

Actividad: Requisitos Mediante Mapeo de Historias de Usuario y Wireframes

Kalio Fernando O'Farril Villalpando

Plataforma de Desarrollo de Software

Grupo 1001

Requisitos

Pet-Clinic es un proyecto de ejemplo clásico para aprender a programar con el *framework* Spring de Java.

- Identificar y analizar qué plataforma de gestión de requisitos que soporte historias de usuario se adapta a tu necesidad.
- Definir con la misma tu visión de *user story mapping* de las funcionalidades que tiene el Pet-Clinic. Ordénalas en un mapa de historias de usuario para cubrir al menos dos versiones, una que fuera un producto mínimo viable y otra la actual.
- Detallar al menos una historia de usuario con información más detallada que explique la funcionalidad desde la perspectiva del usuario.
- Identificar y analizar qué plataforma de *wireframes* se adapta a tu necesidad.
- Diseñar un *wireframe* de la pantalla de la ficha de un propietario de mascota (no tiene que coincidir exactamente con la que se ha desarrollado, puedes plantear tu solución).
- Documentar el proceso y justificar tu elección de plataformas y tu aprendizaje y dificultades.

Plataforma de Gestión

Para la planeación y seguimiento del proyecto PetClinic se decidió utilizar Jira. Jira es un producto de software propietario para la gestión de proyectos, seguimiento de errores e incidencias. Es usado en muchas empresas a nivel global y su costo es bajo en comparación de otros según los criterios de Carillo de Gea et al [1]. Siguiendo estos criterios Jira obtiene una puntuación de 19 según la Tabla 1.

Tool name	DOORS DNG	DOORS	Jira	Siemens Polarion	PTC Integrity	Regility	Quality Center	ie-STEP Blue	PREvision	Enterprise Architect	Jama Connect	RMTrak	Microsoft Office
Version, date	7.0.3, December 2020	9.7.2, July 2020	8.14.0, November 2020	20R2, October 2020	12.2.1.0, October 2019	2020x, November 2019	15.5, September 2020	6.5, January 2020	9.5, 2021	15.2, October 2020	8.563, November 2020	5.3.22, 2017	365, January 2021
Vendor name	IBM	IBM	Atlassian	Siemens	PTC	Dassault	Micro Focus	Micro-TOOL	Vector	Sparx Systems	Jama Software	Pro-engineer	Microsoft
Cost	High	High	Low	High	High	Middle	High	High	High	Middle	Middle	Middle	Low
OR	+++	+++	0	+++	++	+	+	+	++	++	++	+	++
RDOIL	++	+	+	++	++	+	++	+	++	++	++	+	+
IC	++	++	+	++	++	+	+	+	++	++	++	+	0
TS	+++	+++	+	+++	+++	+	+	+	++	+++	+++	+	0
VM	+++	++	0	+++	++	0	0	+	+++	+	+++	0	0
RWC	++	+	+	++	++	0	+++	++	+++	+++	++	0	++
MV	+++	++	0	+++	+++	+	+	+	+++	+++	+++	+	+
Perf	++	+	++	++	++	+	+	0	++	++	++	0	+
CWM	+++	+	++	+++	++	+	++	+	+++	+++	+++	+	++
IBP	++	+	++	++	++	+	+	+	++	++	++	0	++
FI	+++	+++	++	+++	+++	0	+	+	+++	+++	+++	0	+
EI	++	++	+	++	++	+	+	++	++	++	++	+	+
MaC	++	++	++	++	++	0	+	0	++	++	++	+	++
TLC	+	+	+	+	+	+	++	+	+	+	+	+	++
DevOps	+	+	++	++	+	+	++	+	+	+	+	0	0
IS	+	0	0	+	0	0	0	0	0	++	0	0	0
Scal	+++	+	+	+++	+++	+	+++	+	++	+++	+++	+	++
Total	38	27	19	39	34	12	23	16	35	37	37	10	19

Values include "0", "+", "++", "+++" from minimum support to full support.

