

Jóvenes a Programar / Testing

REPORTE PROYECTO - ETAPA 3

TESTING
VETERINARIA GUAU
GUAU



TESTING G2

Sebastián Barboza

Fecha de Entrega: 16/11/2025

Introducción General

El proyecto “Testing – Veterinaria Guau Guau” forma parte del proceso de aprendizaje del curso Testing de Jóvenes a Programar y tiene como objetivo aplicar los conocimientos adquiridos en un entorno real de trabajo en equipo. A través de este proyecto, se busca desarrollar un ciclo completo de testing sobre un sistema web, poniendo en práctica las distintas etapas del proceso de aseguramiento de la calidad.

El trabajo se estructura en cuatro etapas: planificación, testing, retesting y evaluación.

- En la Etapa 1, el equipo planificó el proyecto, definió roles, responsabilidades y tareas.
- En la Etapa 2, se realizaron pruebas exhaustivas, tanto basadas en especificaciones como exploratorias, documentando los resultados obtenidos.
- En la Etapa 3, se efectuaron pruebas de confirmación y regresión, corrigiendo defectos detectados y verificando la estabilidad del sistema.
- Finalmente, en la Etapa 4, se evaluaron los resultados finales, se elaboró un informe ejecutivo y se preparó la presentación ante el equipo académico.

A lo largo de todas las fases, el grupo trabajó de manera colaborativa, aplicando técnicas de testing, herramientas de documentación y comunicación constante para asegurar la calidad del producto final. Este proyecto representa una experiencia práctica que consolida los conocimientos teóricos del testing y demuestra la importancia del trabajo en equipo dentro de un proceso de desarrollo de software.

Introducción Etapa 3

La Etapa 3 se enfoca en **verificar y validar las correcciones realizadas** tras los defectos reportados en la Etapa 2.

Su objetivo principal es asegurar que:

- Los defectos reportados hayan sido solucionados correctamente (Pruebas de Confirmación).
- Los cambios no generen errores nuevos en otras partes del sistema (Pruebas de Regresión).
- Se documenta el estado actual del software con respecto a su estabilidad y calidad.

Además, se integraron ideas de mejora relacionadas a experiencia de usuario, consistencia visual y navegación, que serán consideradas para futuras versiones del sistema.

Alcance

El alcance de esta etapa se centra en la verificación de las correcciones realizadas tras los defectos detectados en la Etapa 2, evaluando nuevamente las funcionalidades afectadas para confirmar su correcto funcionamiento.

Incluye la ejecución de pruebas de confirmación sobre los casos que anteriormente presentaron fallas, así como pruebas de regresión en las áreas críticas del sistema para asegurar que los cambios aplicados no generen nuevos errores.

No se incluyó en esta etapa la validación de seguridad, usabilidad ni rendimiento, las cuales se consideran fuera del alcance actual.

Desarrollo

Pruebas de Confirmación:

Se ejecutaron **pruebas de confirmación (retesting)** sobre los defectos que fueron corregidos. Estas pruebas permitieron verificar si las soluciones aplicadas resolvieron efectivamente los problemas detectados y asegurarse de que los cambios no generan nuevos comportamientos inesperados. En los casos donde el defecto persiste o no pudo validarse, se dejó documentado su estado actual.

Pruebas de Regresión:

Se realizaron pruebas de regresión sobre las áreas críticas del sistema para garantizar que las modificaciones implementadas no afectarán negativamente otras funcionalidades. Estas pruebas permitieron evaluar la estabilidad general del software luego de aplicar las correcciones.

Corrección de Defectos:

En esta etapa se abordaron los defectos identificados durante la Etapa 2, analizando su causa y aplicando las correcciones correspondientes. Es importante mencionar que **solo algunos de los defectos reportados pudieron ser corregidos**, ya sea por limitaciones del entorno, falta de acceso a ciertos componentes o porque algunos errores no pudieron reproducirse nuevamente. Cada corrección fue evaluada para confirmar su impacto y determinar si la solución era efectiva.

Conclusión

Las pruebas realizadas en esta etapa permitieron verificar las correcciones aplicadas y confirmar, mediante pruebas de confirmación, que solo algunos de los defectos identificados en la Etapa 2 fueron solucionados correctamente. Las pruebas de regresión mostraron que el sistema

mantiene su funcionamiento general sin introducir nuevos errores en áreas críticas.

En líneas generales, la aplicación continúa cumpliendo con los requerimientos funcionales principales, aunque aún requiere ajustes en validaciones, consistencia entre formularios y mensajes de error.

Hackatón de Mejoras

Durante el proyecto identificamos varias áreas donde la app podría mejorar, tanto en funcionalidad como en usabilidad. Un hackatón permitiría trabajar de manera intensiva en:

- **Optimización del flujo de registro y post-registro**, asegurando tiempos de carga más rápidos y transiciones más fluidas.
- **Revisión completa de validaciones de campos**, para que el usuario reciba mensajes claros e instantáneos ante errores.
- **Unificación del diseño** (estilos, botones, tamaños, colores) para que toda la app mantenga la misma línea visual.
- **Mejorar la navegación general**: Se podrían incluir botones más intuitivos, mejor posicionados y con estados visibles (hover, disabled, etc.)

Pruebas de Confirmación - Registro de Clientes

El objetivo de esta etapa fue verificar si los defectos identificados anteriormente continúan presentes o si fueron corregidos en la última versión de la aplicación. Para ello se re-ejecutaron casos de prueba específicos orientados a validar el comportamiento actual del formulario.

Se evaluaron las validaciones básicas de los campos, la respuesta del formulario ante datos incompletos o erróneos y el mensaje de confirmación tras un registro exitoso.

