio	Cantidad
	Flash Artero	11€	- 52 +
	Dentastix	8€	- 23 +
	prueba	10€	- 13 +
	prueba	10€	- 10 +
	prueba10	20€	- 100 +
	prueba29	40€	- 10 +

At the bottom left, there is a "Atrás" (Back) button with a left arrow icon. On the far right, there is a large circular button with a plus sign (+).

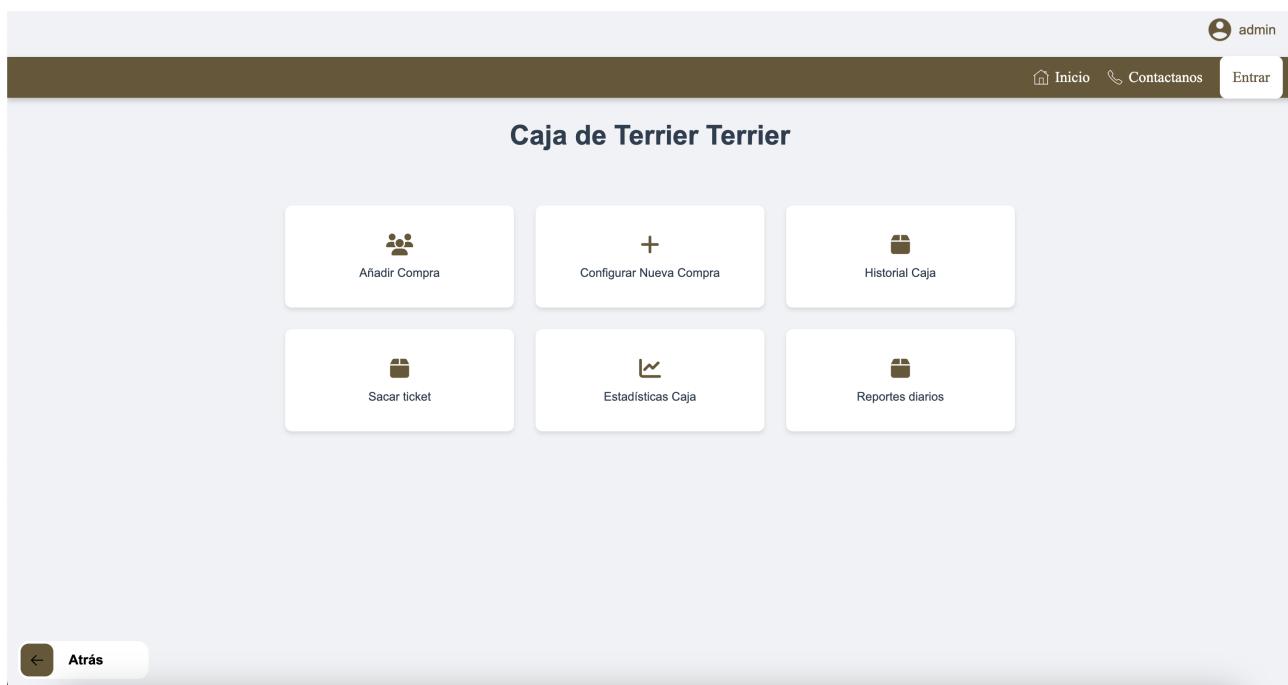
- Como pasa con los clientes si pulsamos en el botón “+” nos lleva a un formulario con el cual podemos añadir un objeto al inventario

The screenshot shows a modal dialog titled "Agregar Nuevo Objeto" (Add New Object). The form contains the following fields:

- Nombre del objeto (Name of the object)
- Cantidad (Quantity)
- Terrier Terrier (Category or brand)
- Precio (Price)
- Imagen (URL o nombre archivo) (Image (URL or file name))

At the bottom of the form is a large brown "Agregar Objeto" (Add Object) button.

- Al entrar en el apartado “Caja”, nos saldrán 6 opciones:



- En la sección “Añadir Compra” tendremos las opciones de compras predefinidas, cuando queramos añadir una, la elegimos, seleccionamos el método de pago y le damos a añadir.