Tabla 1. Evaluación de herramientas de Ingeniería de Requisitos. Fuente: Carrillo de Gea et al., 2021.

La puntuación obtenida está basada en las siguientes funcionalidades:

- Informes, consultas a bases de datos y lenguaje de interfaz abierta (RDOIL): la herramienta es capaz de generar informes estructurados, sofisticados y/o estandarizados a partir de los datos.
- Controles internos, es decir, consistencia, dependencias e historia (IC): la herramienta ofrece mecanismos para la detección y resolución de conflictos entre requisitos y/o verificaciones de consistencia.
- Apoyo a la trazabilidad, es decir, arrastrar y soltar (horizontal y verticales) (TS): la herramienta ofrece soporte a trazabilidad visual y/o diferentes tipos de trazas.
- Trabajo remoto, solo en la nube (RWC): la herramienta puede funcionar sin conectividad.

- Rendimiento (Perf): la herramienta puede gestionar grandes proyectos con grandes cantidades de datos y está optimizada para cargar grandes modelos.
- Colaboración, gestión del *workflow* (CWM): la herramienta brinda soporte para el trabajo colaborativo y/o distribuido, ofrece funcionalidad de revisión, discusión y/o voto —así como comunicación asíncrona y/o síncrona—, incluye control de acceso y seguridad granular. Proporciona flujos de trabajo (*workflows*) predefinidos, flexibles y personalizables.
- Fácilmente adaptable e integrable en los procesos de negocio (IBP): la herramienta es personalizable y flexible para adaptarse a las condiciones del proyecto, los *stakeholders* o la naturaleza de los requisitos. Puede adaptarse fácilmente a cualquier tipo de proceso de negocio o industria.
- Federación y notificación con herramientas *application and product lifecycle management* (ALM/PLM) (FN): la herramienta ofrece funcionalidad ALM/PLM. Ofrece la posibilidad de integrarse con soluciones ALM/PLM de terceros.
- Exportación/Importación con formatos estándar (EI): la herramienta ofrece una funcionalidad completa de importación/exportación y soporte para formatos estándar: CSV.
- Macros para repetir comandos (MaC): la herramienta incluye la funcionalidad de creación y gestión de macros. Proporciona un lenguaje de scripting.
- Esfuerzo de formación y curva de aprendizaje (TLC): la herramienta incluye documentación adecuada, tutoriales, materiales de capacitación y recursos.
- *Agile*, integración y distribución continuas (CI/CD) y desarrollo y operaciones (DevOps): la herramienta está concebida o es fácilmente adaptable a metodologías ágiles y *pipelines* DevOps. Permite la gestión de requisitos en marcos ágiles, *lean* o *scaled agile*.
- Escalabilidad (Scal): la herramienta ofrece soporte básico a grandes proyectos.

A pesar de no tener la puntuación más alta, para la planeación de Sprints e Historias de Usuario Jira cubre con todas las necesidades de este pequeño proyecto. Con una amplia documentación y una interfaz gráfica sencilla e intuitiva puede agilizar el proceso de onboarding al equipo, sin dejar fuera las necesidades del mismo.

User Story Mapping

Se ha creado un proyecto en Jira llamado “petclinic-kalio”. Se ha propuesto una duración de 4 Sprints, cada uno de dos semanas, para la entrega del proyecto completo. El MVP (Minimum Viable Product) será completado en las primeras dos iteraciones o Sprints, mientras que las tareas PGL (Post Go-Live) que darán por completado el proyecto serán elaboradas durante la tercera y cuarta iteración. El User Story Map en forma de un Timeline, mostrado en la Figura 2, puede ser vistos mediante el siguiente enlace:

<https://kalioofvil.atlassian.net/jira/software/projects/PETCLINIC/boards/1/timeline>