ID	Componente	Descripción	Resultado esperado	Resultado obtenido	Estado
TC01	registro_cliente	Completar todos los campos correctamente	Debe mostrar mensaje de registro exitoso	Muestra mensaje de confirmación	Correcto (Sin cambios)

TC02 Debe impedir el registro por uno, no muestra todos los campos sin aplicar requeridos a la vez
registro_client e mostrar mensaje "Por Impide el registro y muestra todos los campos Sigue dejar todos los favor, campos obligatorios complete todos los vacíos" e muestra el mensaje de error, pero uno vez
correcto (Sin cambios - sigue sin aplicar mejoras)

TC03	registro_client.e.html	Ingresar solo números en el campo "Nombre"	Debería impedir el registro y mostrar mensaje de error	Permite registrar "1234" como nombre válido	Defecto funcional aún ocurre
------	------------------------	--	--	---	------------------------------

TC04	registro_cliente.html	Formato de Email Inválido	Debería impedir el registro y mostrar mensaje de error	Aparece mensaje de error indicando formato incorrecto	Correcto (Sin cambios)
TC05	registro_cliente.html	Carga de la siguiente página luego de crear la cuenta	Página debería abrir sin errores	Error: archivo no encontrado	Correcto (Solucionado)

Las pruebas de confirmación permitieron verificar el estado actual de la funcionalidad **Registro de Clientes**.

Durante las pruebas se confirmó que varias validaciones funcionan como deberían, como el registro exitoso y el manejo del email inválido.

Además, se verificó que **el problema de carga de la página post-registro fue solucionado**, ya que la siguiente pantalla ahora se muestra sin errores técnicos ni mensajes de “archivo no encontrado”.

Sin embargo, aún persisten defectos que requieren ajustes, especialmente:

- La visualización de mensajes de error cuando los campos obligatorios están vacíos, ya que se muestran uno por uno en lugar de todos juntos.
- La validación del campo “Nombre”, que todavía permite el ingreso de valores numéricos.

Estos hallazgos indican que aún se requieren ajustes para asegurar una experiencia de usuario adecuada y un funcionamiento completamente correcto del formulario.

Registro de Clientes - Problemas Identificados

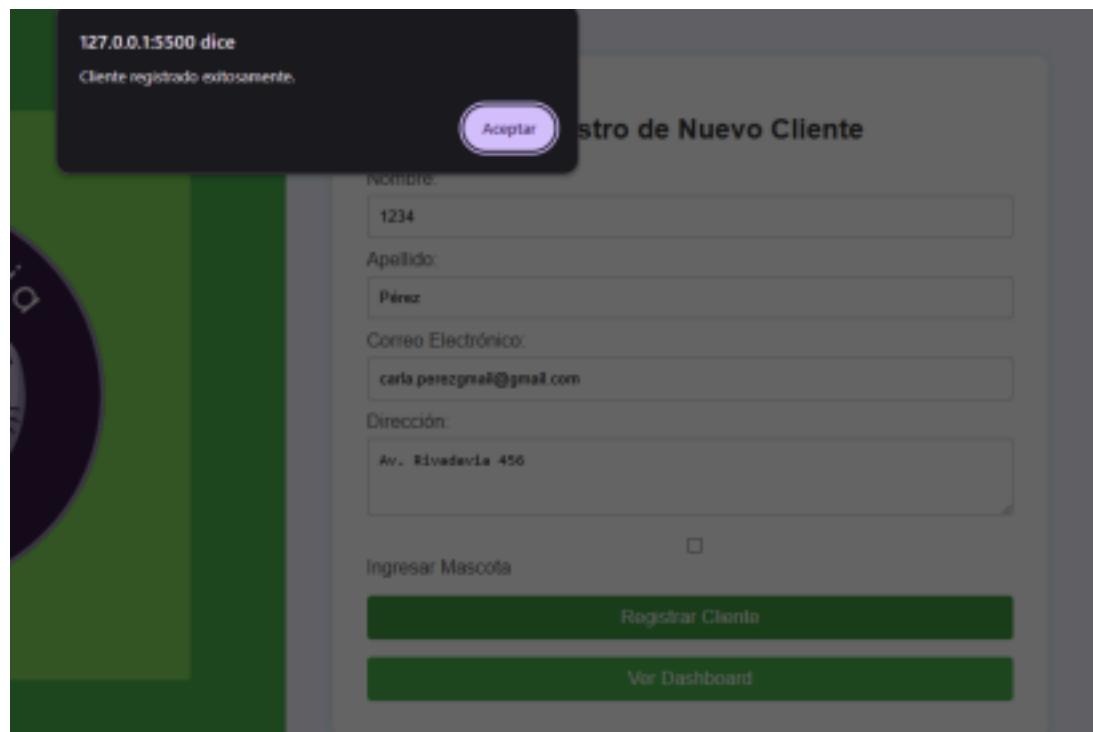
- **Validación de campos vacíos:** El sistema muestra errores de forma secuencial en lugar de simultánea

The screenshot shows a registration form titled "Registro de Nuevo Cliente". The form fields are as follows:

- Nombre: An input field containing a single character.
- Apellido: An input field with a validation message: "Completa este campo" (Complete this field) with an exclamation mark icon.
- Correo Electrónico: An empty input field.
- Dirección: An empty input field.
- Ingresar Mascota: A checkbox labeled "Ingresar Mascota" with an unchecked state.
- Buttons: Two green buttons at the bottom: "Registrar Cliente" and "Ver Dashboard".

- **Falta de indicadores visuales:** No hay asteriscos ni etiquetas que identifiquen campos obligatorios

