**ESCUELA POLITÉCNICA NACIONAL**

**ESCUELA DE FORMACIÓN DE TECNÓLOGOS**

**DESARROLLO DE UNA APLICACIÓN PARA LA ADOPCIÓN DE PERROS Y GATOS LLAMADA PET MAP**

**PROYECTO PREVIO PARA LA FINALIZACIÓN DEL SEMESTRE EN LA MATERIA FUNADAMENTOS DE DESARROLLO DE SOFTWARE**

**YOMARA ISABEL DIAZ OSCULLO**

yomara.diaz@epn.edu.ec

**ELVIS LENIN AGILA PALADINES**

elvis.agila@epn.edu.ec

**GUILLERMO ALEXANDER RIVERA GUERRERO**

guillermo.rivera@epn.edu.ec

**CARLOS ANDRE MONTESDEOCA GARCIA**

carlos.montesdeoca@epn.edu.ec

**DIRECTOR: Ing. Juan Pablo Zaldumbide Proaño, MSc.**

Quito, Septiembre 2020

**INDICE DE CONTENIDO**

**RESUMEN**

1. INTRODUCCIÓN…………………………………………………………………
   1. Objetivo general ……………………………………………………………...
   2. Objetivos Específicos…………………………………………………………
   3. Alcance………………………………………………………………………..
   4. Justificación…………………………………………………………………...
2. METODOLOGÍA ………………………………………………………………….
   1. Metodología de Desarrollo Scrum…………………………………………….
      1. Fases deSrum…………………………………………………………….
      2. Beneficios de la Metodología Scrum…………………………………….
      3. ¿Qué es un Sprint? ………………………………………………………
      4. Roles …………………………………………………………………….
      5. Fundamentación…………………………………………………………
   2. Diseño de Interfaces del Sistema web………………………………………...

2.2.1. Prototipo del sistema……………………………………………………

* 1. Diseño de la arquitectura del sistema Web……………………………………
  2. Herramientas para la implementación…………………………………………

2.4.1. Herramientas para las bases de datos…………………………………..

2.4.2. Herramientas para el desarrollo del sistema Web……………………………..

1. RESULTADOS ……………………………………………………………………
   1. Requerimientos Funcionales………………………………………………….
   2. Usuarios del Sistema Web ……………………………………………………
   3. Restricciones de desarrollo……………………………………………………
   4. Consideraciones……………………………………………………………….
   5. Conformación del equipo de trabajo………………………………………….
   6. Actividades……………………………………………………………………

3.6.1. Historias de Usuario……………………………………………………

3.6.2. Sprint de Desarrollo……………………………………………………

* 1. Pruebas en el sistema Web……………………………………………………

3.7.1. Pruebas de Caja Negra…………………………………………………

3.7.2. Pruebas de Compatibilidad…………………………………………….

3.7.2. Pruebas de interfaz con el usuario …………………………………….

1. CONCLUSIONES Y RECOMENDACIONES…………………………………...
   1. Conclusiones………………………………………………………………….
   2. Recomendaciones…………………………………………………………….

**RESUMEN**

Este proyecto corresponde al desarrollo de un sistema web como medida de ayuda humanitaria para animales que se encuentran en estado de vulneración en la calle.

En el presente se utilizó metodología Scrum como metodología principal para el trabajo de los desarrolladores, debido a su eficiencia dentro de os grupos de trabajo. En el presente documento se detallan 4 secciones que detallan al proyecto.

En la primera sección se presenta un estudio sobre animales de la calle en Quito, dicho estudio permitió generar un objetivo y alcances, dando el significado a la pregunta del por qué se desarrolló el sistema.

En la segunda sección se detallan los beneficios que brinda el sistema, los roles entre los miembros del equipo y se da una introducción acerca de la metodología scrum.

En la tercera sección se muestra el proceso de desarrollo del Sistema Web, incluyendo los resultados, análisis y pruebas de compatibilidad que se realizaron.

Para concluir con la cuarta sección se presentan las conclusiones y las recomendaciones que se obtuvieron del desarrollo del proyecto.

**Palabras clave:** Desarrollo, Scrum.

1. **INTRODUCCIÓN**
   1. **Objetivo general**

Ayudar a los animales en estado de vulnerabilidad teniendo una participación activa de la sociedad y fundaciones que velan por el bien y seguridad de estos animales, para darles un estilo de vida adecuado con personas responsables y calificadas para la tenencia de animales.

* 1. **Objetivos específicos**
* Registrar la mayor cantidad de animales en estado de calle o abandono.
* Identificar las zonas con mayor cantidad de animales en situación de calle.
* Dar un hogar seguro a los animales rescatados asegurando que estarán en manos de personas responsables.
* Fomentar la ayuda ciudadana por medio de campañas para la concientización del cuidado de animales.
* Ayudará a las fundaciones por medio de donaciones monetarias o donaciones físicas.
* Desarrollar una aplicación amigable que ayude a facilitar las gestiones para reportar los animales en estado de calle.
  1. **Alcance**
  2. **Justificación**

1. **METODOLOGÍA**
   1. **Metodología de Desarrollo Scrum**

Esta metodología se origen gracias a que en el año 1986 dos investigadores tomaron como análisis a un grupo de empresas tecnológicas que tenían un tipo de metodología muy eficaz, con ella era muy notorio que los proyectos realizados salían en un tiempo inferior en comparación a los demás y no solo eso, sino también la calidad era notablemente mejor.

La metodología se basaba en un equipo con habilidades múltiples que trabajan de principio a fin del proyecto, este tipo de trabajo de equipo es lo que se comenzó a denominar SCRUM.

SCRUM es un método que permite recolectar lo mejor de las buenas prácticas en el desarrollo de software, tiene como principal característica el desarrollo continuo y ciclos de avance cortos, estos ciclos o entregables son conocidos como iteraciones o más propiamente Sprints dentro de la metodología.

* + 1. **Faces de Scrum**

**Concepto:** se definirán las funciones y requisitos generales que tendrá que cumplir el proyecto, y también los roles o encargados del desarrollo.

**Especulación:** es un análisis más detallado de las funciones, se deben establecer los límites que tendrán no solo lógico, sino también físico o sociales dentro del proyecto.

**Exploración:** el proyecto va creciendo en función de las funcionalidades que se vayan añadiendo.

**Revisión:** se debe evaluar el desarrollo de cada Sprint y que los objetivos establecidos sean cumplidos.

**Cierre: se** debe dar la versión inicial o la acordad del producto dentro de los limites establecidos en el inicio del contrato.

* + 1. **Beneficios de la metodología Srum.**

Como se mencionó en un inicio de la documentación es un método más organizado que permita trabajar de forma más dinámica y al aplicarlo en el proyecto da la ventaja de establecer tiempos de entrega, mejorar la calidad y tiempo de entrega de un proyecto.

Al aplicar estos conceptos se puede simular un ambiente profesional de desarrollo lo que mejora el rendimiento y formación profesional de los desarrolladores participantes.

Las entregas son controladas y revisadas por todo el equipo por lo que es muy difícil o casi imposible que un cambio dentro los repositorios de trabajo presente errores graves en el sistema.

Ventajosamente el proyecto entregado en el presente documento no presento conflictos internos dentro del equipo de desarrollo, ya que todas las sugerencias siempre se tomaron en cuenta en las reuniones de los Sprint eso fue de gran ayuda para mejorar no solo funcionalidad sino también apariencia de las interfaces.

* + 1. **¿Qué es un Sprint?**

Los sprint son las iteraciones aplicadas dentro de cada proyecto, en cada una de ellas se establece los limites de los entregables dentro de un tiempo predefinido, esto ayuda mucho a marcar un flujo constante de trabajo, esta es la principal característica que mejora esta metodología frente a otra.

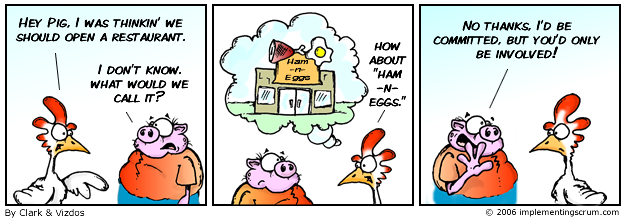
Dentro de los Sprint esta lo que se conoce como Sprint backlog. Cada ves que se realizan los sprit deben ser evaluados por el equipo, tomando en cuenta si se cumplió los objetivos del mismo, si no es así llevara a cabo un pequeño retraso en algunas o acciones o solo un aplazamiento de la funcionalidad que se quiere aplicar en el proyecto.

Los más recomendable es hacer reuniones diarias del Sprint para evaluar los avances del proyecto.

Para el proyecto PetMap se establecieron los limites de sprint de dos semanas, y las reuniones de avances se realizaron cada 2 o 3 dias dependiendo de la complejidad de la funcionalidad implementada.

* + 1. **Roles.**

Para entender un poco más los dos tipos de roles que ahí en este método de desarrollo se puede tomar como referencia este chiste:



1. Comprometidos:   
   Product Owner.- es aquel que tiene conocimiento total de el negocio del producto.  
   SrumMaster.- verrifica que los métodos empleados sean los correctos y den resultados prometedores.

Equipo de desarrollo.- el equipo que codifica , tiene la autoridad para tomar ediciones en función de los objetivos.

1. Involucrados:

Usuarios.- son los consumidores finales del producto.

Stakeholder.- son los beneficiados por el proyecto.

Managers.- toman deciciones finales en las reuniones de toma de objtivos.

Al ser un minirpoyecto semestral PetMap no usa completamente lo roles definidos dentro de la metodología establecida.

El encargado principal en observar el desarrollo de la aplicación es el Ing. Juan Pablo Zaldumbide De la Escuela Politecnica Nacional.(SrumMaster).

El equipo de trabajo estará dentro de los roles de (equipo de desarrollo y Product Owner) ya que PetMap es una ideas establecida y apoyada en total por los 4 integrantes.  
el usuario final del producto tiene que ser un usuario sin un perfil estrictamente definido, los únicos requisitos que debe cumplir son total honestidad al querer ayudar a animalitos de la calle para poder acceder a los servicios de PetMap.

* + 1. **Fundamentacion.**

El proyecto tiene origen gracias a al cariño por los animales que tenían los desarrolladores y lo que busca promover es la adopción de animalitos sin timar en cuenta las razas puras o razas mestizas en los animalitos.

Gracias a algunos proyectos organizados dentro de la Escuela Politécnica Nacional y organizados por la fepoon se conoció un poco de la problemática que existe en quito a cerca de los animales callejeros, lo que se busca no es ofrecer un pagina de anuncian de venta o adopción directa de mascotas el objetivo principal es que los animalitos sean adoptados sin un fin remunerado pero que sean animales que hayan rescatado de las calles.