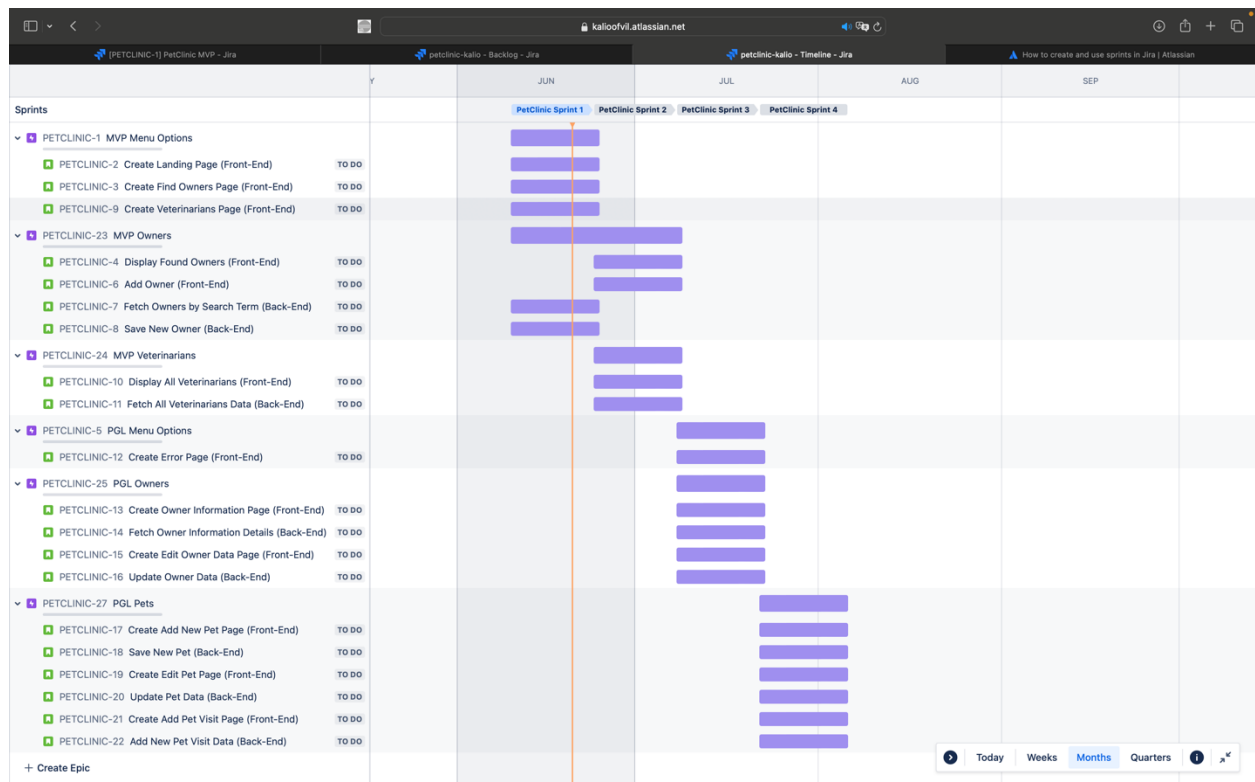
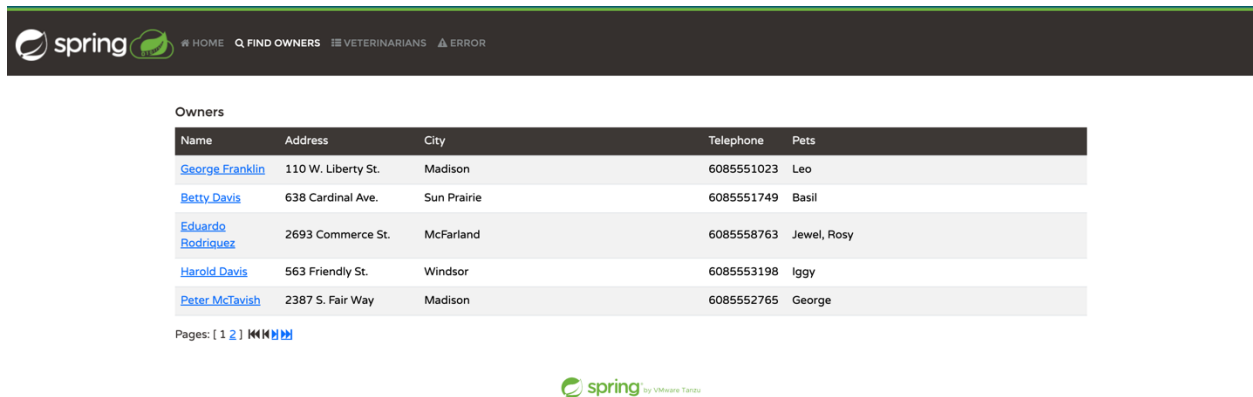