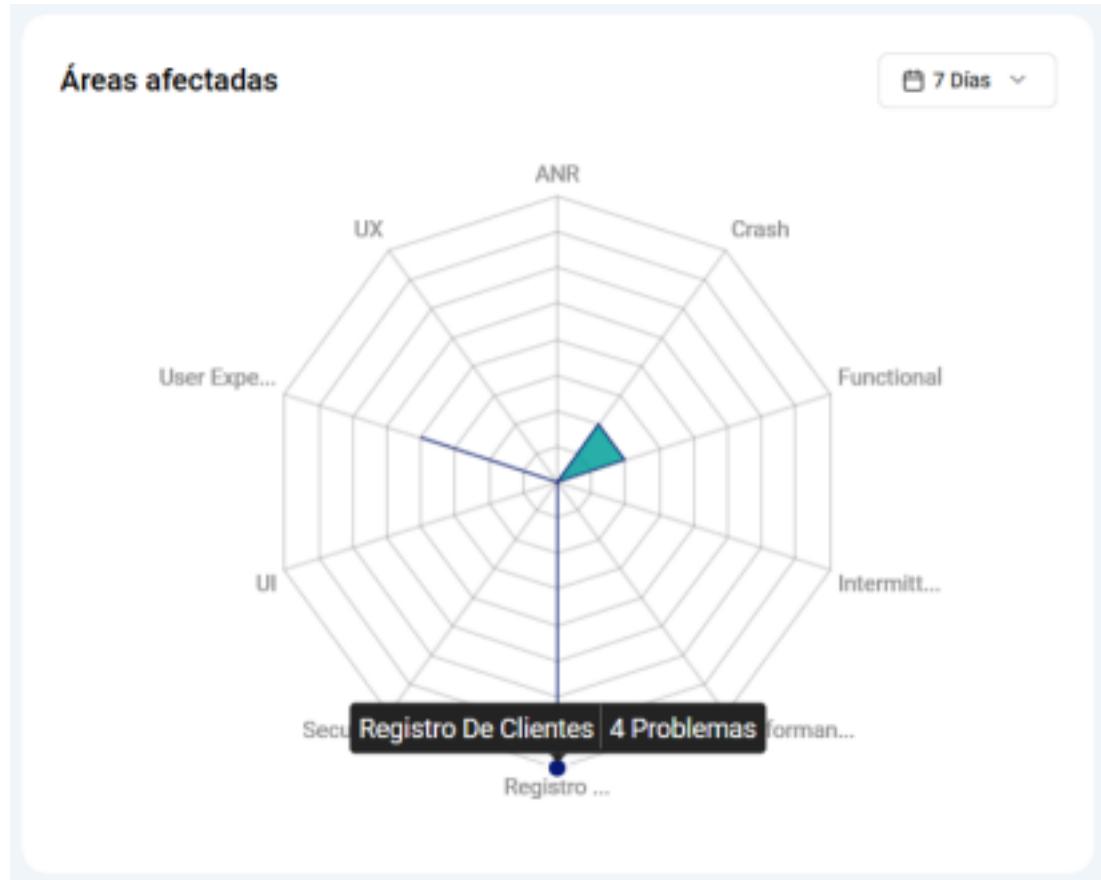
- **Validación de formato de nombre:** Permite ingreso de números en campo destinado a texto



Registro de Clientes - Problemas Identificados (Gráficas)

A screenshot of a bug tracking system interface. At the top, there are four summary boxes: "TOTAL ISSUES" (8), "NEW" (4), "IN PROGRESS" (4), and "CLOSED" (0). Below this is a search bar and filter options. The main area lists four identified issues:

- VET05** MEDIUM: Registro Clientes - Validación de campos vacíos: El sistema muestra errores de forma secuencial en lugar de simultánea. Status: Not Fixed. Created by: User Experience / Registro De Clientes. Last updated: 1st Nov 2025 by María Pía Silvera.
- VET07** CRITICAL: Registro Clientes - Validación de formato de nombre: Permite ingreso de números en campo destinado a texto. Status: Not Fixed. Created by: User Experience / Registro De Clientes. Last updated: 1st Nov 2025 by María Pía Silvera.
- VET08** CRITICAL: Registro Clientes - Error técnico post-registro: Problema de carga de archivos después del registro exitoso. Status: Fixed. Created by: User Experience / Registro De Clientes. Last updated: 1st Nov 2025 by María Pía Silvera.
- VET06** LOW: Registro Clientes - Falta de indicadores visuales: No hay asteriscos ni etiquetas que identifiquen campos obligatorios. Status: Not Fixed. Created by: User Experience / Registro De Clientes. Last updated: 1st Nov 2025 by María Pía Silvera.



Pruebas de Confirmación - Registro de Mascotas

Durante la verificación se repitieron los casos de prueba de la Etapa 2 para comprobar si los errores fueron corregidos.

Se observó que los formularios aún presentan problemas de validación:

El formulario de registro de mascotas permite ingresar números y caracteres especiales en el campo nombre y fechas futuras e inexistentes en fecha de nacimiento.

Las validaciones no fueron corregidas correctamente. Los formularios continúan aceptando datos inválidos y muestran mensajes de éxito al registrar información incorrecta.

ID	Componente	Datos de entrada	Resultado esperado	Resultado obtenido	Estado

CP M1	registro_mascota	Nombre: "Azul" Fecha: 2019-06-08 Especie: "Canicoker"	Mensaje de confirmación: "Registrado correctamente"	Mensaje de confirmación: "Registrado correctamente"	Correcto (Sin Cambios)
CP M2	registro_mascota	Nombre: "" Fecha: 2019-06-08 Especie: "Canicoker"	Mostrar mensaje: "Rellenar este campo"	Muestra mensaje: "Rellenar este campo"	Correcto (Sin Cambios)

CP M3	registro_mascota	Nombre: "12345" Fecha: 2024-05-10 Especie: "Gato"	Debería mostrar mensaje de error por tipo de dato inválido	No muestra error, permite el registro	Observación
CP M4	registro_mascota	Nombre: "9187" Fecha: "" Especie: "Canicoker"	Mostrar mensaje: "Rellenar este campo"	Muestra mensaje: "Rellenar este campo"	Correcto (Sin Cambios)

CP M5	registro_mascota	Nombre: "Nina" Fecha: 2024-05 -10 Especie: "123"	Debería mostrar mensaje de error por tipo de dato inválido	Registro exitoso sin validación	Observación
----------	------------------	--	--	---------------------------------------	-------------

ID	Componentes	Datos de Entrada	Resultado Esperado	Resultado Obtenido	Estado
CP-M6	registro_mascota.html	Nombre:Luna Fecha:01/01 /2030 Especie:Perro	El sistema debe mostrar un mensaje de error indicando que la fecha no puede ser futura y no registrar la mascota.	No muestra error permite el registro	Incidente

CP-M 8	registro_mascota. ht ml	Nombre:Bo by \$\$ Fecha:20/06 /2 018 Especie:Perro	El sistema debe mostrar un mensaje de error indicando caracteres no permitidos en el nombre y no registrar la mascota.	No muestra error permite el registro	Incidente
CP-M 9	registro_mascota. ht ml	Nombre:Mo rg an Fecha: 02/01/2757 60 Especie:Ga to	El sistema debe mostrar un mensaje de error indicando que la fecha es inexistente	No muestra error permite el registro	Incidente