Descripción	Forma de Pago	Precio	
Baño perro mediano	Efectivo	25€	Añadir a Caja
Baño y arreglo perro mediano	Efectivo	30€	Añadir a Caja

- En la sección de “Historial de Compra” nos aparecerán las compras que vayamos añadiendo, si pulsamos el botón que tenemos en la parte superior derecha

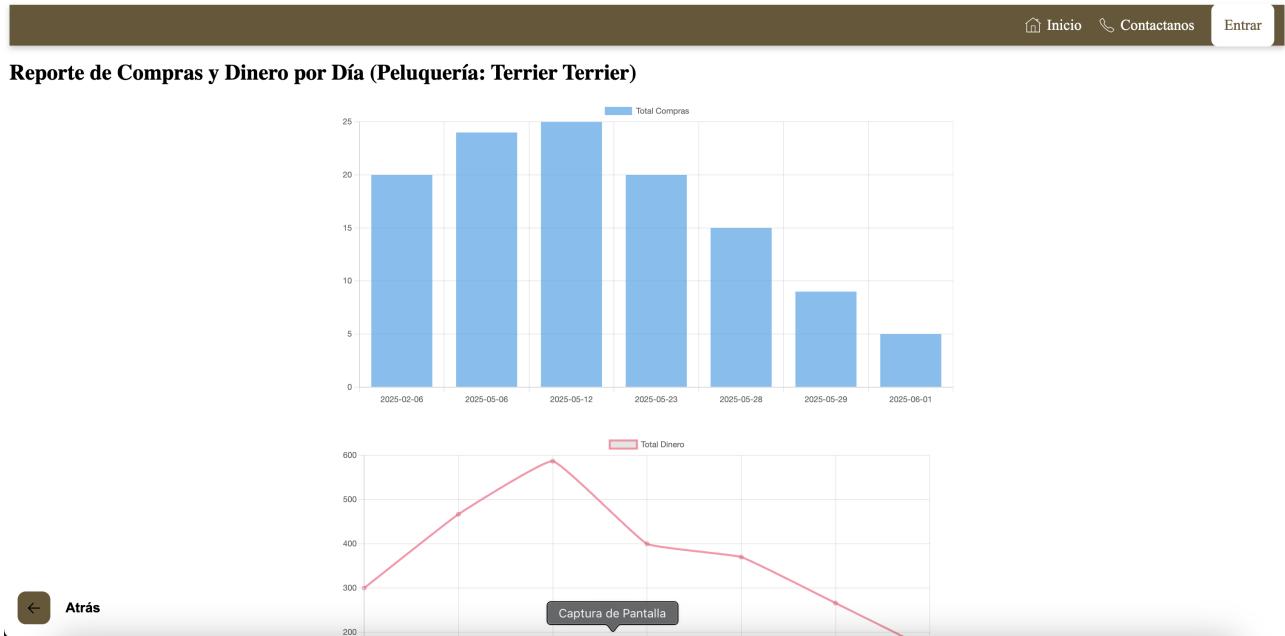
Descripción	Forma de Pago	Precio	Fecha Compra
Baño perro mediano	Efectivo	25€	2025-05-29
Baño perro mediano	Tarjeta	25€	2025-05-29
Baño y arreglo perro mediano	Tarjeta	30€	2025-05-29

Nos creara un “Reporte diario” con todas las compras realizadas en ese día, posteriormente se eliminarán de historial de compra

- En la sección de historial de reportes, tenemos todos estos reportes que generamos con el paso anterior

ID Reporte	Recaudacion reporte	Fecha Reporte
1	160€	2025-05-23
2	160€	2025-05-23
3	160€	2025-05-23
4	160€	2025-05-23

- En la sección “Estadísticas caja” podremos ver dos gráficos, uno nos muestra las ventas por día y otro nos hace un gráfico de lo ganado por día:



- En la sección de “Ver Empleados” podemos ver a todos los empleados que tenemos datos de alta, si eres administrador te aparecerá un botón en la parte inferior derecha “+”

admin

Inicio Contactanos Entrar

Los empleados registrados son:

- admin CEO
- admin ceo 2
- jorge ceo 2
- jorge2 ceo 22

Atrás Captura de Pantalla +

- Al pulsar este botón te llevará a este formulario a través del cual podrás dar de alta a otro empleado

Agregar Nuevo Empleado

Empleado añadido con éxito.

Nombre de usuario

Correo electrónico

Nombre

Primer apellido

Segundo apellido

Contraseña

Dirección

Puesto

URL de imagen (opcional)

□

¿Es administrador?

¿Está activo?

Terrier Terrier

Agregar Empleado

VIABILIDAD:

ESTUDIO DE MERCADO:

Gracias a mi amplia experiencia en el sector de las peluquerías caninas, se de primera mano que las aplicaciones de gestión para estas son nulas o las que están disponibles están pensadas para peluquerías de personas por lo que no se ajustan a las necesidades de estos negocios. Esto hace a muchas peluquerías caninas tener que recurrir a una web de gestión personalizada lo cual les resulta muy caro o muchas veces directamente inasumible.