Según un censo realizado y publicado por el diario EL COMERCIO el 07 de noviembre del 2019, en las calles de quito se estimó que aproximadamente existe un can en estado de abandono por cada 47 habitantes.

Muchas veces las personas que tiene un cariño por los animales ayudan a esos animalitos pero al no tener muchos recursos no pueden darles un hogar permanente, por ello se optó por crear PetMap donde cualquier usuario con disposición a ayudar podrá publicar la ubicación de esos animalito y otros usuarios que quiera y tengan la posibilidad de dar un hogar a una nueva mascota podrán adoptarlas.

* 1. **Desarrollo de las interfaces del Sistema web**

Se utilizó Marvel App para realizar los prototipos del Sistema Web.

**2.2.1**. **Prototipo del sistema**

**Inicio**

En la figura 1 se muestra la pantalla de inicio con los campos: barra de navegación, registrarse, iniciar sesión.

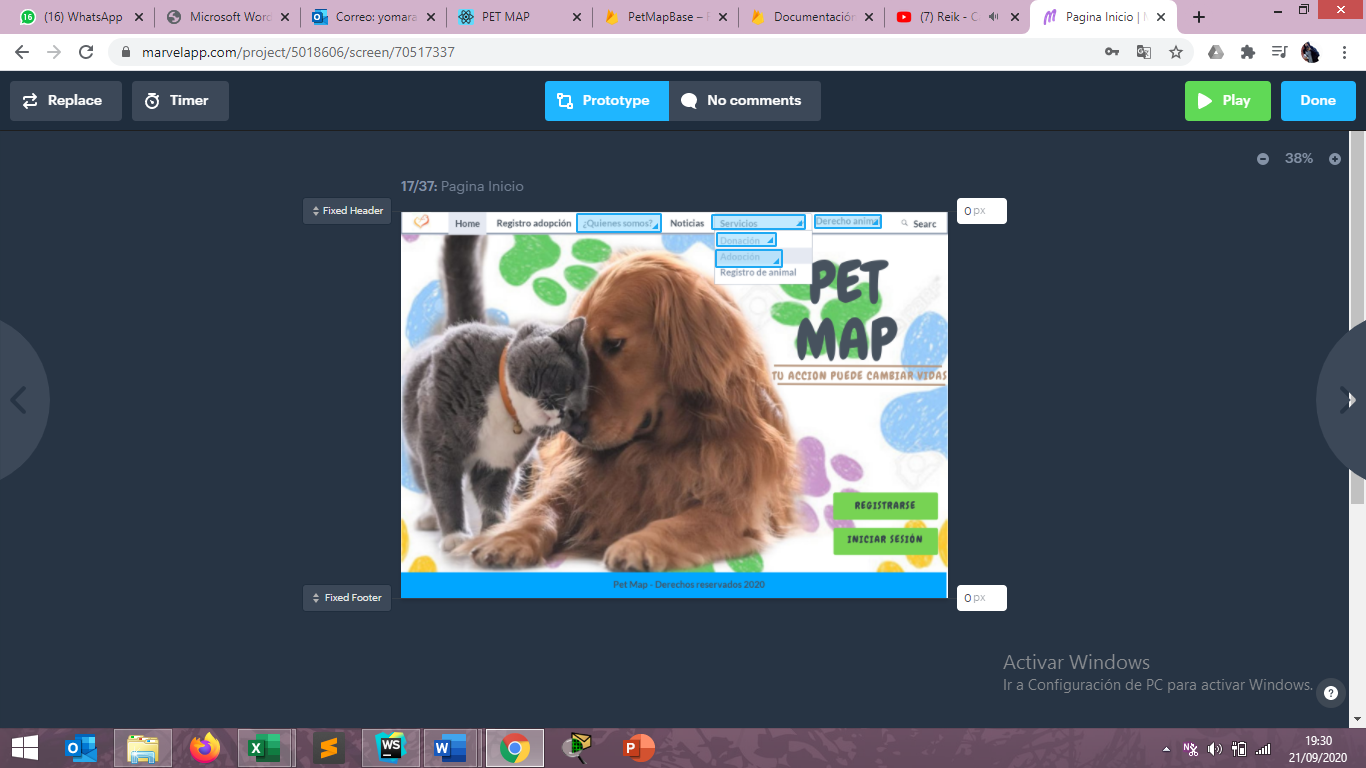


Figura 1 Diseñado del Inicio del administrador Elaborado por: Yomara Diaz

**Inicio de Sesión**

En la figura 2 muestra la pagina de inicio de sesión en caso de estar registrado en PET MAP, cuenta con: cajas de texto para mail y contraseña, olvido contraseña y botón de inicio de sesión.

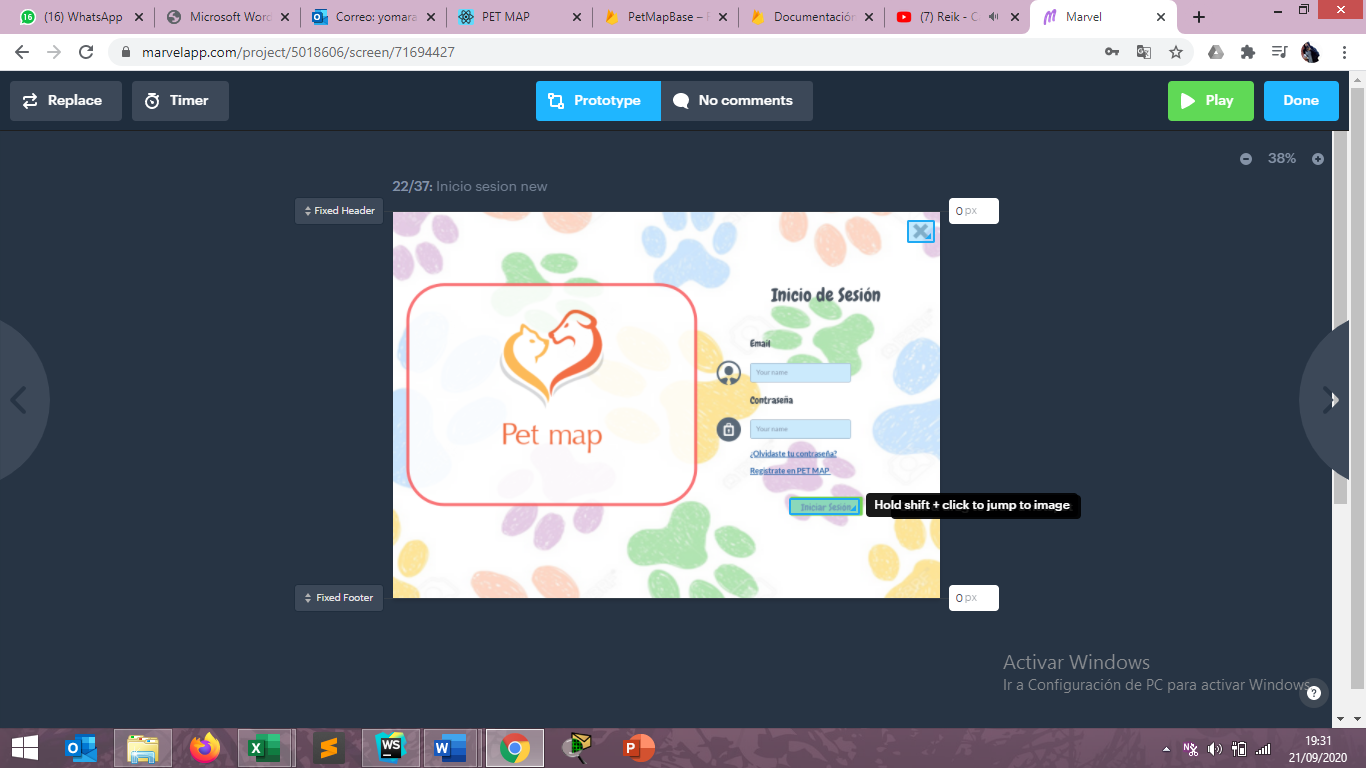


Figura 2 Diseñado del Inicio de Sesión del usuario Elaborado por: Elvis Agila

**Registro**

En la figura 3 se muestra la opción de registro para un nuevo usuario, cuenta con cajas de texto para el ingreso de los datos del nuevo usuario: nombres, apellidos, email y contraseña. También tiene el botón registro.

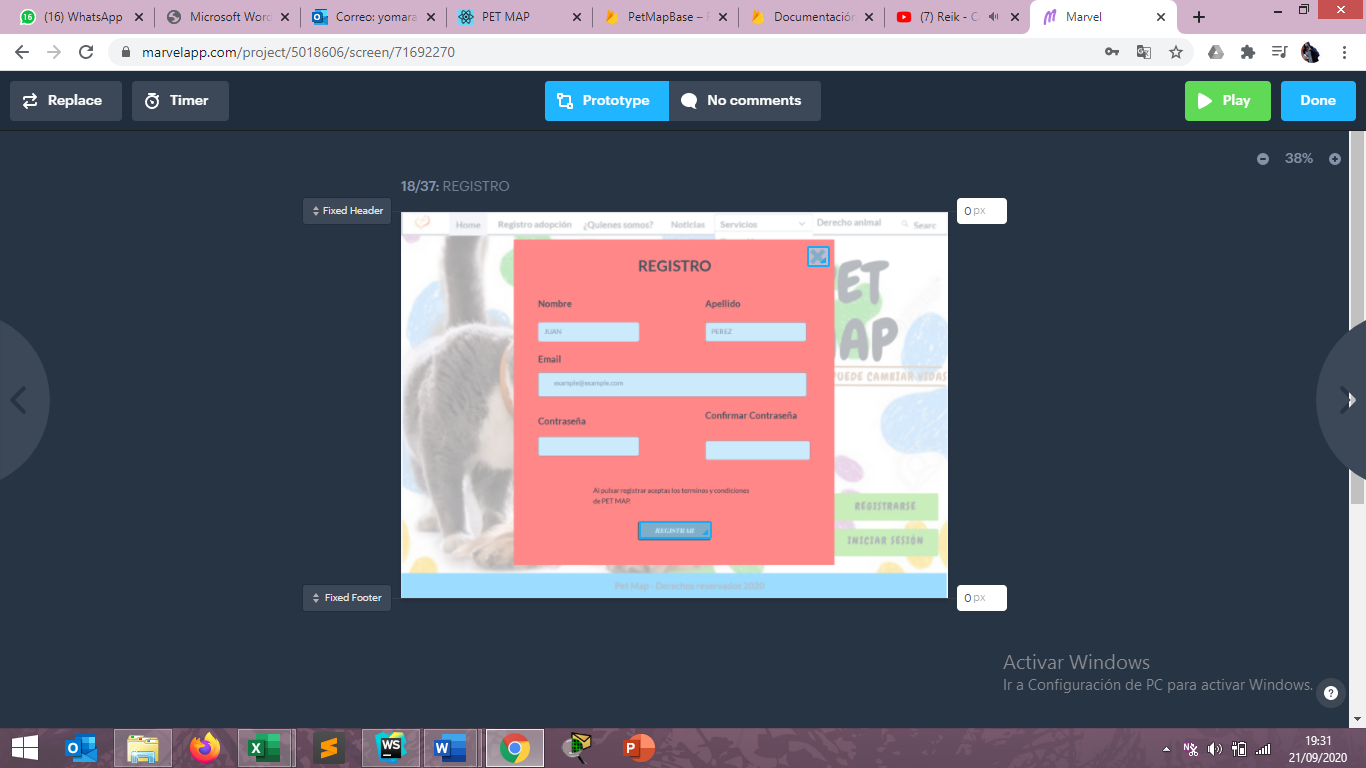


Figura 3 Diseñao de registro de usuario Elaborado por: Elvis Agila

**Página Principal**

En la figura 2 se observa las diferentes opciones que brinda la aplicación: donación, adopción, registro animal y barra de navegación.