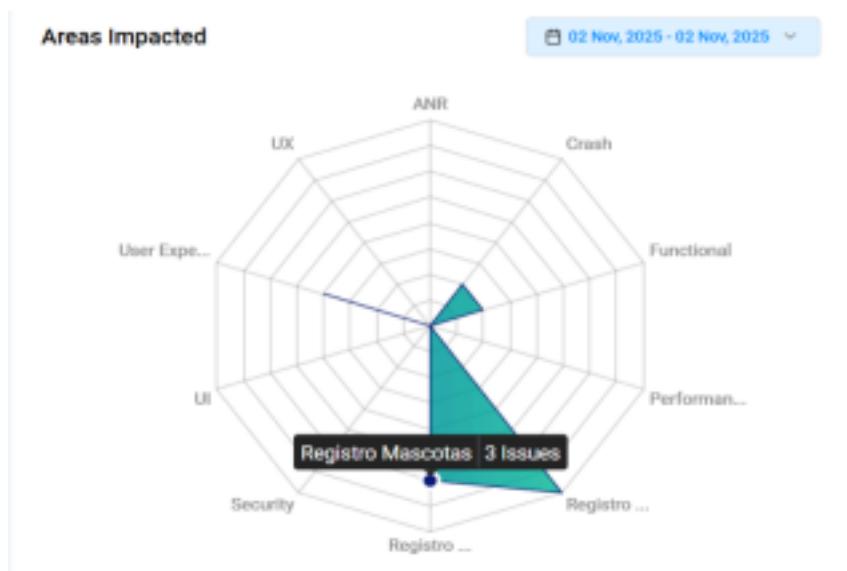
El formulario cumple parcialmente con las validaciones requeridas. Los campos obligatorios muestran mensajes de advertencia cuando están vacíos, pero aún no existen restricciones que impidan el ingreso de datos no válidos (como números y caracteres especiales en campos de texto). Se recomienda agregar validaciones de tipo de datos para mejorar la calidad y consistencia de la información registrada.

Si vemos corregido el ingreso de especies y datos no válidos en el campo especie, al cual se le fue agregado un menú para la selección de la especie correspondiente con categorías pre definidas, mitigando la posibilidad de datos que no correspondan.

En cuanto al campo fecha también se le fue agregado un calendario desplegable para la selección de la fecha correspondiente, pero el mismo no impide el registro de fechas no existentes o futuras, además de que detecta la incidencia de que al marcar la fecha luego por error queda el día en el lugar correspondiente al mes. Como sugerencia de mejora se recomienda que no se pueda registrar una fecha mayor al día actual y que se restrinja el número permitido en el campo correspondiente al año.

Registro de Mascotas - Problemas Identificados

(Gráficas)



Pruebas de Confirmación - Registro de Vacunas

El objetivo de este caso de prueba es validar el funcionamiento del formulario `registro_vacunas.html`, comprobando que las vacunas puedan asociarse correctamente a una mascota registrada, que las fechas sean válidas y que el sistema responda adecuadamente ante valores incorrectos o campos vacíos.

Se realizaron diferentes combinaciones de entrada para evaluar la respuesta del formulario ante datos válidos, omisiones de campos y entradas incorrectas (como letras en campos numéricos).

ID	Componente	Datos de entrada	Resultado esperado	Resultado obtenido	Estado
----	------------	------------------	--------------------	--------------------	--------

CP-V1
 registro_vacunas.html Dosis: "2"
 Nombre: "Rabia" Fecha aplicación: "24-10-2025" Mensaje de confirmación: "24-10-2025"
 Fecha vencimiento: "24-11-2025" Mensaje de confirmación: "24-11-2025"
 "Registra do correcta mente" Mensaje de confirmación: "Correcto (Sin Cambios)"

CP-V2 registro_vacu Nombre: Mostrar Registro Observación

	nas.html	"Vegetta" Dosis: "siete" Fechas Fecha de aplicación: "24-10-2025" Fecha de vencimiento: "24-11-2025"	error por tipo de dato no numérico	exitoso, sin validación del tipo de dato	
--	----------	---	------------------------------------	--	--

CP-V3	registro_valores_cuanas.html	Nombre: "" Dosis: "5" Fechas válidas	Mostrar mensaje: "Rellenar este campo"	Muestra mensaje: "Rellenar este campo"	Correcto (Sin Cambios)
CP-V4	registro_valores_cuanas.html	Nombre: "Willyrex" Dosis: "5" Fechas vacías	Mostrar mensaje: "Rellenar este campo"	Muestra mensaje: "Rellenar este campo"	Correcto (Sin cambios)

Pruebas de Confirmación – Login

Tras la ejecución de los casos de prueba correspondientes al módulo de Login de la aplicación “Veterinaria Guau Guau”, se verificó que todas las funcionalidades evaluadas operan correctamente según los criterios establecidos.

Las pruebas cubrieron comportamientos clave como validación de credenciales válidas, manejo de usuarios inexistentes, contraseñas incorrectas, campos obligatorios vacíos, distinción entre mayúsculas y minúsculas, y detección de espacios dentro de los campos de autenticación. En todos los casos, el resultado obtenido coincidió con el resultado esperado, lo que confirma que el sistema gestiona de forma adecuada tanto los ingresos exitosos como los intentos fallidos.

Asimismo, se comprobó que los mensajes de error se muestran de manera clara y coherente, y que las validaciones cumplen con los requisitos funcionales definidos.

En conclusión, el proceso de login demuestra un comportamiento estable,

consistente y alineado con las especificaciones, por lo que puede considerarse funcionalmente validado dentro del alcance de estas pruebas.