Figura 2. User Story Map para PetClinic-Kalio.

MVP

Para el MVP se ha considerado como la página principal así como la de dueños y veterinarios pero con funcionalidades básicas. Los detalles de los dueños y veterinarios serán accesibles más no editables y no detallados.

En la página de dueños: Se podrá buscar entre los dueños existentes para desplegar sus datos básicos, como vienen siendo Nombre, Dirección, Ciudad, Teléfono y Mascotas. La Figura 3 muestra la vista preliminar de ello.



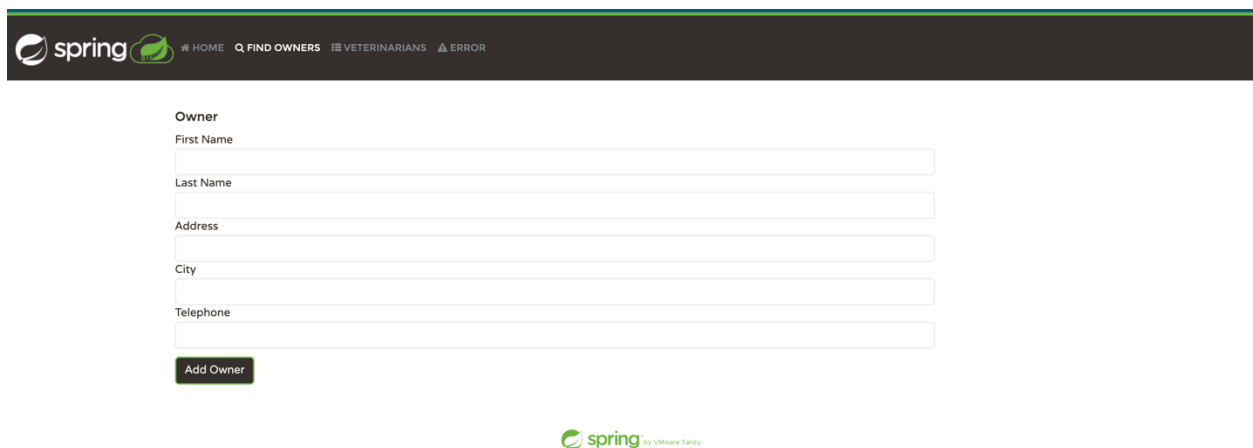
Name	Address	City	Telephone	Pets
George Franklin	110 W. Liberty St.	Madison	6085551023	Leo
Betty Davis	638 Cardinal Ave.	Sun Prairie	6085551749	Basil
Eduardo Rodriguez	2693 Commerce St.	McFarland	6085558763	Jewel, Rosy
Harold Davis	563 Friendly St.	Windsor	6085553198	Iggy
Peter McTavish	2387 S. Fair Way	Madison	6085552765	George

Pages: [1 2]

spring by VMware Tanzu

Figura 3. Tabla con datos básicos de los dueños.

Se podrá agregar nuevos dueños con los mismos datos básicos a través de una forma. La Figura 4 ejemplifica la forma recién mencionada.