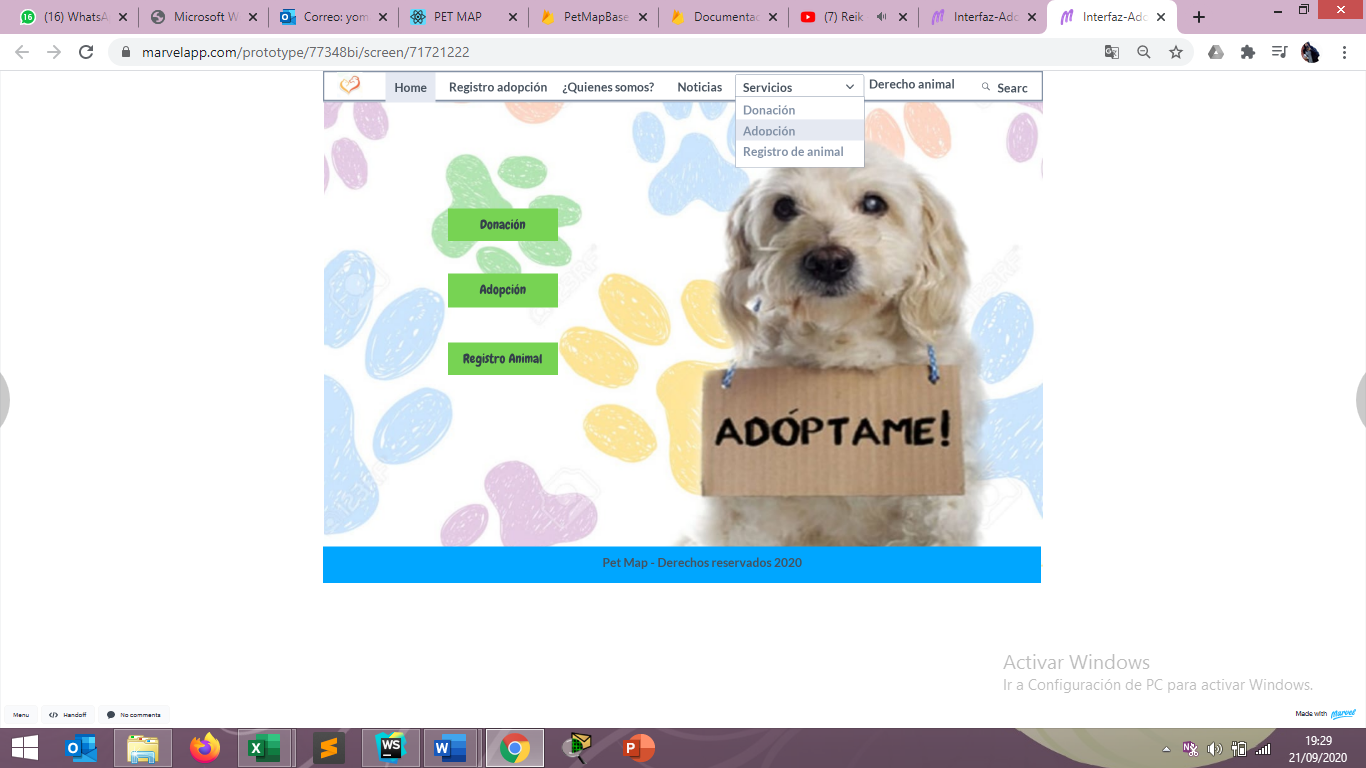


Figura 4 Diseñado página principal del administrador Elaborado por: Yomara Díaz

**Adopción**

En la figura 5 muestra la página de donaciones, la cual consta de 3 opciones: donaciones físicas, donaciones transferencia bancaria y donaciones débito bancario.

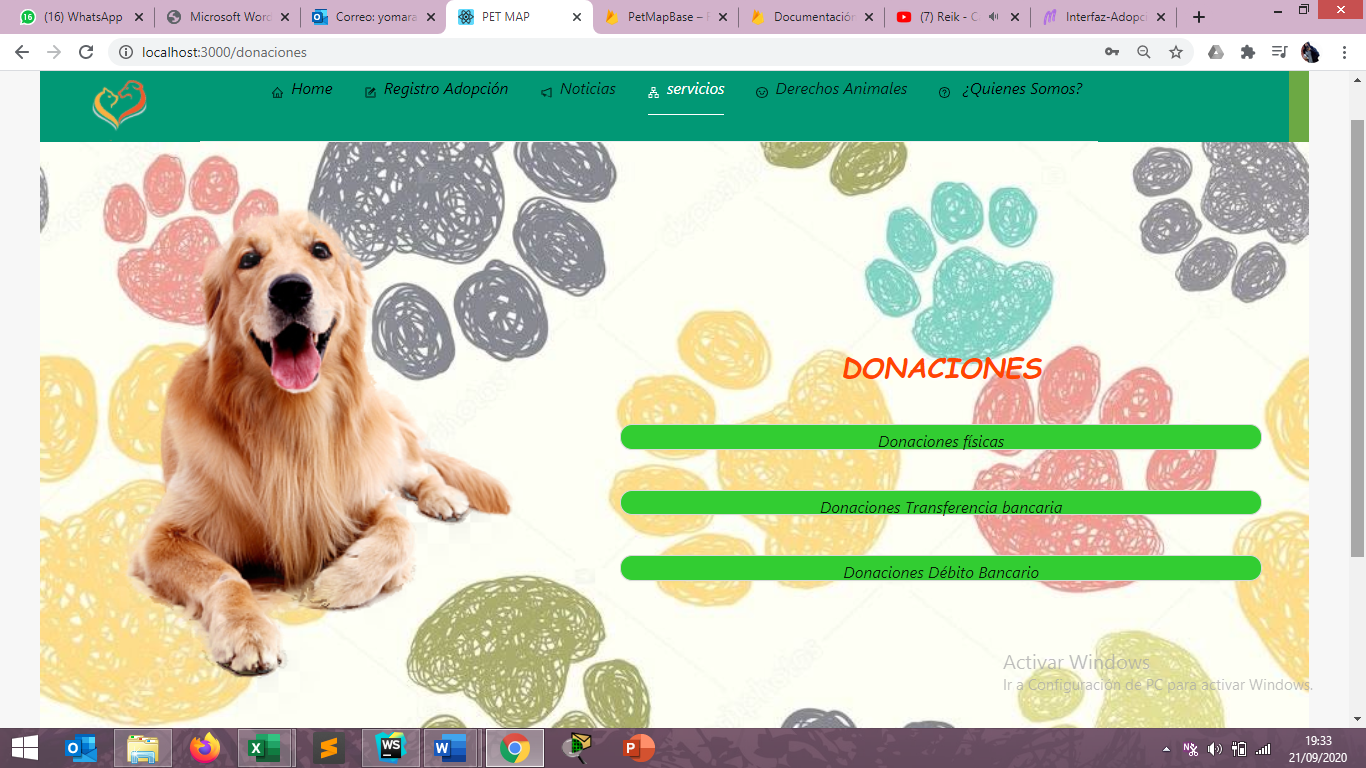


Figura 5 Diseñado donaciones del usuario Elaborado por: Yomara Díaz

**Registro adopción**

Figura 6 se observa cajas de texto para ingresar los datos de las personas que desean adoptar.

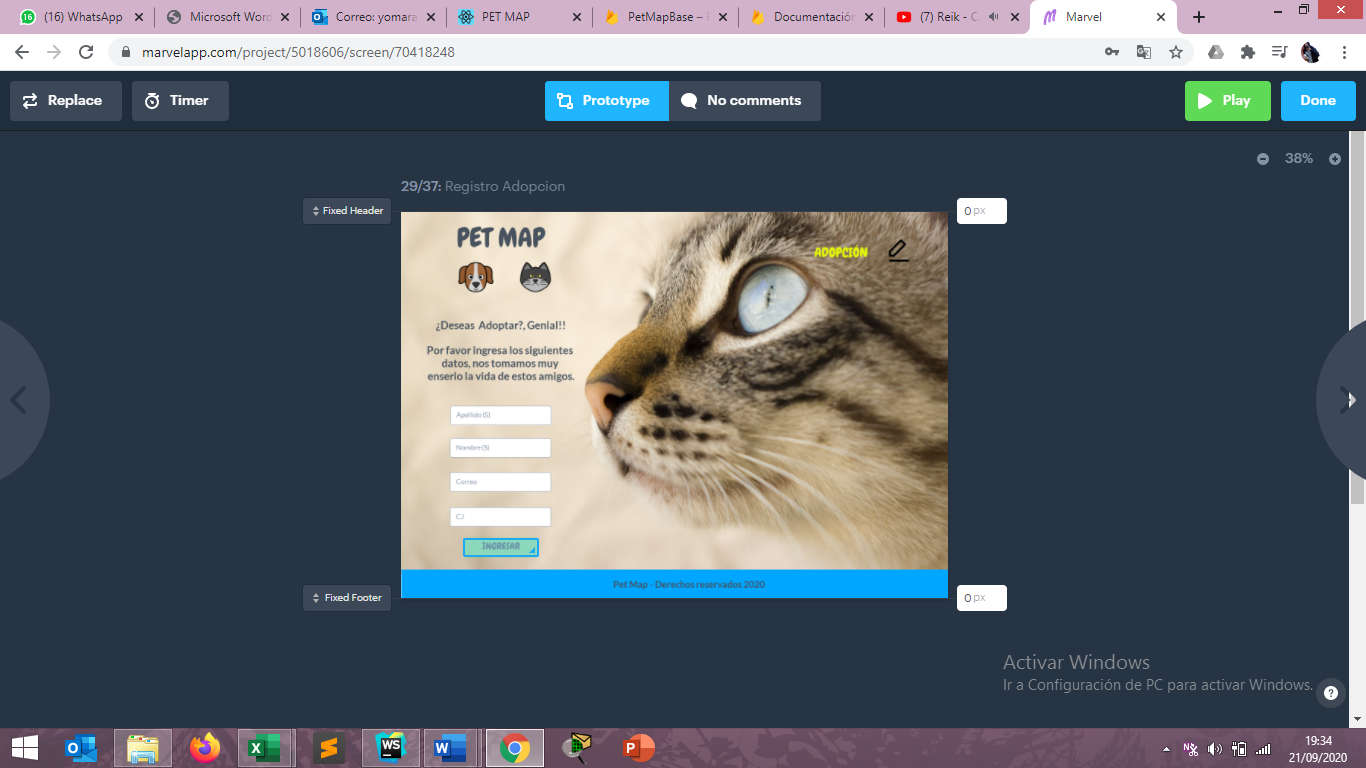


Figura 6 Diseñado Registro Adopción de animal Elaborado por: Guillermo Rivera

**Formulario de Adopción**

**Figura 7** muestra una serie de preguntas para la validación de la persona que desea adoptar el animal.



Figura 7 Diseñado Formulario de adopción de usuarios Elaborado por: Guillermo Rivera

**Animales en adopción**

Figura 8 muestra los animales para adoptar.