ID	Precondiciones	Datos de Entrada	Resultado Esperado	Resultado Obtenido	Estado
LN01	El usuario debe estar registrado con credenciales válidas.	Usuario: estudiante Contraseña: 12345	El sistema valida las credenciales y redirige al usuario a la pantalla de “Registro de Nuevo Cliente”.	Permite el acceso y redirige a la pantalla de registro de nuevo cliente.	Correcto
LN02	Debe existir al menos una cuenta registrada en el sistema.	Usuario: usuarioInexistente Contraseña: 12345	Resultado esperado: El sistema muestra un mensaje de error genérico “Usuario o contraseña incorrectos”.	Muestra advertencia de “Usuario o contraseña incorrectos”	Correcto

LN03	Debe existir al menos una cuenta registrada en el sistema.	Usuario: estudiante Contraseña: claveInvalida123	El sistema muestra un mensaje de error genérico “Usuario o contraseña incorrectos”.	Muestra advertencia de “Usuario o contraseña incorrectos”	Correcto
LN04	El formulario de inicio de	Usuario: (vacío)	El sistema resalta los campos	El sistema resalta el campo de	Correcto

	sesión debe estar visible.	Contraseña: (vacío)	obligatorios y muestra un mensaje de error indicando “Por favor rellene el campo”	usuario y muestra un mensaje de error indicando “Por favor rellene el campo”	
--	----------------------------	---------------------	---	--	--

LN05	El formulario de inicio de sesión debe estar visible.	Usuario: (vacío) Contraseña: 12345	El sistema resalta el campo “Usuario” y muestra un mensaje de error indicando “Por favor rellene el campo”	El sistema resalta el campo “Usuario” y muestra un mensaje de error indicando “Por favor rellene el campo”	Correcto
LN06	El formulario de inicio de sesión debe estar visible.	Usuario: estudiante Contraseña: (vacío)	El sistema resalta el campo “Contraseña” y muestra un mensaje de error indicando “Por favor rellene el campo”	El sistema resalta el campo “Contraseña” y muestra un mensaje de error indicando “Por favor rellene el campo”	Correcto

LN07	Debe existir una cuenta registrada.	Usuario: EStuDianTE Contraseña: 12345	El sistema debe rechazar las credenciales distinguiendo mayúsculas y minúsculas.	Muestra advertencia de “Usuario o contraseña incorrectos”	Correcto
LN08	Debe existir una cuenta registrada con credenciales válidas sin espacios.	Usuario: es tudente Contraseña: Gua 12 3 45	El sistema debe rechazar las credenciales.	Muestra advertencia de “Usuario o contraseña incorrectos”	Correcto

Testing de Regresión

En esta etapa se realizaron pruebas de regresión con el objetivo de verificar que las funcionalidades ya implementadas —formularios de registro de clientes, mascotas y vacunas— continúen funcionando correctamente tras la incorporación de nuevos cambios o correcciones en el sistema.

Las pruebas se ejecutaron de forma manual, utilizando los casos de prueba anteriores como referencia, y se llevaron a cabo después de cada actualización de código o despliegue.

Se validaron distintos escenarios en cada módulo:

- Clientes: registro exitoso con datos válidos y validación de campos vacíos.
- Mascotas: alta correcta de una mascota y control de errores al dejar campos obligatorios vacíos.
- Vacunas: registro con datos válidos y bloqueo ante el ingreso de datos inválidos (como fechas futuras).

Como criterios de aprobación, se estableció que todos los casos mantuvieran sus resultados esperados previos, que las relaciones entre entidades (mascotas y vacunas) siguieran estables y que no surgieran errores nuevos en las funcionalidades probadas.

Testing Exploratorio

Resumen Testing Exploratorio (Sebastian Barboza)

En la Etapa 3 – Retesting re-ejecuté pruebas clave sobre los formularios y el DDL corregido, actualizando el estado de las incidencias y realizando testing exploratorio con foco en: validación de datos, flujo de navegación, mensajes al usuario y comportamiento de los botones.

Los resultados muestran que las validaciones básicas de campos vacíos funcionan, pero faltan controles más avanzados (formato de correo, longitud mínima de nombres, coherencia entre fechas). El flujo principal entre pantallas es utilizable, aunque existen inconsistencias como la redirección a login.html con mensaje de error, la ausencia de listado de clientes en el Dashboard y el checkbox “Ingresar Mascota” sin comportamiento definido. Los mensajes al usuario se basan casi exclusivamente en alertas del navegador, lo que genera una experiencia poco moderna.

Entre las principales oportunidades de mejora se destacan: reforzar

validaciones (especialmente fechas y correos), definir mejor las redirecciones posteriores al guardado, reemplazar las alertas por mensajes visuales dentro del formulario, resaltar campos con error y mostrar más contexto (por ejemplo, la mascota asociada en el registro de vacunas).

Respecto a la Base de Datos, en Etapa 3 se verificó nuevamente el script original con la base VeterinariaGuauGuau, confirmando que en SQL Server el DDL se ejecuta correctamente y crea las tablas Clientes, Mascotas y Vacunas sin errores, mientras que para otros motores se mantienen los ajustes propuestos en la etapa anterior.

Misión General

Explorar los tres formularios de registro del sistema de veterinaria (cliente, mascota y vacunas) con el objetivo de evaluar la validación de campos, la navegación entre pantallas, los mensajes de error y el comportamiento general de los botones.

Misión 1: Registro de Cliente

Archivo: registro_cliente.html

Objetivo: Verificar el correcto funcionamiento del formulario de registro de cliente, las validaciones de los campos y la navegación hacia el registro de mascota.

Pruebas realizadas:

1. Enviar formulario vacío.
2. Completar parcialmente los campos.
3. Ingresar formato de correo inválido.
4. Completar correctamente todos los campos.
5. Usar botón “Registrar Mascota”.
6. Activar checkbox “Ingresar Mascota” sin usar el botón.

7. Usar botón “Ver Dashboard”.

Resultados observados:

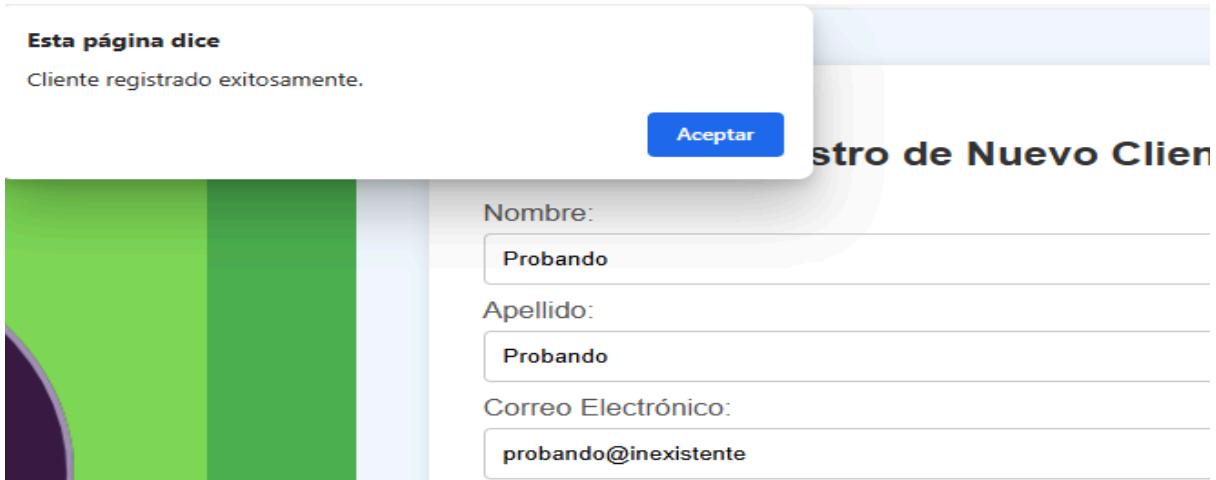
- Al dejar campos vacíos se muestra el mensaje “Por favor, complete todos los campos.”

