

Proyecto PetChain

Blockchain para la gestión de macotas

Versión 1.0

Fecha 03/05/2019

Preparado para:

[Universidad Internacional de la Rioja](#)

Preparado por:

?

Índice

- 1 [Introducción](#)
 - 1.1 [Propósito del documento](#)
 - 1.2 [Alcance del documento](#)
 - 2 [Participantes en el proyecto](#)
 - 2.1 [Presentación de participantes](#)
 - 2.2 [Organizaciones involucradas](#)
 - 3 [Descripción del sistema](#)
 - 3.1 [Perspectiva del producto](#)
 - 3.2 [Funciones del producto](#)
 - 3.3 [Características del usuario](#)
 - 3.4 [Restricciones](#)
 - 3.5 [Descripción técnica del sistema](#)
 - 3.5.1 [Front-End](#)
 - 3.5.2 [Seguridad](#)
 - 3.5.3 [Back-End](#)
 - 3.5.4 [Entorno de ejecución](#)
 - 4 [Objetivos del sistema](#)
 - 5 [Catálogo de requisitos del sistema.](#)
 - 5.1 [Requisitos de información](#)
 - 5.2 [Requisitos funcionales](#)
 - 5.3 [Requisitos no funcionales](#)
 - 6 [Matriz de rastreabilidad](#)
 - 7 [Glosario de términos](#)
-

1 Introducción

PetChain es un proyecto destinado al sector veterinario que permite la gestión de los sistemas de identificación de mascotas. La aplicación se basa en un desarrollo web tradicional, pero con la característica fundamental de estar implementada en una Blockchain bajo una red Ethereum. PetChain gestiona de las identidades de las mascotas domésticas dando de alta, modificando los datos y

gestionando la baja de los datos de los animales, bajo la ejecución de Smartcontracts creados en lenguaje Solidity.

1.1 Propósito del documento

El presente documento pretende realizar una descripción de los requisitos informativos, funcionales y no funcionales de PetChain, con el fin de comprender y describir su funcionamiento, así como la enumeración de los participantes en el desarrollo, de cada una de sus partes y conseguir al final del mismo, una relación entre las partes y los creadores de las mismas.

Pretende ser una descripción de los requisitos informativos, funcionales, no funcionales del proyecto, con el objetivo de ofrecer a los lectores del documento del TFG una visión clara de la aplicación y las partes que lo componen.

1.2 Alcance del documento

El presente documento tiene como alcance especificar:

- La declaración de los participantes en el proyecto.
- Definir de los objetivos de la aplicación.
- Proveer una descripción funcional del sistema que permita la mejor comprensión del mismo al equipo de evaluadores del proyecto TFG.
- Especificar de los requisitos informativos, funcionales y no funcionales que deberá cumplir la aplicación
- Establecer los límites de funcionales de PetChain.

2 Participantes en el proyecto

2.1 Presentación de participantes

Participante	Miriam Martínez Muñoz
Organización	Universidad Internacional de la Rioja
Rol	Dirección de proyecto
Es desarrollador	Sí
Es cliente	No
Es usuario	No
Comentarios	Directora del proyecto de fin de grado.

Participante	Carlos Escario Bajo
Organización	Universidad Internacional de la Rioja
Rol	Analista-Desarrollador
Es desarrollador	Sí
Es cliente	No
Es usuario	No
Comentarios	Ninguno

Participante	Pedro Escario Bajo
Organización	Veterinaria García-Ochoa
Rol	Usuario
Es desarrollador	No
Es cliente	No
Es usuario	Sí
Comentarios	Trabajador en la Veterinaria García-Ochoa

Participante	Iciar García-Ochoa
Organización	Veterinaria García-Ochoa
Rol	Usuario
Es desarrollador	No
Es cliente	No
Es usuario	Sí
Comentarios	Propietaria de la Veterinaria García-Ochoa