Owner

First Name

Last Name

Address

City

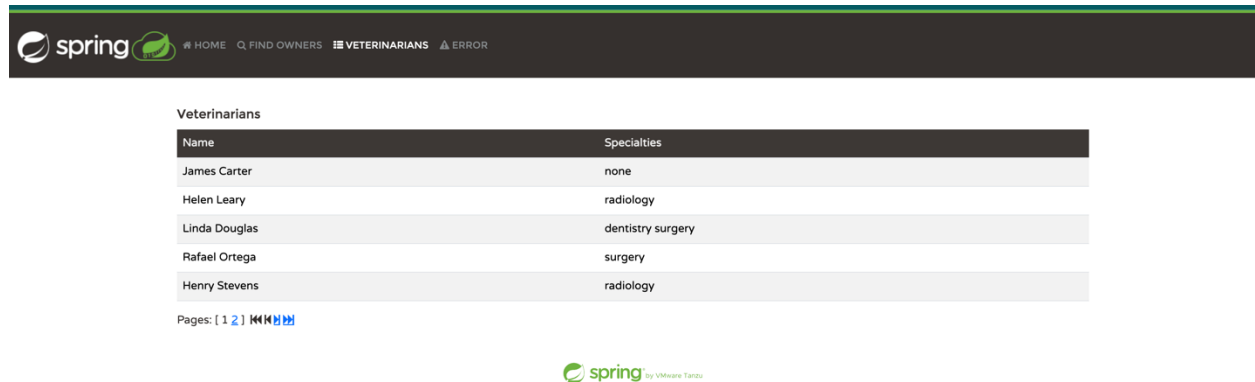
Telephone

Add Owner

spring by VMware Tanzu

Figura 4. Forma para agregar nuevos dueños.

En términos de los veterinarios, el MVP incluye la página que mostrará todos los veterinarios disponibles en la clínica, como puede ser visto en la Figura 5.



Name	Specialties
James Carter	none
Helen Leary	radiology
Linda Douglas	dentistry surgery
Rafael Ortega	surgery
Henry Stevens	radiology

Pages: [1 2]

Figura 5. Tabla mostrando los veterinarios de la clínica.

Con estas funcionalidades que pueden ser entregadas en dos Sprints al cliente, se podrá tener datos de los dueños así como de los veterinarios. El contacto con los dueños es primordial para las operaciones día a día de la clínica y por eso son considerados dentro de los primeros entregables.

PGL

Los PGL (Post Go-Live) Enhancements ampliarán los datos de los dueños dando cabida a más detalles como pueden ser los datos de sus mascotas y las citas que han tenido de manera histórica. Además, se agrega la opción de editar los datos ya existentes de los dueños, incluyendo sus mascotas, en el caso de que cambien de dirección, teléfono, etc. También se integra una página de error en el caso de que alguno de los datos agregados no haya sido podido ser procesado adecuadamente.

La página de detalles del dueño mostrará los datos básicos ya mostrados en la tabla principal pero expande en los datos de la mascota. Las posibles acciones relacionadas a las mascotas o a la edición de datos serán accesibles mediante botones e hipervínculos mostrados en esta pantalla, como puede ser visto en la Figura 6. En la Figura 7 se muestra la forma en la que se pueden editar los datos de los dueños.

The screenshot shows the Spring application interface. At the top is a dark navigation bar with the Spring logo and links: HOME, FIND OWNERS, VETERINARIANS, and ERROR. Below this, the 'Owner Information' section displays a table with the following data:

Name	George Franklin
Address	110 W. Liberty St.
City	Madison
Telephone	6085551023

Below the table are two buttons: 'Edit Owner' and 'Add New Pet'. The 'Pets and Visits' section shows a table with one pet entry:

Name	Leo	Visit Date	Description
Birth Date	2010-09-07	Edit Pet	Add Visit
Type	cat		

At the bottom of the page is the Spring logo with the text 'by VMware Tanzu'.

