



Elite Animal Spa



Parcial II:

Plataformas de Desarrollo

Noviembre 2025

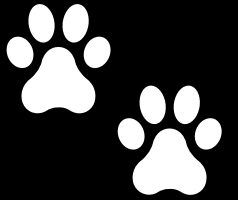
Spa Veterinario

Realizado por: *Carelis Fagundez Frías*
DNI:19.094.687

Carrera: Análisis de Sistemas
Comisión: ACN3BV
Profesor: Sergio Medina

Da Vinci
PRIMERA ESCUELA DE ARTE MULTIMEDIAL

PRESENTACIÓN



Élite Animal SPA es una aplicación web desarrollada como trabajo práctico para gestionar turnos de un centro de estética para perros y gatos.

El sistema permite que cada usuario pueda registrarse, administrar sus turnos y que el personal del SPA gestione los servicios de manera eficiente.

El proyecto sigue una arquitectura **Full Stack**:

Backend: Node.js + Express

Frontend: React

Persistencia: Archivos JSON

Roles: Cliente, Recepción, Admin

Diseño: Negro y dorado, estilo elegante.



OBJETIVOS:

Crear un backend modularizado con Express.
Implementar rutas REST para usuarios, mascotas y turnos.

Manejar datos mediante archivos .json simulando persistencia.

Crear un frontend con React Router dividido en secciones y rutas por rol.

Implementar login con validación por rol.
Gestionar turnos: alta, baja, edición.

Permitir que cada tipo de usuario vea solo lo que corresponde.

Entregar un proyecto completo:
backend + frontend + documentación.



OBJETIVOS:

Alcance Funcional

Cliente

Registrarse.

Iniciar sesión.

Cargar datos de su mascota.

Solicitar turnos.

Ver sus propios turnos.

Eliminar turnos propios.

Recepción

Ver todos los turnos.

Crear turnos para cualquier cliente.

Editar estado y observaciones del turno.

No puede eliminar usuarios.

Administrador

CRUD de usuarios (crear, editar y eliminar).

Ver todo el contenido del sistema.

Acceso total, excepto eliminar turnos en uso.

Mi Cuenta

[Perfil](#)[Mis Mascotas](#)[Pedir Turno](#)[Historial](#)

Pedir un Turno

Fecha *

11/26/2025

Horario *

09:00

Servicio *

Peluquería y spa

Mascota *

Lupe (Perro)

Confirmar Turno

Faltan datos obligatorios del turno o no seleccionaste una mascota.

[Cerrar sesión](#)

ALCANCE FUNCIONAL

Proceso de Desarrollo (reconstruido para entregar)

1. Planificación inicial

Definición del alcance.
Identificación de roles.
Diseño del flujo de turnos.

2. Configuración del Backend

Creación del proyecto con npm init.
Instalación de Express.
Estructuración de rutas y controladores.
Implementación de autenticación básica.
Pruebas de rutas con Thunder Client.

3. Persistencia

Creación de los archivos JSON.
Implementación de funciones de lectura/escritura.
Validación para evitar duplicados.

4. Frontend

Proyecto creado con Vite.
Estructura de páginas.
Implementación del login.
Protecciones por rol.
Formularios de turnos y mascotas.

5. Integración

Consumo del backend desde React con fetch.
Manejo de errores de API.
Estado global (AuthContext).

6. Diseño

Ajuste de estilos (negro + dorado).
Navbar responsivo.
Tarjetas, formularios y tablas estilizadas.

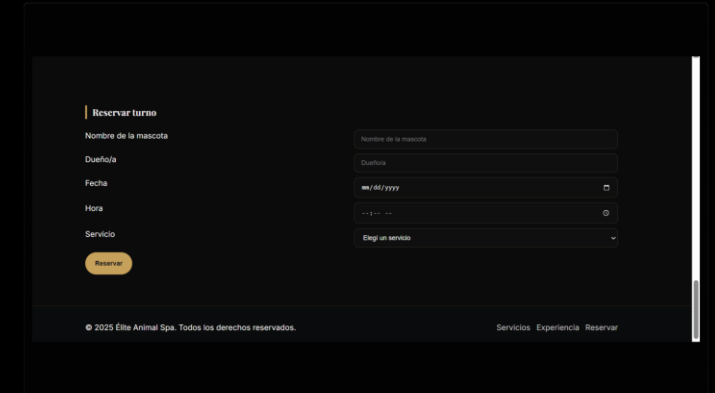
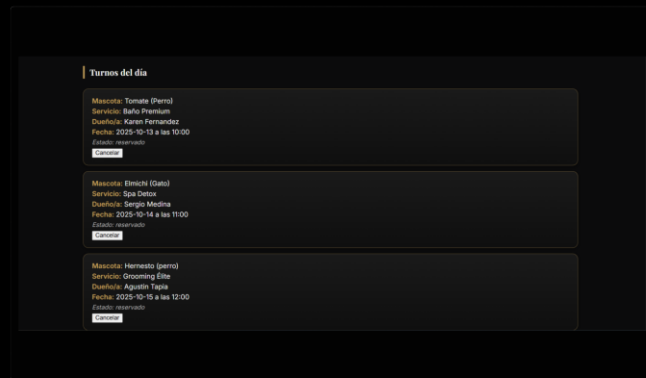
7. Testing manual

Pruebas de turnos por cada rol.
Validación de login y vista correspondiente.
Edición y eliminación de turnos.
Creación de usuarios desde admin.
Revisión de permisos.

8. Deploy local

Documentación de cómo correr el backend.
Documentación de cómo correr el frontend.

ROADMAP:



Flujo de uso de la pantalla

El usuario ingresa y ve los turnos existentes (persistidos en localStorage).

Puede filtrar por mascota, fecha o servicio, o buscar directamente por nombre.

Tiene la opción de limpiar los filtros.

Puede crear un nuevo turno completando el formulario (mascota, dueño/a, fecha, hora, servicio).

El sistema valida los campos y evita solapamientos de horario.

Los turnos creados aparecen en la lista y se guardan en localStorage.

Desde cada tarjeta, el usuario puede cancelar un turno (con confirmación).

ROADMAP:



Link Repositorio:

<https://github.com/carelis66/parcial-2-pd-acn4bv-fagundez.git>