Cuando el usuario intenta registrar un cliente sin completar alguno de los campos obligatorios, el sistema ejecuta una validación básica que impide el envío del formulario. La interfaz muestra una alerta emergente con el mensaje “Por favor, complete todos los campos.”, notificando al usuario que debe completar todos los campos antes de continuar.

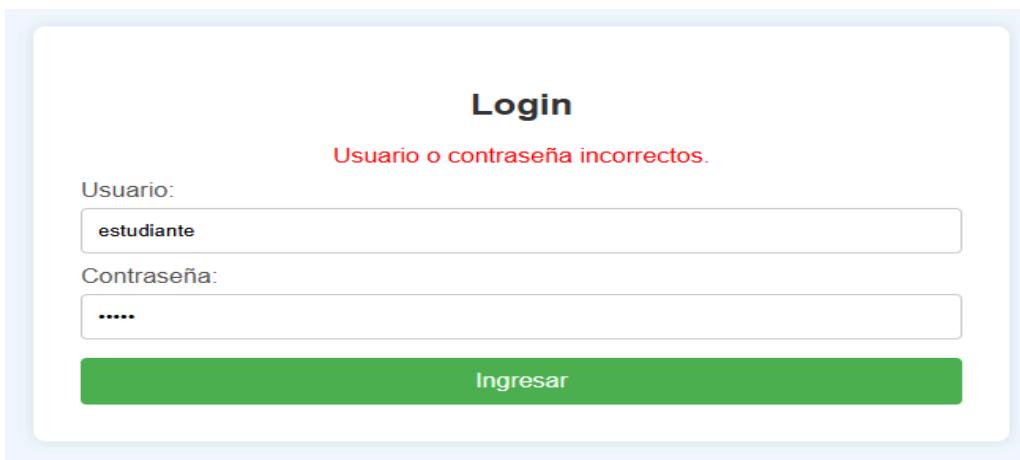
La validación funciona correctamente, aunque se limita únicamente a la presencia o ausencia de valores, sin aplicar reglas más avanzadas sobre el contenido ingresado.

- El campo email no valida correctamente el formato, permitiendo correos inválidos, durante el testing exploratorio se completó el formulario con un correo electrónico que no contiene un dominio válido.

por ejemplo: probando@inexistente



- Al completar correctamente todos los campos se muestra una alerta de éxito (“Cliente registrado exitosamente”). Al presionar Aceptar, el sistema redirige a [login.html](#), pero la pantalla de Login se carga con las credenciales del usuario administrador previamente usadas y ya muestra el mensaje “Usuario o contraseña incorrectos.”, sin que el usuario haya intentado autenticarse en ese momento.



- El botón “Registrar Mascota” redirige correctamente a “registro_mascota.html”.

El checkbox “Ingresar Mascota” no tiene función asociada.

Al activar el botón “Registrar Mascota”, el sistema realiza correctamente la navegación esperada: redirige al formulario [registro_mascota.html](#) sin errores visibles y sin interrupciones en el flujo.

La funcionalidad se comporta de acuerdo a lo previsto en el diseño del sistema, permitiendo continuar con el proceso de registro de mascotas inmediatamente después de haber completado o revisado los datos del cliente.

Sin embargo, la acción se ejecuta sin validar previamente el formulario del cliente y sin considerar el estado del checkbox “Ingresar Mascota”, lo cual puede provocar saltos de flujo inesperados si el usuario no completó los datos correctamente antes de pasar al formulario siguiente.

- El botón “Ver Dashboard” redirige correctamente a la pantalla de Dashboard, pero allí solo se muestran el Listado de Mascotas y el Listado de Vacunas. No se visualiza ningún listado de clientes, por lo que no es posible comprobar desde la interfaz si el cliente recién registrado fue efectivamente almacenado.

Dashboard

[Registrar Cliente](#)[Cerrar Sesión](#)

Listado de Mascotas

ID Mascota	Nombre	Fecha de Nacimiento	Especie	Cliente ID
1	Perro	2025-11-16	Gato	2

Listado de Vacunas

ID Vacuna	Nombre	Dosis	Fecha de Aplicación	Fecha de Vencimiento	ID Mascota
1	s	Segunda	2025-11-13	2025-12-06	1

Oportunidades de mejora:

- Agregar validación de formato de correo electrónico.
- Resaltar visualmente los campos con error

Además del mensaje general, cada campo incompleto o inválido debería marcarse en rojo para mejorar la usabilidad.

- Validar longitud mínima para Nombre y Apellido

Actualmente el formulario permite cargar nombres de un solo carácter. Se recomienda exigir mínimo 2–3 caracteres alfabéticos para garantizar integridad de datos.

- Incorporar un listado de clientes en el Dashboard.
La pantalla de Dashboard debería incluir también una tabla de clientes (por ejemplo: ID Cliente, Nombre, Apellido, Correo, Dirección), de forma consistente con las tablas de Mascotas y Vacunas. Esto permitiría:
 - 1 - Verificar rápidamente que el cliente registrado aparece en el sistema.
 - 2 - Mantener la coherencia del concepto de “Dashboard” como resumen general de la información.
 - 3 - Facilitar la trazabilidad entre clientes, mascotas y vacunas.
- Implementar funcionalidad para el checkbox “Ingresar Mascota”.
- Alinear el checkbox de Ingresar Mascota con el texto.
- Sustituir las alertas por mensajes visuales en pantalla.