En PeluDog Manager desarrollamos una web de gestión ya personalizada para estos negocios por lo que el precio respecto a la competencia se reduce considerablemente y a su vez aumentan las funcionalidades específicas para este sector que podemos ofrecer.

ANALISIS ECONÓMICO:

Recursos y plazos:

- **Duración:** 12 semanas de desarrollo inicial más 3-6 meses de mantenimiento
- **Miembros necesarios para el equipo:**
 - 1 Project Manager
 - 2 Desarrolladores full stack
 - 1 Diseñador (para tema de imágenes, logos o ayudar con el front)
 - 1 tester

- 1 Técnico de Sistemas

Presupuesto estimado:

- **Costes humanos:** 42.600€
- **Costes materiales:** 750€
- **Total aproximado:** 43.350€

Financiación:

- Fondos propios, préstamos, inversores privados.
- Posibilidad de ayudas públicas o subvenciones para digitalización

ESTUDIO DE RIESGOS:

1.Riesgos técnicos:

- Incompatibilidad entre tecnologías:
 - Descripción: Posibles conflictos entre versiones de PHP, SQL y BootStrap
 - Plan de prevención: Verificar que son compatibles desde el inicio y mantener las dependencias actualizadas
- Fallos de seguridad:
 - Descripción: Vulnerabilidades en la gestión de contraseñas o datos personales
 - Plan de Prevención: Aplicar buenas prácticas de seguridad como hashear las contraseñas
- Pérdida de datos:
 - Descripción: Fallos en la base de datos o errores en el código que borren información
 - Plan de Prevención: Implementar copias de seguridad automáticas

2.Riesgos del Proyecto:

- Retrasos en el desarrollo:
 - Descripción: Fallos en la planificación o cambios de requisitos
 - Plan de Prevención: Cumplir la metodología ágil con entregas por sprints y mantener la comunicación constante con el cliente
- Baja calidad del código:
 - Descripción: Código difícil de mantener o poco escalable
 - Plan de Prevención: Establecer estándares de codificación y revisión por pares

- Falta de documentación técnica:
 - Descripción: Puede dificultar el mantenimiento en un futuro
 - Plan de Prevención: Documentar código, base de datos y manual de usuario durante el desarrollo

3.Riesgos económicos y financieros:

- Superación del presupuesto:
 - Descripción: Costes no contemplados o subestimados
 - Plan de Prevención: Reservar un margen.
- Dificultad para conseguir financiación:
 - Descripción: Retrasos o rechazos de ayudas o subvenciones.
 - Plan de Prevención: Buscar varias fuentes de financiación

4.Riesgos de Usuarios y Clientes

- Resistencia al cambio:
 - Descripción: Empleados de peluquerías poco habituados a herramientas digitales
 - Plan de Prevención: Formación de cara a usar la web e interfaz intuitiva.
- Mal soporte:
 - Descripción: Incapacidad de atender incidencias tras el lanzamiento
 - Plan de Prevención: Establecer un canal de soporte claro y eficaz y también planificar los recursos para estos mantenimientos

CALIDAD:

1. Pruebas Funcionales

Objetivo: Verificar que el sistema cumple los requisitos establecidos.

Ejemplo:

- Crear, editar y cancelar una cita.
- Gestionar empleados y controlar el inventario.

2. Pruebas de Usabilidad

Objetivo: Evaluar que la interfaz sea intuitiva y fácil de usar para cualquier usuario.

Ejemplo:

- Un nuevo usuario puede registrar una venta sin necesidad de ayuda externa.
- Los formularios son claros y fáciles de completar.

3. Pruebas de Seguridad

Objetivo: Garantizar que los datos sensibles están protegidos y que no hay accesos indebidos.

Ejemplo:

- Verificar que las contraseñas estén hasheadas.
- Comprobar que usuarios no autorizados no puedan acceder al panel de administración.

4. Pruebas de Rendimiento

Objetivo: Comprobar que la aplicación responde correctamente bajo distintas cargas.

Ejemplo:

- Medir el tiempo de carga con 50 usuarios simulados.
- Añadir 100 elementos al inventario y verificar la estabilidad.

5. Pruebas de Compatibilidad

Objetivo: Confirmar que la aplicación funciona en diferentes dispositivos y navegadores.

Ejemplo:

- Probar en navegadores como Chrome, Firefox y Safari.
- Asegurar un diseño responsive en móviles, tablets y ordenadores.

6. Pruebas de Integración

Objetivo: Asegurar que los diferentes módulos del sistema interactúan correctamente.

Ejemplo:

- Registrar un nuevo cliente desde el frontend y confirmar que los datos se almacenan correctamente en la base de datos.
- Realizar una venta y verificar que se genera un reporte correctamente.