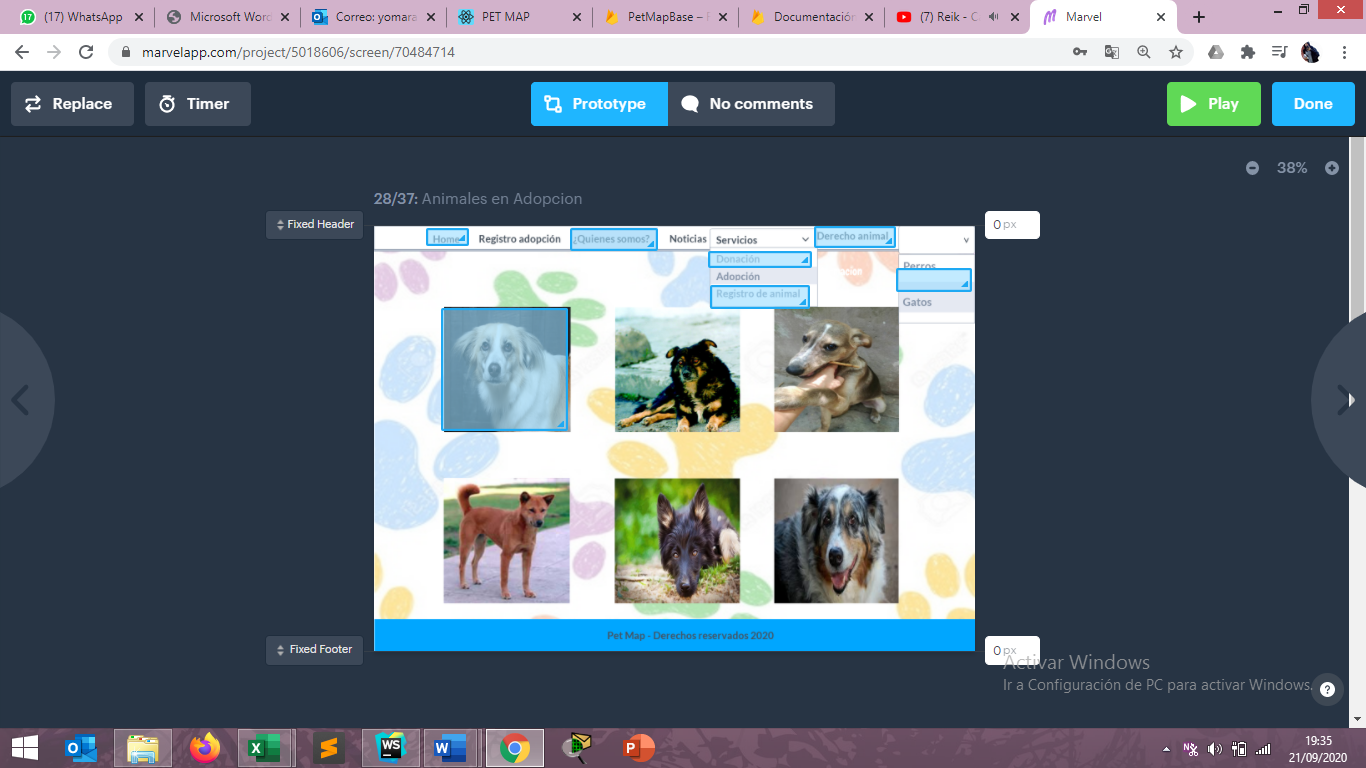


Figura 8 Diseñado de Animal en adopción Elaborado por: Carlos Montesdeoca

**Registro de animal**

Figura 9 muestra la pantalla en la cual los usuarios pueden registrar un animal para poder ser rescatado, deben registrar: ubicación del animal, descripción, foto (opcional). Tiene dos botones: registrar y cancelar.

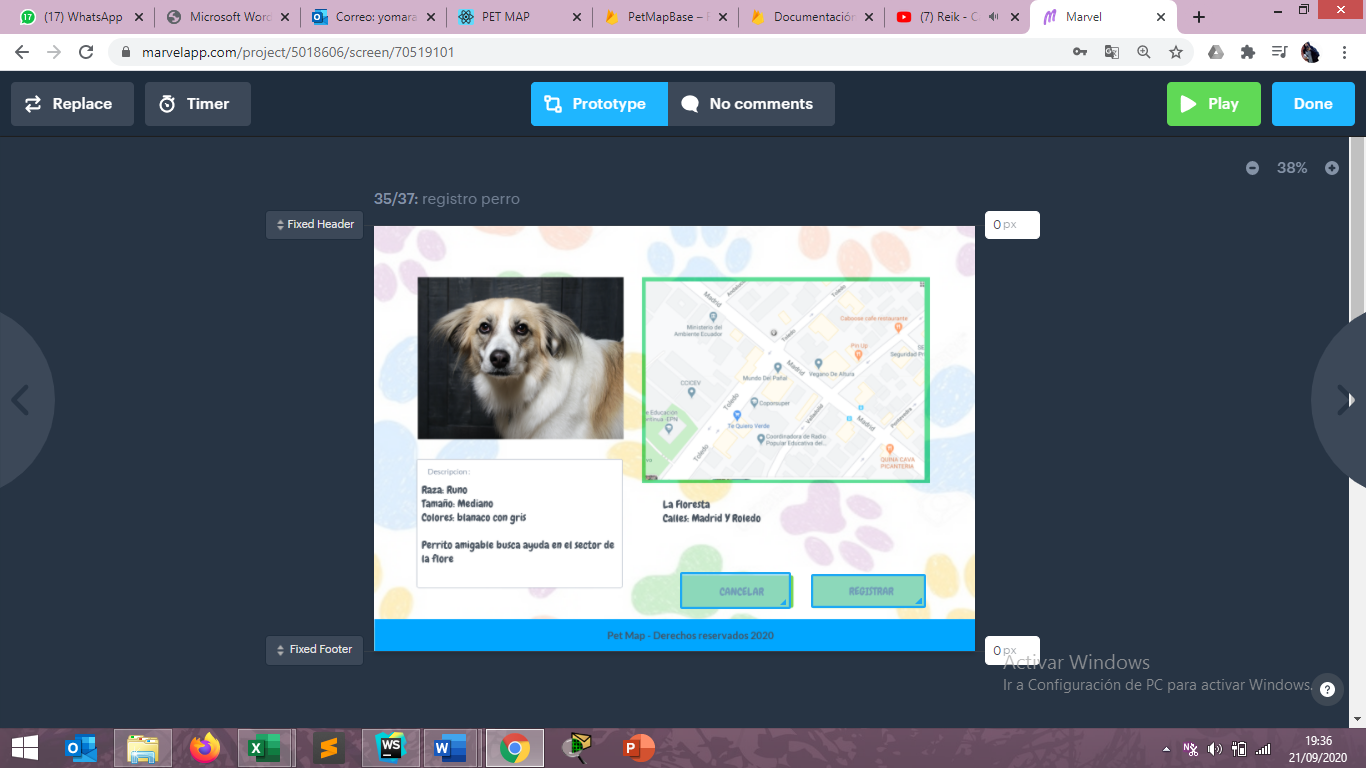


Figura 9 Diseñado Registro Animal Elaborado por: Carlos Montesdeoca

**Acerca de**

Figura 10 muestra datos de la página, misión y visión.

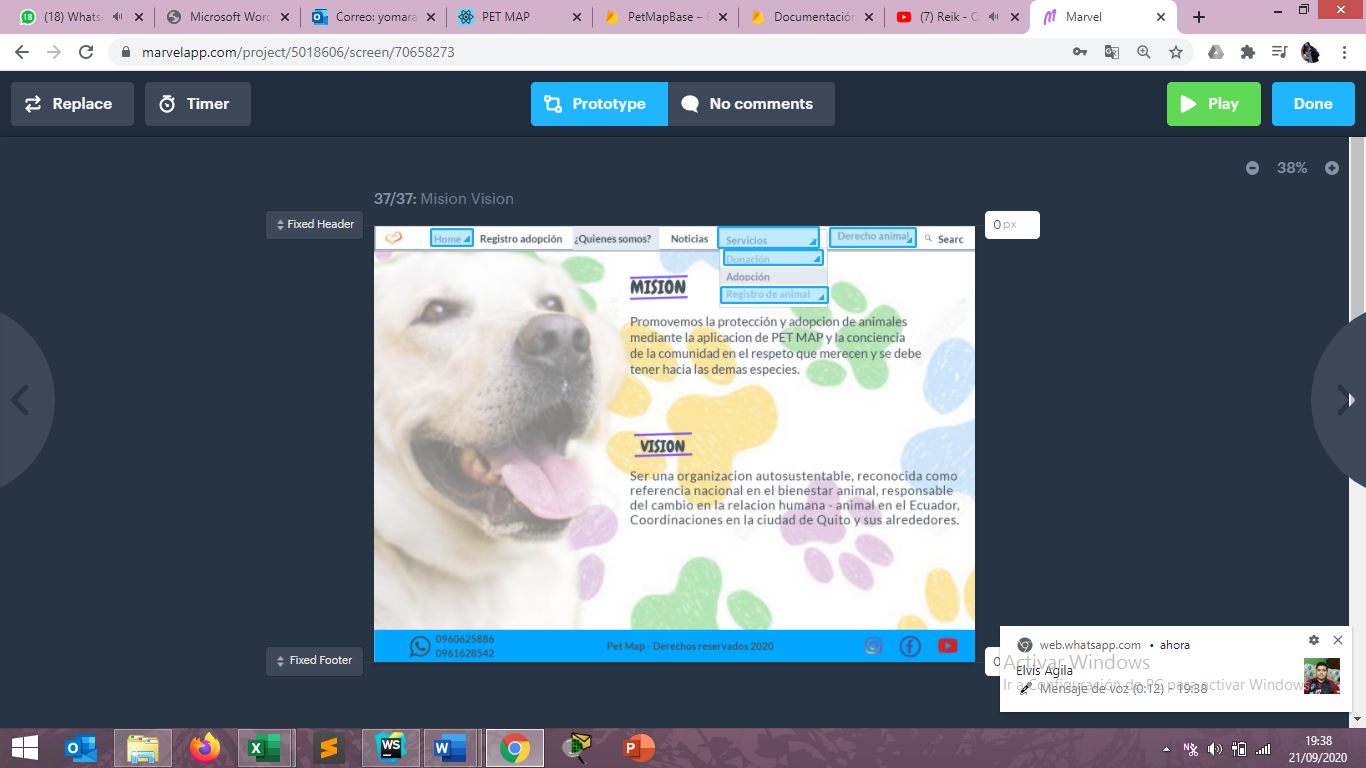


Figura 10 Diseñado acerca de para información Elaborado por: Yomara Díaz

**Derecho Animal**

Figura 11 muestra los derechos de los animales y consecuencias en caso de no cumplirlas