2.2 Organizaciones involucradas

Organización	Universidad Internacional de la Rioja
Dirección	Av. de la Paz, 137, 26006 Logroño, La Rioja
Teléfono	N/A
Fax	N/A
Comentarios	www.unir.net

Organización	Veterinaria García-Ochoa
Dirección	Carretera de Casillas 6 local 3, 05420 Sotillo de la Adrada, Ávila
Teléfono	N/A
Fax	N/A
Comentarios	Ninguno

3 Descripción del sistema

3.1 Perspectiva del producto

PetChain es una aplicación web, que tiene como objeto mostrar las capacidades que posee la Blockchain para el desarrollo de aplicaciones de gestión tradicionales. No es objeto de la misma ofrecer la visión de que el modelo que se presenta "es mejor" que cualquier desarrollo web tradicional, sino que como nuevo paradigma de desarrollo, es necesario tenerlo en cuenta como un futuro modelo para la generación de aplicaciones de gestión corporativas.

Su ejecución se realiza íntegramente en un entorno local, permitiendo este modelo, la visualización de cada una de las partes que componen el desarrollo. Esto es posible gracias al uso del framework Truffle, que con el servidor Ganache que lleva incorporado, ofrece la posibilidad de crear una red Ethereum y con ello poder desarrollar, desplegar y testear Smartcontracts desarrollados con lenguaje Solidity.

3.2 Funciones del producto

PetChain es una aplicación que debe de permitir a los veterinarios:

- Identificar a las mascotas domesticas mediante un código alfanumérico único, así como asociarlas a una persona que actuará como dueño, y que estará identificado mediante DNI o Pasaporte y los datos personales (nombre, apellidos, domicilio, y teléfono de contacto).
- Modificar los datos sanitarios de vacunaciones,
- Modificar los datos personales del dueño de la mascota.
- Dar de baja a las mascotas del sistema.
- Visualizar el saldo en ethers y weis de cada uno de los veterinarios correspondientes a las transacciones realizadas.

3.3 Características del usuario

Los usuarios de PetChain serán los veterinarios, que a nivel nacional, deben de realizar acciones de tipo administrativo a la hora de gestionar las identificaciones de animales de comañía.

Como perfil tecnológico, deben tener competencias a la hora de manejar formularios web, sin necesitar unos conocimientos técnicos profundos, pues el modelo visual no difiere de cualquier página web existente en el mercado.

Por otro lado el tipo de información que es necesaria cumplimentar en la web, está dentro del dominio del sector veterinario, con lo que su cumplimentación no debería presentar problemas para el colectivo que usará la aplicación.

3.4 Restricciones

PetChain es un proyecto web que debe ser posible su visualización en toda clase de dispositivos, por lo tanto la estructura de las pantallas se debe de adaptar a la resolución de la pantalla en donde se ejecute, siendo el rango de estas desde dispositivos Smartphone, Tablets y PCs.

A nivel de seguridad, es necesario implementar características que permitan la autenticación y la autorización de los usuarios que intenten acceder a la aplicación, pero al ser una ejecución en un entorno local no es obligatorio el diseño de una base de datos SQL o [NoSQL](#), que albergue sus credenciales de acceso.

3.5 Descripción técnica del sistema

3.5.1 Front-End

La web se desarrollará bajo el modelo de visualización responsive, por lo que su diseño se deberá adaptar al tipo de resolución de la pantalla desde la que se esté accediendo. Para conseguir este objetivo se empleará en su diseño el framework open source [Twitter Bootstrap](#), ver. 4.3.1.

El desarrollo de las interfaces de usuario se realizará con [HTML5](#) estándar, [CSS3](#) y la programación de los scripts de cliente se realizará con Javascript ([ECMA Script 6](#)) y el framework de cliente [jQuery](#) versión 3.1.1., siendo uno de los objetivos del desarrollo el de aprovechar al máximo el paradigma de programación [MVC](#) (Model View Controller).