Misión 2: Registro de Mascota

Archivo: registro_mascota.html

Objetivo: Verificar que se puedan registrar mascotas correctamente y validar el funcionamiento de los botones “Cancelar” y “Registrar Vacunas”.

Pruebas realizadas:

1. Enviar formulario vacío.

2. Completar parcialmente los campos.
3. Completar todos los campos correctamente.
4. Usar botón “Cancelar”.
5. Usar botón “Registrar Vacunas”.
6. Ver Dashboard

Resultados observados:

- Al dejar campos vacíos se muestra un mensaje indicando que complete el campo o los campos faltantes.

Registro de Mascota

Nombre:

Fecha de Nacimiento:
 CALENDAR

Especie:
 ▼

! Selecciona un elemento de la lista.

[Registrar Mascota](#)

[Registrar Vacunas](#)

[Ver Dashboard](#)

- Al completar todos los campos se muestra una alerta de éxito y redirige a “registro_cliente.html”.

Esta página dice
Mascota registrada exitosamente.

Aceptar

Registro de Mascota

Nombre:

Pedrito

Fecha de Nacimiento:

03 / 10 / 2025



Especie:

Gato

[Cancelar](#)

[Registrar Mascota](#)

[Registrar Vacunas](#)

[Ver Dashboard](#)

El botón “Cancelar” limpia los campos y redirige correctamente.

Esta página dice

¿Estás seguro de que deseas cancelar el registro de la mascota?

Aceptar

Cancelar

Registro de Mascota

Nombre:

Gatito

Fecha de Nacimiento:

13 / 11 / 2025



Especie:

Gato

[Cancelar](#)

[Registrar Mascota](#)

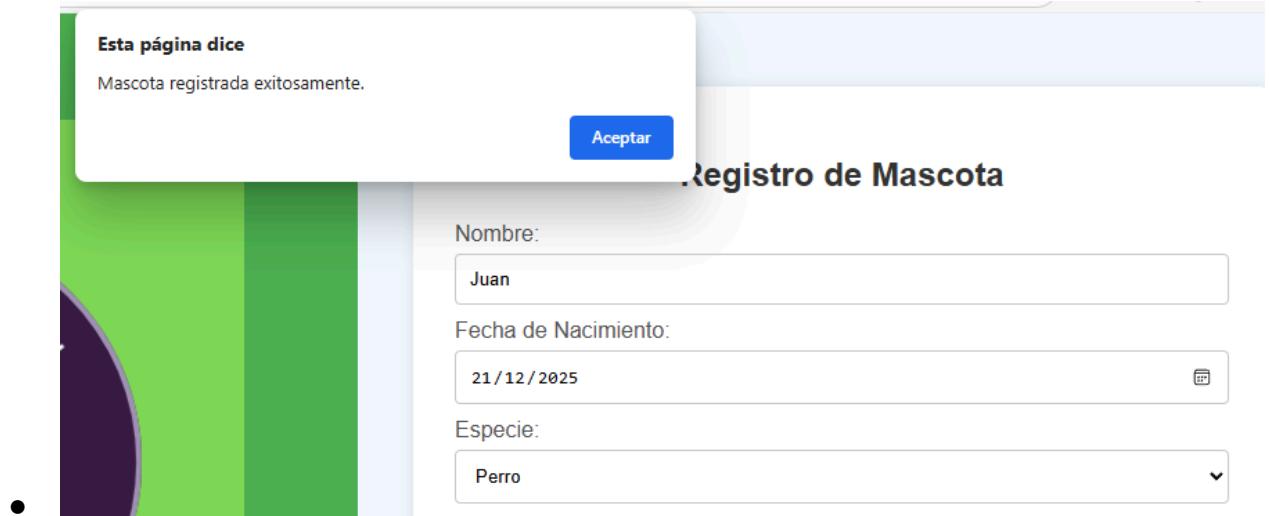
[Registrar Vacunas](#)

[Ver Dashboard](#)

- El botón “Registrar Vacunas” redirige a “registro_vacunas.html”.

Oportunidades de mejora:

- Cambiar las alertas por mensajes en pantalla.
- Validar que la fecha de nacimiento no sea futura:



A screenshot of a web application showing a successful pet registration. On the left, a green sidebar features a circular profile picture of a dog. A white modal window titled "Esta página dice" (This page says) contains the message "Mascota registrada exitosamente." (Pet registered successfully.) with a blue "Aceptar" (Accept) button. To the right, the main "Registro de Mascota" (Pet Registration) page is visible, showing fields for Nombre (Name) containing "Juan", Fecha de Nacimiento (Birth Date) containing "21/12/2025", and Especie (Species) containing "Perro" (Dog). The entire interface has a light blue and white color scheme.

- Mejorar la coherencia del flujo al volver al registro de cliente (mantener información previa).

Misión 3: Registro de Vacunas

Archivo: registro_vacunas.html

Objetivo: Verificar el registro de vacunas, la validación de campos y el funcionamiento de los botones “Cancelar” y “Salvar”.

Pruebas realizadas:

1. Enviar formulario vacío.
2. Completar parcialmente los campos.
3. Completar correctamente todos los campos.
4. Usar el botón “Cancelar”.

Resultados observados:

- Al intentar guardar la vacuna con campos obligatorios sin completar, el navegador muestra el mensaje nativo “Rellena este campo.” sobre el primer campo vacío (por ejemplo, “Nombre de la Vacuna”). Esta validación impide el envío del formulario mientras existan campos requeridos sin datos, por lo que cumple correctamente su función básica de obligatoriedad.

Registro de Vacunas

Nombre de la Vacuna:

Dosis:

Seleccionar ! Rellena este campo.

Fecha de Aplicación:

 dd / mm / aaaa [calendario]

Fecha de Vencimiento:

 dd / mm / aaaa [calendario]

Cancelar

Salvar

Ver Dashboard

- Registro exitoso de vacuna

Cuando se completan todos los campos con valores válidos (Nombre, Dosis, Fecha de Aplicación y Fecha de Vencimiento), al presionar “Salvar” aparece una alerta con el mensaje “Vacuna registrada exitosamente.”. Al aceptar la alerta, el sistema redirige a [registro_mascota.html](#), lo que permite continuar el flujo desde el contexto de la mascota.