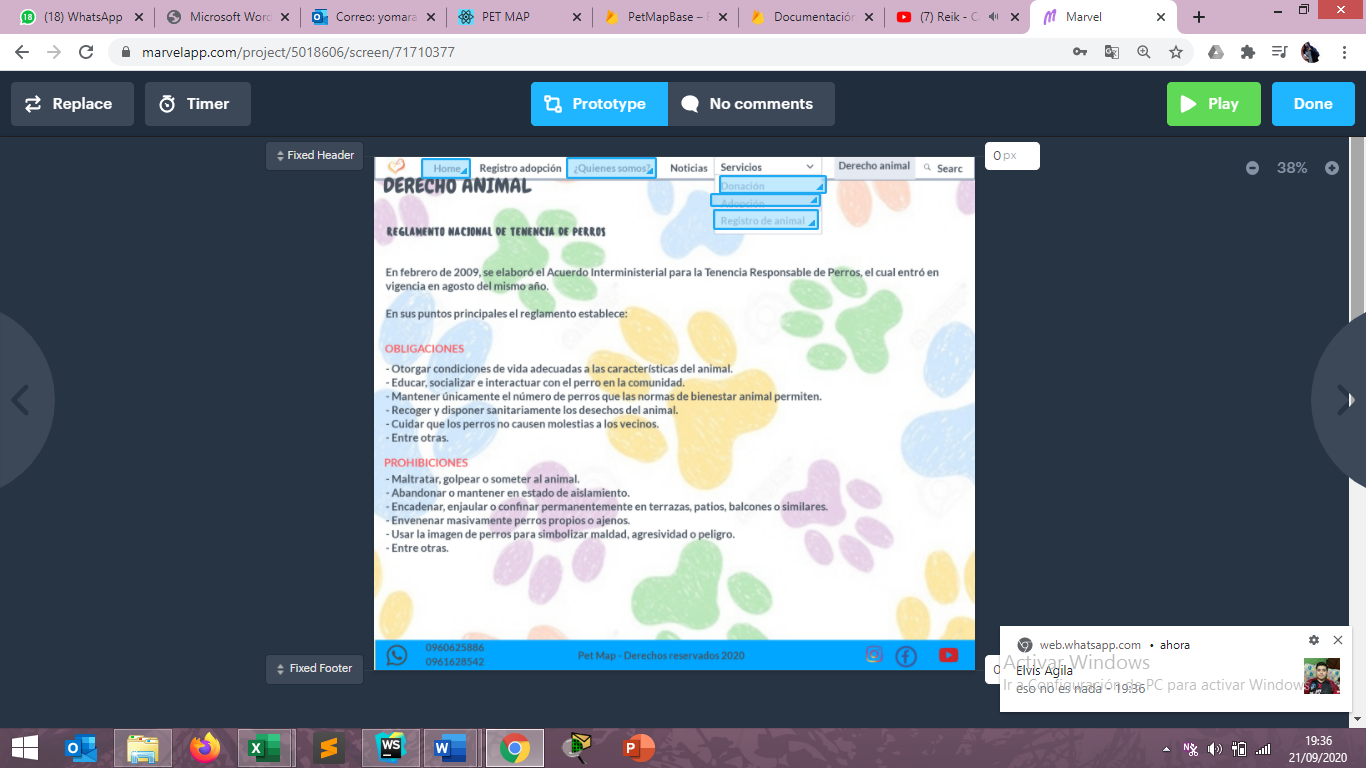


Figura 11 Diseñado Derecho Animal para protección del animal Elaborado por: Yomara Díaz

**2.3. Diseño de las arquitecturas del sistema web**

En la figura 12 se muestran tres secciones que corresponden a usuarios, servicios y base de datos.

En la primera sección se muestra a los usuarios: Usuario registrado, usuario no registrado y administrador, los cuales van a interactuar con el sistema Web de acuerdo a los roles y permisos otorgados.

En la segunda sección se muestra los servicios que realiza el sistema Web, por ejemplo: Registro de animal, Adoptar animal, Ver listado de animales, estas actividades las puede realizar el usuario que este registrado en la página.

En la tercera y última sección se muestra la base de Datos (FireBase) en la cual se va almacenando los diferentes datos como ejemplo: cuando una persona va a realizar el registro de un animal o cuando una persona va a registrarse.

En la parte inferior de muestra la plataforma a utilizar Web, además, se menciona las tecnologías que se usa en la sección de Servicios y Base de datos.

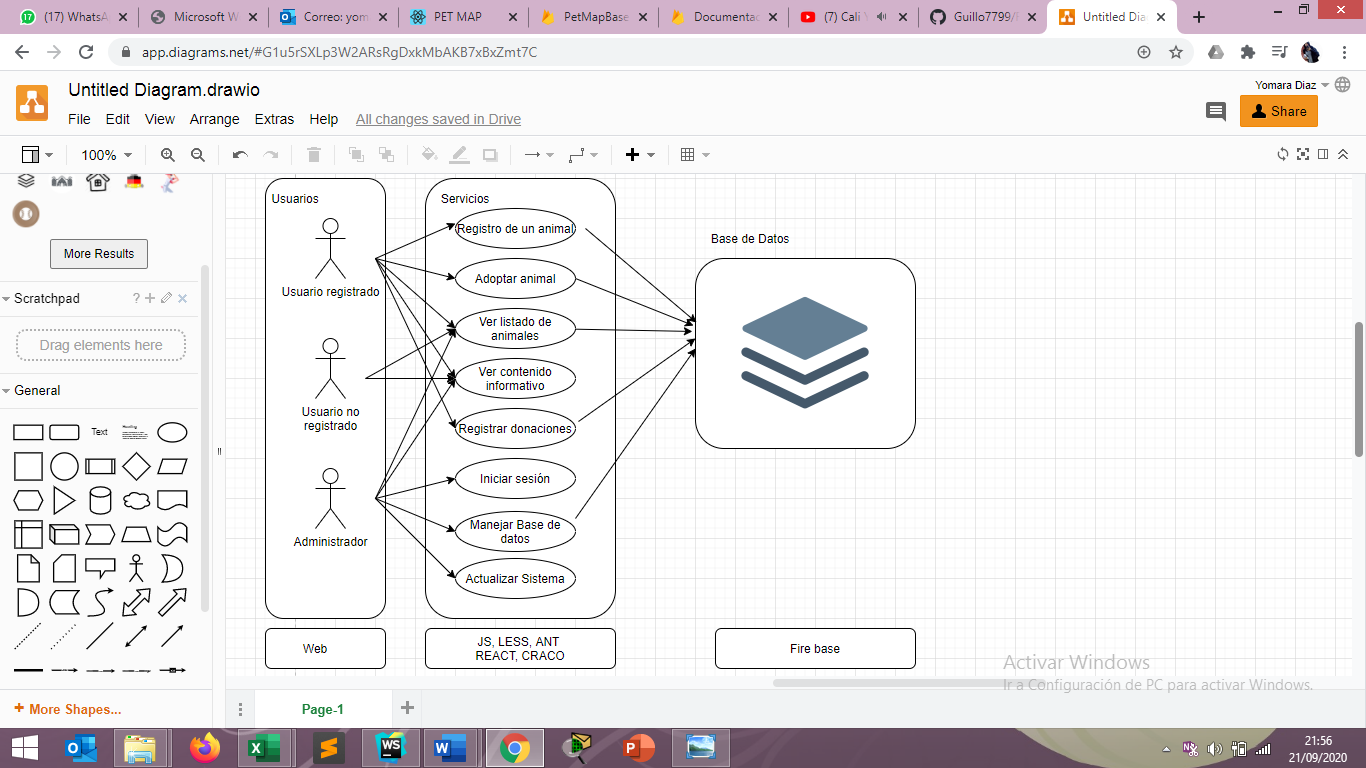


Figura 12 Caso de uso y base de datos Elaborado por: Yomara Diaz

En la figura 13 se muestra El diagrama de Infraestructura del Sistema Web con el fin de comprender el diseño y su implementación.