3.5.2 Seguridad

Como antes se indicaba, es necesario implementar un sistema que verifique la autenticación de los usuarios al sistema, así como la autorización para la realización de acciones asociadas al rol que tenga determinado.

El sistema de credenciales empleado para el proyecto actual debe quedar preparado para poder implementar una conexión a una base de datos [NoSQL](#), en el caso de que PetChain se instale en un entorno productivo. En la situación de ejecución actual, sólo se exigirá que los usuarios estén definidos en un fichero [JSON](#), almacenado dentro de la estructura local de proyecto.

Aunque no se exija el almacenamiento de credenciales en una base de datos, si es necesario que la aplicación tenga implementados sistemas de protección de [URL](#) mediante [JWT](#), así como la validación de usuarios mediante nombre de usuario y contraseña.

3.5.3 Back-End

La aplicación se ejecutará en una Blockchain desplegada en un entorno local. Esta red estará funcionando gracias a un servidor Ganache, el cual es un componente del framework de desarrollo y ejecución Truffle.

La conexión entre la capa de Back-End y el Front-End es posible mediante el uso del framework Web3.js. Este es un entorno que permite la interacción de Smartcontracts con el javascript alojado en el cliente y la ejecución de la lógica en ellos definida.

3.5.4 Entorno de ejecución

La aplicación debe de ejecutarse en cualquier navegador estándar (IE11, Chrome o Firefox) en un servidor local o localhost.

Para la ejecución de funcionalidades como la seguridad, lectura y escritura de ficheros y paso de mensajes entre pantallas, es necesario la realización de solicitudes POST y GET, por lo que la aplicación se va a desplegar con un servidor ExpressJS y bajo un entorno de ejecución NodeJS.

4 Objetivos del sistema

OBJ-0001	Alta de mascotas
----------	------------------

Versión	1.0 (04/05/2019)
Autores	• Carlos Escario Bajo
Fuentes	• Iciar García-Ochoa • Pedro Escario Bajo
Descripción	El sistema deberá <i>permitir el alta de una mascota en el sistema, asociándola a un código de chip. Este es un identificador con logitud fija de quince (15) caracteres alfanumérico.</i>
Subobjetivos	Ninguno
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	PD
Comentarios	Ninguno

OBJ-0002	Alta de dueños de mascotas.
Versión	1.0 (04/05/2019)
Autores	• Carlos Escario Bajo
Fuentes	• Iciar García-Ochoa • Pedro Escario Bajo
Descripción	El sistema deberá <i>permitir el dar de alta a una persona, asociándola a una mascota.</i>
Subobjetivos	Ninguno
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	PD
Comentarios	Ninguno

OBJ-0005	Modificación de datos
Versión	1.0 (04/05/2019)
Autores	• Carlos Escario Bajo
Fuentes	• Iciar García-Ochoa • Pedro Escario Bajo
Descripción	El sistema deberá <i>permitir modificar los datos asociados a las mascotas y a sus dueños. el veterinario modificador deberá pagar un canon fijo en weis al veterinario identificador.</i>
Subobjetivos	<ul style="list-style-type: none"> • [OBJ-0006] Modificación de datos de mascotas: El sistema deberá <i>permitir la modificación de los datos sanitarios de las mascotas, así como la inclusión de las vacunaciones periódicas obligatorias.</i> • [OBJ-0007] Modificación de datos de los dueños: El sistema deberá <i>permitir cambiar los datos correspondientes al dueño de la mascota, incluido el cambio a un nuevo propietario.</i>
Importancia	importante
Urgencia	inmediatamente
Estado	PD
Estabilidad	PD
Comentarios	Ninguno

OBJ-0008	Baja de mascotas
Versión	1.0 (05/05/2019)
Autores	• Carlos Escario Bajo

Fuentes	• Iciar García-Ochoa
Descripción	El sistema deberá <i>permitir dar de baja a una mascota del sistema, por motivos de defunción.</i>
Subobjetivos	Ninguno
Importancia	importante
Urgencia	PD
Estado	PD
Estabilidad	PD
Comentarios	Ninguno