Figura 6. Página con detalles de los dueños y botones e hipervínculos a demás acciones.


The screenshot shows the Spring application interface with the 'Owner' edit form. The form contains the following fields:

- First Name: George
- Last Name: Franklin
- Address: 110 W. Liberty St.
- City: Madison
- Telephone: 6085551023

Below the form is an 'Update Owner' button. At the bottom of the page is the Spring logo with the text 'by VMware Tanzu'.

Figura 7. Forma para editar los datos de un dueño.

Al darle click en el botón de agregar una mascota a un dueño, una forma en donde los datos de la mascota puede ser insertados será mostrada (Figura 8). Al darle click en el botón de guardar los datos serán almacenados en la base de datos. La misma forma será reutilizada a la hora de editar los datos de la mascota, como se muestra en la Figura 9. Por último, se agregará una forma más en donde se llevará el registro y adición de las citas de las mascotas a la clínica. Ésto es visualizado en la Figura 10.



[HOME](#) [FIND OWNERS](#) [VETERINARIANS](#) [ERROR](#)

New Pet

Owner

George Franklin

Name

Birth Date

dd/mm/yyyy

Type

bird

Add Pet



 by VMware Tanzu

Figura 8. Forma para guardar una nueva mascota.



[HOME](#) [FIND OWNERS](#) [VETERINARIANS](#) [ERROR](#)

Pet

Owner

George Franklin

Name

Leo

Birth Date

07/09/2010

Type

cat

Update Pet



 by VMware Tanzu

Figura 9. Forma para editar los datos existentes de una mascota.



[HOME](#) [FIND OWNERS](#) [VETERINARIANS](#) [ERROR](#)

New Visit

Pet

Name	Birth Date	Type	Owner
Leo	2010-09-07	cat	George Franklin

Date

21/06/2024

Description

Add Visit

Previous Visits

Date	Description
------	-------------


 by VMware Tanzu

Figura 10. Registro e historial de las citas de una mascota.

Ejemplo de Historia de Usuario

Como ejemplo de los requisitos para detallar una historia de usuario de manera adecuada en el proyecto hemos detallado la historia para la creación de la ficha del propietario de mascota (Create Owner Information) y puede ser vista a través del hipervínculo [PETCLINIC-13](#).

Se han puesto tres secciones obligatorias para la correcta documentación de la historia: Resumen, Descripción y Criterios de Aceptación. La historia tiene una épica asignada, que es la fase del proyecto a la cual la historia depende, en este caso es “PGL Owners”. Además se han agregado los campos de estimado de puntos de historia en la cual quien vaya a trabajar en ella puede dar un estimado de cuanto esfuerzo le requerirá terminar dicha actividad. La historia es después agregada a un Sprint en donde se han contemplado las secuencias de las tareas así como su estimada duración y disponibilidad del equipo. Por último, se ha agregado una imagen del wireframe correspondiente a la historia para tener claridad de como se espera el resultado.

Los detalles de la historia en cuestión son los siguientes.

Resumen: Como usuario de la aplicación “PetClinic” quiero poder visualizar, agregar y editar información detallada de los dueños y sus mascotas para llevar un correcto control de los pacientes y su historial y datos de contacto.

Descripción: Al dar click en el nombre del dueño en la tabla principal de “Dueños”, se debe redirigir a una nueva página donde desplegará los datos generales del usuario en una tabla. Debajo de esta tabla deben encontrarse dos botones; uno para editar la información del usuario y otro para agregar una mascota al su perfil. Continuando en el camino descendente en la página, se mostrará la información de las mascotas así como de sus respectivas visitas.

Criterios de Aceptación:

- La tabla debe mostrar los datos de nombre, dirección, ciudad y teléfono del dueño seleccionado.
- El botón de “editar usuario” y “agregar nueva mascota” deben estar presentes bajo la tabla de datos del dueño.