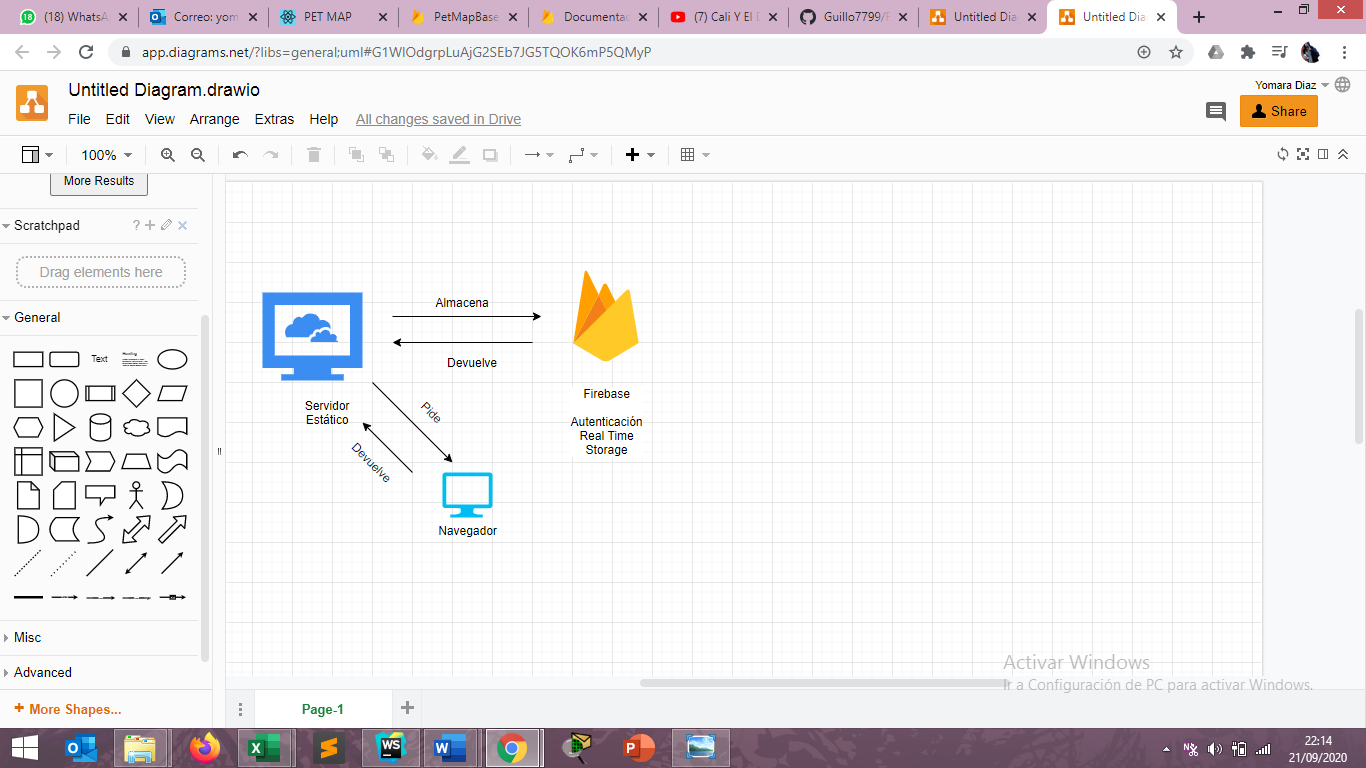


Figura 13 Diagrama de Infraestructura Elaborado por: Yomara Díaz

**2.4. Herramientas para la implementación**

**2.4.1. Herramientas para la base de datos**

**Firebase RealTime Database**

Es una plataforma para el desarrollo en la nube que provee una API para almacenar y sincronizar datos en tiempo real con un formato JSON, los cuales se mantienen disponibles cuando la aplicación no tiene conexión, Utiliza la infraestructura de Google y tiene beneficios como escalabilidad, estadística, autenticación, entre otros.

**2.4.2. Herramientas para el desarrollo del sistema Web**

**NPM**

Node Package Manager (NPM) es un gestor de paquetes, facilita tener cualquier librería disponible con solo implementar una línea de código, npm ayuda a administrar módulos, distribuir paquetes y agregar dependencias de manera sencilla. (Benites, 2015)

**JS**

Lenguaje de programación o de secuencias de comandos que permiten implementar funciones complejas en la página web, muestra información estática para ser vista. (Chris, 2017)

**ANT**

Kit de interfaz de usuario que contiene más de 2 mil componentes de interfaz artesanales para la aplicación web. Botones, entradas, casillas de verificación, tablas, información sobre herramientas, iconos y otros elementos de interfaz de uso común son parte de ANT. (UI, 2020)

**LESS**

Se configura especificando parámetros de invocación, además, es una tecnología que se emplea para la creación y diseño de paginas web, sirve para poner color al contenido, enlaces, entre otros. (Wikipedia, 2015)

**CRACO**

Create React App Configuration Overide es una capa de configuración fácil y comprensible para créate-react-app.

**REACT ROUTE**

Es una librería para gestional rutas en aplicaciones que utilicen ReactJS. Está inspirada en el sistema de enrutado de Ember. Js y s forma de manejar las rutas. (Hernandez, 2014)

**3 RESULTADOS**

**3.1 Requerimientos Funcionales**

En el momento en el que el usuario inicia sesión o se registra en nuestra aplicación web puede acceder a todos los servicios que está brinda, al momento en el que se registra por primera vez, esta información es almacenada en la base de datos.

Si el usuario desea registrar a un animal de la calle, puede acceder al apartado de “Registro de animal”, de esta manera se le pedirá al usuario que ingrese ciertos datos acerca del estado y dirección en la que encontró al animalito.

La aplicación cuenta con una interfaz dinámica y que le permite al usuario conocer un poco más acerca de los derechos de los animales, también acerca de las personas que se encargaron de desarrollo, además de poder realizar una visualización de los perros/gatos que ya se han registrado en el sistema.

Nuestro sistema cuenta con un apartado de Adopción, en el cual el usuario puede optar por alguno de nuestros amigos de 4 patas que se encuentran en el sistema, para lograr esto es necesario que el usuario llene el formulario de adopción para poder verificar si es que es apto o no, en caso de ser apto se le mostrará una ficha de adopción, en la cual constarán ciertos datos tanto del usuario como del perro o gato que está adoptando.

En caso de que se quiere hacer una donación de algún tipo como humanitaria o monetaria, el sistema cuenta con una interfaz de donaciones en la cual, dependiendo del tipo será registrada en la base de datos.

**3.2 Usuarios del Sistema Web**

A parte del grupo de desarrollo y los clientes, el proyecto está dirigido para individuos que tengan interés en el uso del sistema, este no tiene restricciones de edad en sí, está desarrollado con la intensión de poder ser utilizado por personas que tengan conocimientos básicos acerca de dispositivos inteligentes y conexión a internet, para ser mas exacto el sistema es para personas mayores a 7 años.

Los usuarios que interactúen con el sistema pueden conocer los servicios que brinda este con solo ingresar los datos necesarios para el funcionamiento de dichos servicios que se pueden utilizar a medida que vaya conociendo la interfaz del sistema.

**3.3 Restricciones de Desarrollo**

Para aprobar una adopción es necesario que el usuario ingrese sus datos, esto para validar si es que se encuentra en la base de datos, de esta manera se puede acceder al apartado de formulario.

De la misma manera ciertos servicios como la donación, es necesario que el usuario se registre o inicie sesión, para poder almacenar sus datos en el apartado de donaciones.

Por el momento la aplicación no cuenta está disponible como plataforma en móviles, pero si se encuentra disponible para poder visitarla desde el navegador de su dispositivo.

Un requerimiento para mejor manejo en la base de datos es en el formulario de adopción el cual solo cuenta con las opciones “si” y” no” para contestar a las preguntas, esto para agilizar el proceso.

En el apartado de donaciones, se permite realizar débitos mensuales por medio de la plataforma Paypal y si se trata de transferencias bancarias, está incorporado para 4 bancos, los enlaces a dichos bancos se encuentran en la interfaz de donaciones.

**3.4 Consideraciones**

El proyecto pretende dar una ayuda a la parte de rescate de animales de la calle, de una manera tecnológica, amigable y fácil de usar para el usuario, está desarrollado con la intensión de poder ser usado por un público en general que tenga conocimientos básicos sobre usos de herramientas web y dispositivos inteligentes. El equipo de desarrollo consiente de que no es una ayuda total y que no es una solución final, es autónomo en su desarrollo y busca ser aprobado de manera colectiva, para en un futuro poder optar con este proyecto a llevarlo a un nivel superior y de esta manera poder brindar una ayuda mas completa y de paso llevar a sus desarrolladores a encontrarse entre sí dentro del mundo de la programación.

El personal del proyecto, desarrolladores y clientes como pilares y sustento de vida del proyecto son los cuales han logrado llevar a cabo la elaboración de este, concluyendo que el proceso de desarrollo es continuo e iterativo en sus partes. Como instrumentos de elaboración, se tienen dispositivos y materiales que se encuentran disponibles para la comunidad.

**3.5 Conformación del equipo de trabajo**

El equipo de trabajo fue conformado por cuatro estudiantes de Desarrollo de Software de la Escuela de Formación de Tecnólogos (ESFOT) de la Escuela Politécnica Nacional (EPN) que actualmente se encuentran cursando la mitad de carrera.

El equipo fue dividido de la siguiente manera: Yomara Díaz, como encargada de recepción de requisitos de usuario, Carlos Montesdeoca como administrador de bases de datos, Elvis Agila como coordinador de proyecto y Guillermo Rivera como Scrum Master.

Los cuatro integrantes se encargaron del desarrollo del sistema con la finalidad de afianzar conocimientos, mejorar el trabajo en equipo, la organización estuvo realizada por medio de reuniones continuas por lo integrantes. El proyecto utiliza la metodología de desarrollo Prototipado, puesto que se realizaron mockups de las interfaces de usuario que el sistema contiene, para poder hacer una revisión y tener en cuenta la aceptación del Cliente, para fines del proyecto se definen como Clientes a los Ingenieros Edwin Salvador y Juan Pablo Saldumbide, los cuales han llevado una revisión continua del proyecto, para verificar que se cumplan con los sprints definidos con anterioridad.

El proyecto y su organización fue definida para cuatro sprints, cada uno cuenta con las tareas a realizarse y el tiempo que va a durar cada sprint para entrega del proyecto.

**3.7 Pruebas en el Sistema Web**

Las pruebas del sistema son aquellas acciones que permiten verificar la integración de un sistema en un medio, verificando de esta manera el correcto funcionamiento de cada una de sus interfaces.

La comunicación entre subsistemas y las interfaces tiene que ser el correcto ya que de esto depende la comunicación del usuario y la funcionalidad del sistema.

Estas pruebas permiten probar el sistema en su totalidad, permitiendo de esta manera detectar problemas previos a su implementación y entrega.

Existen diferentes tipos de pruebas que se pueden implementar en un sistema, para caso de estudio y de uso se procederá a detallar tres, las cuales fueron las necesarias para poder comprobar el funcionamiento del sistema.

**3.7.1 Pruebas de Caja negra**

Las pruebas de caja negra son aquellas que permiten comprobar la funcionalidad de un sistema sin tomar en cuenta la estructura interna del mismo, en este tipo de pruebas el desarrollador se enfoca en las entradas y salidas que puede tener el sistema.

Para poder entender cuales son las entradas y salidas del sistema hay que tener en cuenta los requerimientos del sistema, este tipo de datos dependerá de la funcionalidad que realice el sistema.

Dentro de las pruebas de caja negra existen varias técnicas, las que se han utilizado en el proyecto han sido las técnicas de transición entre estados y las pruebas de casos de uso (Fig. 14-16), este tipo de técnicas permiten conocer las vías principales al momento de dar uso al sistema.

Dentro de la técnica de transición entre estados se verificaron los cambios en las interfaces al momento de seleccionar algún tipo de acción o tarea, a demás de poder verificar y comprobar la integridad de datos al momento de la entrada y salida de estos.