The screenshot shows the Katalon Recorder application window. On the left, there's a sidebar with 'Test Suites' (Passed: 1, Failed: 0), 'Dynamic Test Suites', 'Test Data', 'Extension script', and 'Profiles'. The main area is titled 'Untitled Test Case' and contains a table of recorded commands:

Command	Target	Value
open	http://localhost/proyectoFinal/Codigo/app/view/PHPva%C3%BDadirEmpleo.php	
click	name=nombre_usuario	manolo22
type	name=nombre_usuario	manolo22
click	name=correo	manolo33
type	name=correo	manolo33
click	name=correo	manolo33@manolo33
type	name=correo	manolo33@manolo33
click	name=nombre	manolo
type	name=nombre	manolo
click	name=apellido1	Gonzlez
type	name=apellido1	Gonzlez
click	name=apellido2	Gonzlez
type	name=apellido2	Gonzlez

At the bottom, there are tabs for 'Log', 'Screenshots', 'Variables', 'Reference', 'Self-healing', and a 'Dashboard' button. The 'Log' tab displays the recorded test steps:

```
[info] Playing test case CreateUsuarioTest / Untitled Test Case
[info] Time: Mon Jun 02 2025 08:45:37 GMT+0200 (hora de verano de Europa central) Timestamp: 174887537286
[info] OS: macOS Version: 10.15.7
[info] Browser: Chrome Version: 137.0
[info] If the test cannot start, please refresh the active browser tab
[info] Executing: | open | http://localhost/proyectoFinal/Codigo/app/view/PHPva%C3%BDadirEmpleo.php |
[info] Executing: | click | name=nombre_usuario |
[info] Executing: | type | name=nombre_usuario | manolo22 |
[info] Executing: | click | name=correo | manolo22 |
```

At the very bottom, it says 'Sign in to enable automatic backup. Refresh'.

CONCLUSIONES Y MEJORAS FUTURAS:

CONCLUSIÓN FINAL:

El desarrollo de PeluDog Manager ha resultado en una aplicación web funcional, adaptada a las necesidades específicas de las peluquerías caninas. El proyecto cumplió con los principales objetivos establecidos en la fase inicial: permitir la gestión de los perros, empleados, inventario y de la caja a través de una interfaz clara para los empleados de estos negocios.

Resultados Obtenidos:

- Se logró desarrollar e implementar una arquitectura basada en MVC, lo que facilita el mantenimiento y escalabilidad de la aplicación
- La web cuenta con funcionalidades clave como control de caja, seguimiento de inventario y clientes y administración de personal
- Se han cumplido los plazos de desarrollo establecidos en los sprints y se realizaron pruebas antes del despliegue.
- La accesibilidad es buena y se encuentran las funciones con facilidad

Mejoras previstas o pendientes:

- Calendario visual de reservas: Aunque se planificó, quedó como funcionalidad pendiente por priorizar otros aspectos más críticos.
- Panel de estadísticas e informes: Para poder ofrecer un gráfico sobre las ventas o poder agrupar por fechas
- Integración con los diferentes métodos de pago: En vez de registrarse manualmente las ventas con tarjeta se registrarán automáticamente.
- Aplicación móvil: Se tiene planeado desarrollar una versión para teléfono.

BIBLIOGRAFÍA

Para muchos de los link de la biografía utilizare MDN Web Docs la cual me sirvió de mucha ayuda para encontrar documentación sobre estas tecnologías.

- Lenguajes:
 - PHP: <https://www.php.net/>
 - JavaScript: <https://developer.mozilla.org/es/docs/Web/HTML>
 - HTML: <https://developer.mozilla.org/es/docs/Web/HTML>
 - CSS: <https://developer.mozilla.org/es/docs/Web/CSS>
 - Bootstrap (FrameWork de CSS): <https://getbootstrap.com/>
- Bases de datos y servidores:
 - MySQL (MariaDB en XAMPP): <https://mariadb.org/>
 - XAMPP: <https://www.apachefriends.org/es/index.html>
- Control de versiones:
 - GitHub: <https://github.com/>
- Editores de código:
 - Visual Studio Code: <https://code.visualstudio.com/>
- Inteligencia artificial:
 - Copilot: <https://copilot.microsoft.com/>
 - ChatGPT (OpenAI): <https://openai.com/chatgpt>
- Metodologías de Desarrollo:
 - Modelo MVC: <https://developer.mozilla.org/es/docs/Glossary/MVC>
 - Libreria Chart (Gráficos): <https://cdn.jsdelivr.net/npm/chart.js>

Para el diagrama de clases utilice <https://www.plantuml.com/>