Esta página dice

Vacuna registrada exitosamente.

Aceptar

Registro de Vacunas

Nombre de la Vacuna:

Dosis:

Fecha de Aplicación:

Fecha de Vencimiento:

[Cancelar](#)

[Salvar](#)

[Ver Dashboard](#)



- Comportamiento del botón “Cancelar”:
Al presionar “Cancelar” con datos ya ingresados, el sistema muestra un mensaje de confirmación del estilo “¿Estás seguro de que deseas cancelar el registro de la vacuna?”.
Si el usuario confirma, los campos del formulario se limpian y se produce la redirección correspondiente, evitando que se guarde información parcial por error.

Esta página dice

¿Estás seguro de que deseas cancelar el registro de la vacuna?

Aceptar Cancelar

Registro de Vacunas

Nombre de la Vacuna:

Dosis:

Fecha de Aplicación:

Fecha de Vencimiento:

[Cancelar](#)

[Salvar](#)

[Ver Dashboard](#)

Registro de Mascota

Nombre:

Fecha de Nacimiento:

Especie:

[Cancelar](#)

[Registrar Mascota](#)

[Registrar Vacunas](#)

[Ver Dashboard](#)

Oportunidades de mejora

- Validar coherencia entre fechas:
Implementar reglas para que la Fecha de Vencimiento sea posterior a la Fecha de Aplicación. Actualmente es posible ingresar fechas sin relación lógica, lo que puede afectar la calidad de los datos y el seguimiento del calendario de vacunación.
- Definir un flujo de redirección más claro tras el guardado:
Confirmar si, luego de registrar una vacuna, la redirección más adecuada es:
 - volver al perfil de la mascota,
 - mostrar un listado de vacunas
 - permanecer en el mismo formulario con un mensaje de éxito.Esto ayudaría a que el usuario entienda mejor dónde continuar trabajando.

Sustituir las alertas por mensajes visuales dentro del formulario:

Reemplazar los `alert()` por mensajes incrustados en la interfaz (por ejemplo, banners verdes de éxito y rojos de error) que:

- no bloqueen la interacción,
- permanezcan visibles unos segundos,
- sean consistentes con los formularios de Cliente y Mascota.

- Resaltar los campos con error de forma más explícita:
Además del mensaje “Rellena este campo.”, sería conveniente marcar en rojo el borde de los campos vacíos o inválidos. Esto mejora la usabilidad y hace más evidente dónde debe corregir el usuario.
- Mostrar el contexto de la mascota asociada:
Para evitar confusión, el formulario de vacunas podría mostrar en el encabezado el nombre o ID de la mascota a la que se le está registrando la

vacuna. Esto refuerza la trazabilidad entre mascota y vacuna.

Conclusiones Generales

En base al testing exploratorio y al retesting realizado sobre los formularios de Cliente, Mascota y Vacunas, se pueden extraer las siguientes conclusiones:

Validaciones

En los tres formularios las validaciones básicas de campos vacíos funcionan correctamente: el sistema impide el envío cuando falta información obligatoria y muestra mensajes que orientan al usuario.

Sin embargo, se detectó que:

- No se aplican validaciones de contenido más avanzadas (formato de correo, longitud mínima de nombres, coherencia entre fechas, etc.).
- Todavía es posible registrar datos poco confiables, como correos inválidos, nombres de un solo carácter o fechas sin sentido.

Esto impacta directamente en la calidad e integridad de los datos almacenados.

Flujo de navegación

El flujo principal entre pantallas (Registro de Cliente → Registro de Mascota → Registro de Vacunas → Dashboard) es en general correcto y las redirecciones funcionan, pero se observaron algunas inconsistencias:

- Desde el registro de cliente, tras guardar, se redirige a un [login.html](#) con credenciales precargadas y mensaje de error ya visible.
- El botón “Ver Dashboard” no muestra el listado de clientes, sólo mascotas y vacunas, lo que dificulta verificar los registros realizados.
- El comportamiento del checkbox “Ingresar Mascota” no está integrado al flujo y genera expectativas que no se cumplen.

En resumen, el flujo existe y es utilizable, pero necesita ajustes para ser más coherente, predecible y alineado con los RF/CU definidos.

Mensajes al usuario

Los mensajes actuales comunican el resultado de las acciones (éxito o error), pero se apoyan casi exclusivamente en:

- Alertas del navegador (`alert`)
- Mensajes genéricos como “Por favor, complete todos los campos.”

Esto provoca una experiencia algo básica y poco moderna. Sería deseable:

- Incorporar mensajes visuales dentro del formulario (banners, textos debajo de los campos, colores).
- Mostrar mensajes más específicos, indicando qué campo tiene el problema y por qué.

Con estos cambios, el sistema ganaría en claridad, profesionalismo y facilidad de uso.

Usabilidad y experiencia general

El diseño visual es simple, claro y consistente entre pantallas, lo cual es un buen punto de partida. Aun así, se identifican oportunidades de mejora:

- Alinear correctamente controles como el checkbox “Ingresar Mascota”.
- Resaltar en rojo los campos con error para guiar mejor la corrección.
- Mostrar más contexto en pantallas clave (por ejemplo, en vacunas indicar a qué mascota pertenece).

- Completar el Dashboard con el listado de clientes para que realmente funcione como vista resumen del sistema.

Síntesis

El sistema, en su estado actual, permite completar el flujo básico de registro y cuenta con un primer nivel de validaciones y navegación. No obstante, para avanzar a una etapa más madura de producto se recomienda:

- Fortalecer las validaciones de datos,
- Ajustar el flujo de navegación y las redirecciones,
- Mejorar la comunicación visual de errores y éxitos,
- Y pulir aspectos de usabilidad y trazabilidad entre clientes, mascotas y vacunas.

Estas mejoras aumentan la confiabilidad del sistema y ofrecerían una experiencia más robusta y profesional para los usuarios finales.

Acerca de la Base de Datos

Durante la Etapa 3 se verificó nuevamente el script original actualizado con el nombre de base **VeterinariaGuauGuau**.

En SQL Server, el DDL se ejecuta correctamente y crea las tablas **Clientes**, **Mascotas** y **Vacunas** sin errores.