Con las pruebas de casos de uso se verificó la aceptación por parte del cliente, para uso del proyecto se define a los Ingenieros a cargo de las materias y revisiones de los proyectos como clientes, en caso de que se logre aceptación por parte de los clientes al momento de entregar los diagramas de casos de uso, se procede a desarrollar las pruebas de dichos casos, de esta manera de receptan post-condiciones que son los resultados que se pueden observar directamente del sistema.

**3.7.2 Pruebas de Compatibilidad**

Las pruebas de compatibilidad son aquellas que permiten verificar la eficiencia de un sistema en diferentes dispositivos, este proceso es vital en todo tipo de proyecto de desarrollo de software puesto que permite la generalidad del producto en el mercado.

En el proyecto se desarrollaron las pruebas al inicio, primero con dispositivos de gama baja como ordenadores de segunda y tercera generación con procesadores Intel Core i3 e i5 para verificar que el sistema cumpla con el objetivo de brindar los servicios que el proyecto ha establecido con anticipación, posterior a dichas pruebas se desarrollaron aquellas que son destinadas al software, verificando el funcionamiento del proyecto en sistemas operativos Linux, Mac y Windows permitiendo de esta manera comprobar la compatibilidad del proyecto en los diferentes sistemas operativos.

Con este tipo de pruebas se puede validar en qué tipo de dispositivos se puede implementar el proyecto y en cuales no, por lo tanto, este tipo de pruebas son un pilar fundamental en el proceso de desarrollo, misma razón por la cual han sido implementadas en el proyecto.

**3.7.3 Pruebas de Interfaz de Usuario**

Las pruebas de Interfaz de Usuario son aquellas que están ligadas al levantamiento de requerimientos y al objetivo de entregar un sistema de calidad. Este tipo de pruebas permiten conocer si la app o el producto a entregarse cumplen con los requerimientos del cliente, funcionando de manera correcto y de esta manera aumente sus probabilidades de la aceptación de los usuarios en general una vez que se implemente.

Para poder implementar estas pruebas es necesario identificar a un actor ajeno al proceso de desarrollo, pero que tenga conocimientos sobre lo que se requiere del sistema, de esta manera este “actor” puede manipular el sistema de manera en que lo podría realizar un usuario cualquiera.

El problema con estas pruebas es que conllevan tiempo y costos, a parte de que puede provocar más errores si el usuario hace mal uso del sistema. y para un proceso de desarrollo es fundamental enfocar los tiempos en el sistema, cumpliendo con los sprints ya definidos y con los objetivos de cada una de las pruebas de casos de uso. Para manejar de mejor manera este tipo de pruebas es una buena técnica el realizar un manual, dicho manual ayuda al usuario a ingresar y utilizar el sistema de manera correcta de forma automática, permitiendo a las pruebas de interfaz de usuario funcionar rápido y de forma confiable.

Para el proyecto se desarrollaron este tipo de pruebas por usuarios ajenos a la carrera, en este caso familiares de los desarrolladores, los cuales tras una guía y una inducción acerca del funcionamiento del sistema, realizaron uso de este.