- Una sección “Mascotas y Visitas” debe estar por último en la página, teniendo dos columnas.
- La primera columna de “Mascotas y Visitas” mostrará los datos de nombre, fecha de nacimiento y tipo de mascota.
- La segunda columna tendrá la información de la fecha de las visitas así como la descripción de cada una.
- Debajo de las visitas debe haber dos hipervínculos que redirigirán a “Editar mascota” y “Agregar visita”.

En la Figura 11 podemos ver una captura de pantalla de la historia detallada en la plataforma Jira.

Projects / petclinic-kalio / PETCLINIC-25 / PETCLINIC-13

Create Owner Information Page (Front-End)

Attach Add a child issue Link issue

Summary
Como usuario de la aplicación "PetClinic" quiero poder visualizar, agregar y editar información detallada de los dueños y sus mascotas para llevar un correcto control de los pacientes y su historial y datos de contacto.

Description
Al dar click en el nombre del dueño en la tabla principal de "Dueños", se debe redirigir a una nueva página donde desplegará los datos generales del usuario en una tabla. Debajo de esta tabla deben encontrarse dos botones; uno para editar la información del usuario y otro para agregar una mascota al su perfil. Continuando en el camino descendente en la página, se mostrará la información de las mascotas así como de sus respectivas visitas.

Acceptance Criteria

- La tabla debe mostrar los datos de nombre, dirección, ciudad y teléfono del dueño seleccionado.
- El botón de "editar usuario" y "agregar nueva mascota" deben estar presentes bajo la tabla de datos del dueño.
- Una sección "Mascotas y Visitas" debe estar por último en la página, teniendo dos columnas.
- La primera columna de "Mascotas y Visitas" mostrará los datos de nombre, fecha de nacimiento y tipo de mascota.
- La segunda columna tendrá la información de la fecha de las visitas así como la descripción de cada una.
- Debajo de las visitas debe haber dos hipervínculos que redirigirán a "Editar mascota" y "Agregar visita".

Attachments 1

Wireframe - FI... rio.png
21 Jun 2024, 10:42 PM

Activity
Show: All Comments History Summarize Newest first 17

Add a comment...

Pro tip: press **M** to comment

Pinned fields
Click on the next to a field label to start pinning.

Details

Assignee	Kalio Ofarril
Labels	PetClinicPGL
Parent	PETCLINIC-25 PGL Owners
Sprint	PetClinic Sprint 3
Story point estimate	3
Reporter	Kalio Ofarril

Created 2 days ago
Updated 13 hours ago

Configure

Figura 11. Historia de usuario para la ficha de propietario de mascota.

Plataforma de Wireframe

Un wireframe es un modelo o marco esquelético que describe el diseño básico y las funciones de una interfaz de usuario. Para el proyecto se ha decidido usar Balsamiq como plataforma de Wireframe. La facilidad de arrastar objetos y colocarlos uno sobre otro así como la posibilidad de cambiar el contenido de ellos con formatos sencillos hace rápido y sencillo el proceso de inducción a la plataforma. Ya que a diferencia de un mockup, no queremos invertir tiempo en decisiones de colores, tipos de fuente o referencias a la marca del producto, se ha optado por tomar Balsamiq que ayuda al no proponer estas características como algo requerido.

Al tener una versión cloud, Balsamiq puede garantizar la disponibilidad de los wireframes a todos los miembros del equipo a todo momento con tan solo tener conexión a internet. Aún más, al tener un costo asequible e inclusive versiones básicas gratuitas, no representa un impacto al coste del proyecto.

El proyecto puede ser encontrado en: <https://balsamiq.cloud/sxy2xs2/p149ijc/r1D8o>

Ejemplo de Wireframe

Usando la plataforma de wireframes mencionada, se ha creado el wireframe para la página de ficha de propietario de mascota (Figura 12). Cuenta con el Header que permite al usuario navegar a las distintas páginas de la app. Los primeros datos a desplegar es la información del dueño, la cual es mostrada en una tabla. Debajo de dicha tabla hay dos botones que permiten al usuario editar la información del dueño así como agregar una mascota asociada al dueño. A continuación se tiene una sección de Mascotas y sus respectivas visitas. Se tiene un Scroll-Bar vertical en esta sección pues cada dueño puede tener n número de mascotas. Cada contenedor de mascota tiene dos columnas: la primera conteniendo información básica de la mascota y la derecha de sus visitas. La tabla de las visitas también cuenta con un Scroll-Bar vertical ya que cada mascota puede tener múltiples visitas. Por último, debajo de la tabla de visitas de la mascota hay dos hipervínculos que redirigen a las páginas para editar los datos de la mascota o para agregar una visita a su historial.

PetClinic

https://petclinic.pet

PetClinic Inicio Dueños Veterinarios Error

Información del Dueño

Nombre	Kalio O'Farril
Dirección	La Casa Blanca
Ciudad	Guadalajara
Teléfono	3310884640

[Editar Dueño](#) [Agregar Mascota](#)

Mascotas & Visitas

Nombre	Pogo
Fecha de Nacimiento	10 Noviembre 1990
Tipo	Perro

Fecha de Visita	Descripción
1 Enero 1990	Vacunación.
10 Noviembre 1991	Orquiectomía
29 Febrero 2000	Vacunas.
10 Junio 2012	Baño

[Editar Mascota](#) [Agregar Visita](#)

Nombre	Gala
Fecha de Nacimiento	10 Noviembre 2000
Tipo	Perro

Fecha de Visita	Descripción
1 Enero 1990	Vacunación.
10 Noviembre 1991	Orquiectomía
29 Febrero 2000	Vacunas.

Figura 12. Wireframe para la ficha de propietario de mascota.

Conclusión y Aprendizaje

El proceso de tomar múltiples papeles en la planeación de una aplicación fue una actividad divertido e ilustrativa. Como desarrollador he interactuado con historias de usuario y mockups más nunca había estado a cargo de crear alguna. El proceso de diseccionar una aplicación ya hecha y hacer historias de usuario basandome en deducciones ha sido una actividad que siento ha apoyado mi deseo de convertirme en un arquitecto de software más redondo.

A pesar de saber usar Jira a un nivel de aditar historias y moverlas a través de los flujos del Sprint. Nunca había creado un proyecto y haber asociado historias a épicas, aprendí varias formas de configuración que seguro serán de provecho en mi ambiente laboral en un futuro.

No conocía el concepto de wireframes, solo de mockups. Siempre había sentido que cuando quería hacer aplicaciones propias, a la hora de UI y UX se me iba mucho tiempo ya que nunca he sido bueno en cosas como elecciones de colores y figuras. El hecho de poder crear el wireframe pensando solamente en el posicionamiento de los elementos y su funionalidad me ha quitado un peso de encima y podría ayudar a colaborar con algún diseñador de una manera más cercana, llegando a un producto más similar a como lo concibo en la cabeza.

En general esta actividad fue una buena y pequeña práctica de como utilizar herramientas nuevas y de pulir mis habilidades en otras ya conocidas.

Bibliografía

[1] Carrillo de Gea, J. M., Ebert, C., Hosni, M., Vizcaíno, A., Nicolás, J., y Fernández-Alemán, J. L. (2021). Requirements engineering tools: An evaluation. *IEEE*, 38(3), 17-24.

[2] Jacobson, I., Booch, G. y Rumbaugh, J. (2000). *El proceso unificado de desarrollo de software*. Addison Wesley.

[3] Jira. (2024, 20 de Junio). Project guide to Jira.

(<https://www.atlassian.com/software/jira/guides/getting-started/introduction#what-is-jira-software>)

[4] Balsamiq. (2024, 20 de Junio). Creating your first Wireframe.

(<https://balsamiq.com/tutorials/articles/firstwireframe/>)