**Casos de uso**

**DONACIONES**

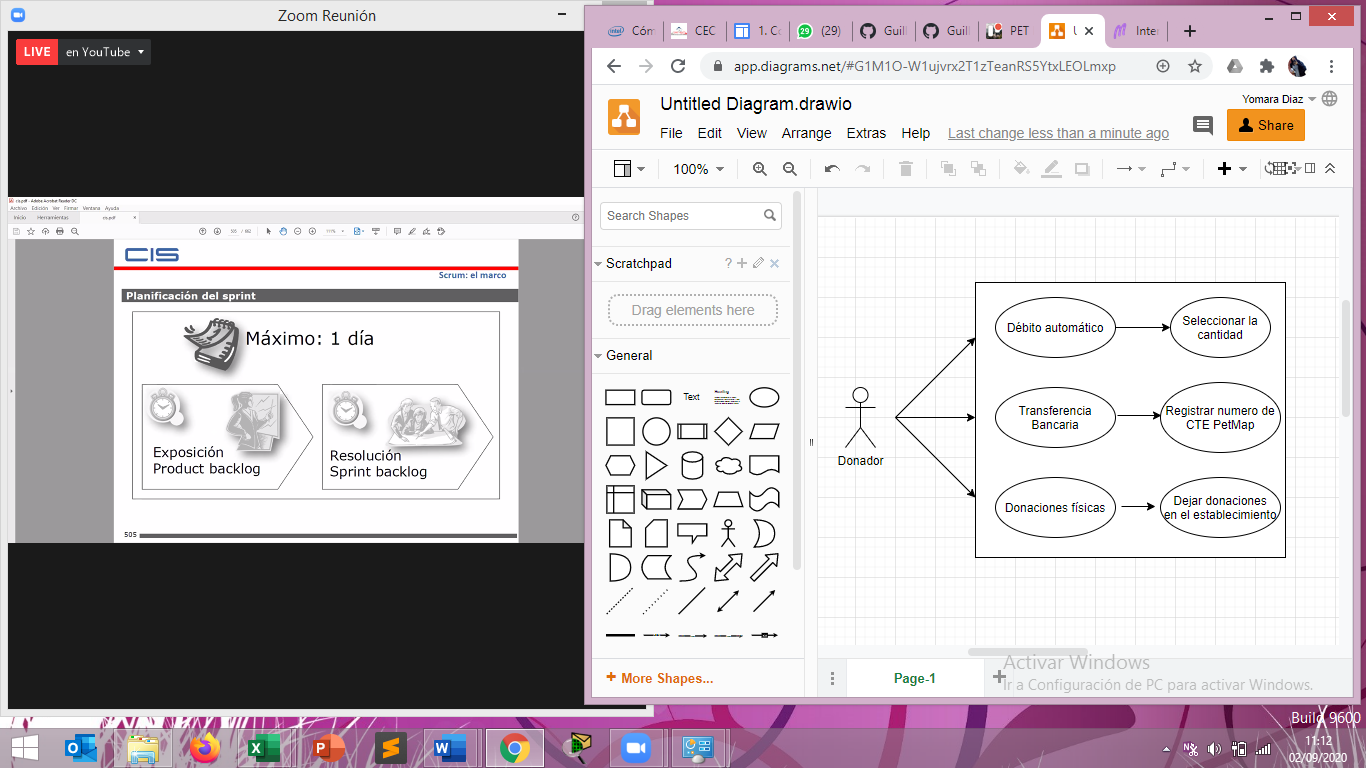


Figura 14 Diagrama de casos de uso de Donaciones

**INICIO DE SESION**

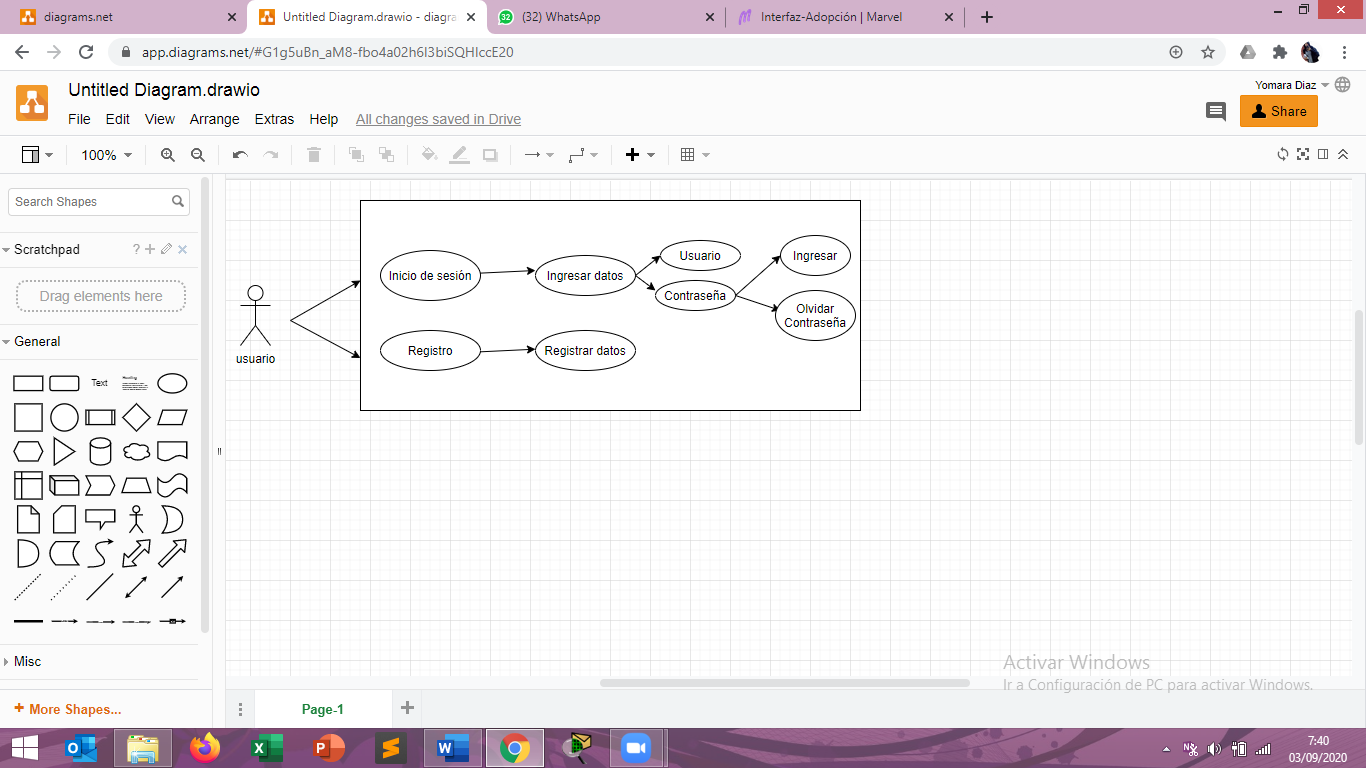


Figura 15 Diagrama de casos de uso de Inicio de sesión

**ADOPCIÓN**

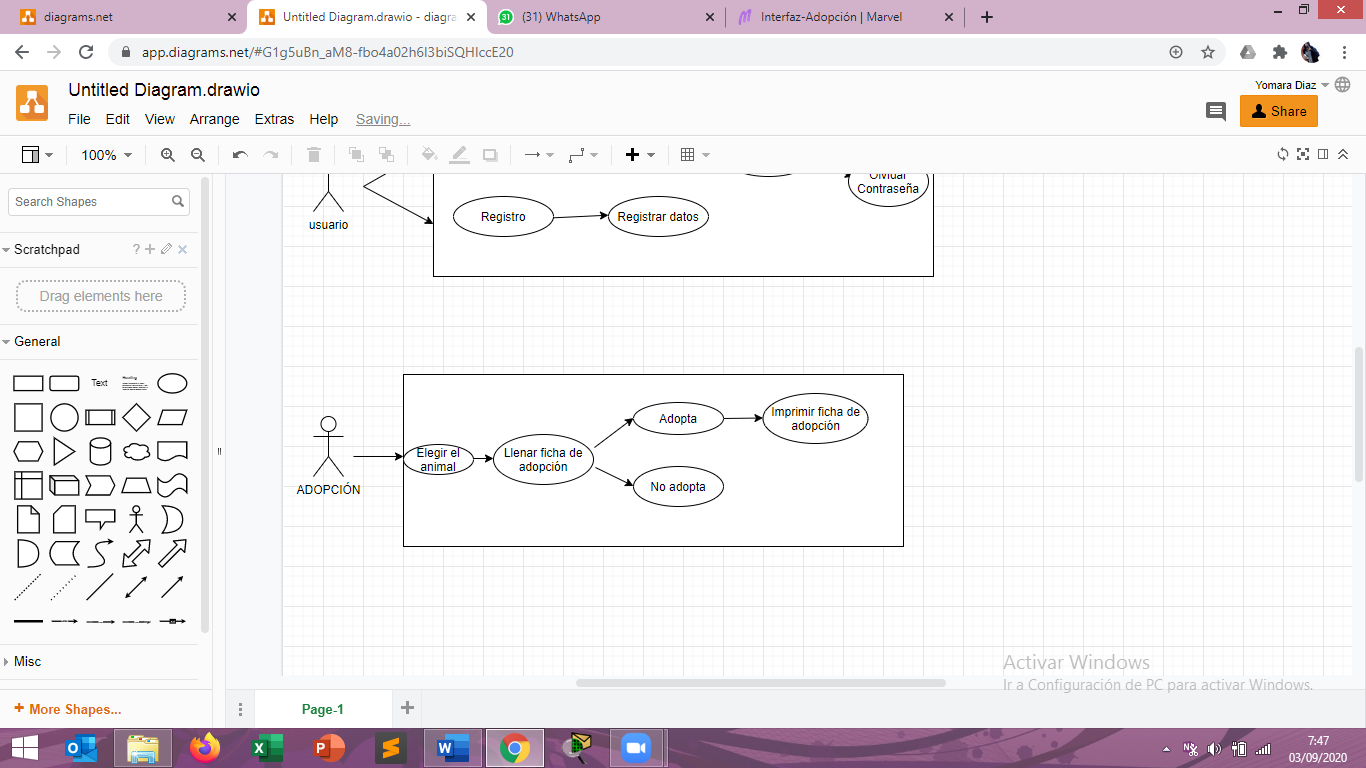


Figura 16 Diagramas de caso de uso de Adopción

**HISTORIAS DE USUARIO**

**Roles**

**Scrum Master:** Guillermo Rivera

**Desarrolladores:** Elvis Agila

Yomara Díaz

Carlos Montesdeoca

Guillermo Rivera

**Administrador:** Elvis Agila

**Primer Sprint**

|  |  |  |
| --- | --- | --- |
| **Historia de usuarios** | | |
| **ID:** 1 | **Usuario:** Desarrollador | |
| **Nombre de historia:** creación del grupo de redes sociales para la comunicación continua del grupo (Whatsapp, Zoom, Facebook). | | |
| **Prioridad de negocio:** alta | | **Riesgo de desarrollo:** bajo |
| **Responsable:** Yomara Diaz | | **Iteración asignada:** 8 |
| **Descripción:**  Se crean las diferentes redes sociales (WhatsApp, Zoom, Facebook), para la comunicación constante del equipo de desarrolladores. | | |
| **Observaciones:** | | |

|  |  |  |
| --- | --- | --- |
| **Historia de usuarios** | | |
| **ID:** 2 | **Usuario:** Desarrolladores | |
| **Nombre de historia:** Diseño de Casos de uso | | |
| **Prioridad de negocio:** alta | | **Riesgo de desarrollo:** media |
| **Responsable:** Guillermo Rivera, Yomara Diaz, Elvis Agila, Carlos Montesdeoca | | **Iteración asignada:** 1 |
| **Descripción:**  Los diagramas de casos de uso sirven para especificar la comunicación y el comportamiento de un sistema mediante su interacción con los usuarios y/u otros sistemas. O lo que es igual, un diagrama que muestra la relación entre los actores y los casos de uso en un sistema. | | |
| **Observaciones:**  Se realizara revisiones y correcciones en caso de ser requerido. | | |

****

****

****

**Segundo Sprint**

|  |  |  |
| --- | --- | --- |
| **Historia de usuarios** | | |
| **ID: 1** | **Usuario:** Desarrollador | |
| **Nombre de historia: C**reación de página principal y enlaces con las otras interfaces | | |
| **Prioridad de negocio:** Alta | | **Riesgo de desarrollo:** Media |
| **Responsable:** Integrantes | | **Iteración asignada:** 0 |
| **Descripción:**  Se llegó a un consenso de los elementos que necesitan ser incluidos en la página principal.   * Botones de inicio de sesión. * Logo de presentación de nuestro equipo. * Descripción de nuestra misión y visión. * Sección de derechos de los animales. | | |

|  |  |  |
| --- | --- | --- |
| **Historia de usuarios** | | |
| **ID: 2** | **Usuario:** Desarrollador | |
| **Nombre de historia:** Definición de Base de Datos a utilizarse. | | |
| **Prioridad de negocio:** Media | | **Riesgo de desarrollo:** Alta |
| **Responsable:** Integrantes | | **Iteración asignada: 0** |
| **Descripción:**  Las presentaciones de los datos de nuestra página deben ser simples y pero específicos, entre los datos sugeridos para el ingreso de un nuevo animalito se necesitara:   * Raza del animal * Especie * Ubicación * Descripciones * Atenciones Requeridas (opcional)   Para que un animalito pueda ser adoptad los usuarios registrados deberán verificar sus identidades, para lo cual el equipo desarrollador concluyo que se necesita los datos de usuario:   * Nombre Completo * CI * Correo electrónico * Teléfono (opcional) * Pregunta: ¿Por qué quiere adoptar? | | |

|  |  |  |
| --- | --- | --- |
| **Historia de usuarios** | | |
| **ID: 3** | **Usuario:** Administrador | |
| **Nombre de historia:** Reunión con los integrantes para revisión final de los diagramas de casos de uso y revisión de avances del proyecto | | |
| **Prioridad de negocio:** Alta | | **Riesgo de desarrollo:** Media |
| **Responsable:** Grupo de Desarrollo | | **Iteración asignada:** 0 |
| **Descripción:**  Para poder mantener un buen manejo de los tiempos establecidos para el desarrollo del sistema se realizan reuniones continuas de los desarrolladores para verificar los diagramas de casos de uso que permitirán tener una base acerca del funcionamiento del proyecto. | | |
| **Observaciones:**  Se pueden presentar problemas como la inasistencia de alguno de los integrantes de grupo de desarrollo por razones de tipo personal o profesional, es necesaria la intervención por parte del administrador en os grupos de comunicación definidos al inicio del proyecto. | | |

**Tercer sprint**

|  |  |  |
| --- | --- | --- |
| **Historia de usuarios** | | |
| **ID: 1** | **Usuario:** Administrador | |
| **Nombre de historia:** Creación e implementación de la base de datos | | |
| **Prioridad de negocio:** Alta | | **Riesgo de desarrollo:** alta |
| **Responsable:** Grupo de Desarrollo | | **Iteración asignada:** 0 |
| **Descripción:**  Se trabajo en una base de datos virtual (FireBase), para implementar el registro de los animales, usuarios, tipo de donaciones. | | |
| **Observaciones:** la base de datos se trabaja en equipo por la complejidad. | | |

|  |  |  |
| --- | --- | --- |
| **Historia de usuarios** | | |
| **ID: 2** | **Usuario:** Administrador | |
| **Nombre de historia:** Pruebas continuas en cada interfaz y corrección de fallos y errores en caso de que existan | | |
| **Prioridad de negocio:** Alta | | **Riesgo de desarrollo:** alta |
| **Responsable:** Grupo de Desarrollo | | **Iteración asignada:** 0 |
| **Descripción:**  Una vez desarrolladas las paginas, se realizará la vinculación entre ellas y se probara. | | |
| **Observaciones:** en caso de haber inconvenientes, se realiza las correcciones correspondientes. | | |

**Cuarto Sprint**

|  |  |  |
| --- | --- | --- |
| **Historia de usuarios** | | |
| **ID: 1** | **Usuario:** Administrador | |
| **Nombre de historia:** Pruebas finales en el sistema y asignación de pruebas para cada integrante del proyecto. | | |
| **Prioridad de negocio:** Alta | | **Riesgo de desarrollo:** alta |
| **Responsable:** Grupo de Desarrollo | | **Iteración asignada:** 0 |
| **Descripción:**   * Yomara Diaz * Pruebas de carga. * Elvis Agila * Prueba de base de datos. * Guillermo Rivera * Enlace de interfaces. * Carlos Montesdeoca * Usabilidad. | | |
| **Observaciones:**  Se realizo la segunda presentación del proyecto. | | |

|  |  |  |
| --- | --- | --- |
| **Historia de usuarios** | | |
| **ID: 2** | **Usuario:** Administrador | |
| **Nombre de historia:** En el caso de haber un fallo en error en el sistema realizar su respectiva corrección y documentación. | | |
| **Prioridad de negocio:** Alta | | **Riesgo de desarrollo:** alta |
| **Responsable:** Grupo de Desarrollo | | **Iteración asignada:** 0 |
| **Descripción:**  Cada desarrollador tiene Pruebas asignadas, encaso de haber fallos en dichas pruebas realizar la corrección y documentación de dicho fallo. | | |
| **Observaciones:** | | |

|  |  |  |
| --- | --- | --- |
| **Historia de usuarios** | | |
| **ID: 3** | **Usuario:** Administrador | |
| **Nombre de historia:** Presentación del proyecto funcional y su respectiva documentación. | | |
| **Prioridad de negocio:** Alta | | **Riesgo de desarrollo:** alta |
| **Responsable:** Grupo de Desarrollo | | **Iteración asignada:** 0 |
| **Descripción:**  Entrega del proyecto. | | |
| **Observaciones:** | | |

BIBLIOGRAFIA:

* GESTION DE PROYECTOS INFORMATICOS, Metodologia Scrum  
  Autor: Manuel Trigas Galego  
  Consultora: Ana Cristina Domingo Troncho.
* ¿Qué es un Sprint?  
  Autor: Abraham requena Mesa  
  Año: 19 de diciembre 2018.
* Pruebas del Sistema

Autor: Manuel Cillero

Año: 12 de marzo de 2017.

* Pruebas de Caja Negra y un enfoque práctico

Autor: Gustavo Terrera

Año: 26 de febrero de 2017.

* Costos de errores en los requisitos

Autores: Boehm-Papaccio

Año: 1988