OBJ-0009	Consulta de saldo
Versión	1.0 (05/05/2019)
Autores	• Carlos Escario Bajo
Fuentes	• Carlos Escario Bajo
Descripción	El sistema deberá <i>permitir consultar el saldo en ethers o weis que cada veterinario participante en el sistema, tiene por realizar acciones dentro del mismo.</i>
Subobjetivos	Ninguno
Importancia	importante
Urgencia	PD
Estado	PD
Estabilidad	PD
Comentarios	Ninguno

OBJ-0011	Control de acceso
Versión	1.0 (05/05/2019)
Autores	• Carlos Escario Bajo
Fuentes	• Carlos Escario Bajo
Descripción	El sistema deberá <i>permitir el acceso al sistema a aquellos usuarios que estén incluidos en el fichero de usuarios creado.</i>
Subobjetivos	Ninguno
Importancia	importante
Urgencia	inmediatamente
Estado	pendiente de validación
Estabilidad	alta
Comentarios	Ninguno

OBJ-0012	Visualización en múltiples dispositivos
Versión	1.0 (05/05/2019)
Autores	• Carlos Escario Bajo
Fuentes	• Carlos Escario Bajo
Descripción	El sistema deberá <i>estar diseñado para poder visualizarse en dispositivos del tipo Smartphone, Tablet y PC</i>
Subobjetivos	Ninguno
Importancia	importante
Urgencia	PD
Estado	PD

Estabilidad	PD
Comentarios	Ninguno

OBJ-0014	Navegación por la aplicación
Versión	1.0 (05/05/2019)
Autores	• Carlos Escario Bajo
Fuentes	• Carlos Escario Bajo
Descripción	El sistema deberá <i>permitir la navegación entre las opciones de alta, modificación y baja de mascotas, así como la consulta del saldo de weis.</i>
Subobjetivos	Ninguno
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	PD
Comentarios	Ninguno

OBJ-0013	Traspasar weis
Versión	1.0 (05/05/2019)
Autores	• Carlos Escario Bajo
Fuentes	• Carlos Escario Bajo
Descripción	El sistema deberá <i>traspasar weis desde el veterinario modificador hasta el veterinario identificador, siempre y cuando el modificador realice un cambio en los datos de una mascota que ya exista en el sistema.</i>
Subobjetivos	Ninguno
Importancia	importante
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	PD
Comentarios	Ninguno

OBJ-0015	Gestión de sesiones
Versión	1.0 (05/05/2019)
Autores	• Carlos Escario Bajo
Fuentes	?
Descripción	El sistema deberá <i>ser capaz de manejar las sesiones de usuario y el correcto funcionamiento de los navegadores.</i>
Subobjetivos	Ninguno
Importancia	importante
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	PD
Comentarios	Ninguno

5 Catálogo de requisitos del sistema.

5.1 Requisitos de información

Los requisitos de información son formas especializadas de requisitos que van a permitir la introducción de datos relativos a los actores que van a participar o componer el sistema.

IRQ-0001	Datos de usuario veterinario	
Versión	1.0 (05/05/2019)	
Autores	• Carlos Escario Bajo	
Fuentes	?	
Dependencias	• [FRQ-0001] Acceso al sistema	
Descripción	El sistema deberá almacenar la información correspondiente a <i>los usuarios veterinarios que usen el sistema</i> . En concreto:	
Datos específicos	<ul style="list-style-type: none"> • Nombre de usuario • Contraseña del usuario • Rol • Número de colegiado • Colegio 	
Tiempo de vida	Medio	Máximo
	PD	PD
Ocurrencias simultáneas	Medio	Máximo
	PD	PD
Importancia	vital	
Urgencia	PD	
Estado	PD	
Estabilidad	PD	
Comentarios	Ninguno	

IRQ-0002	Datos de mascota	
Versión	1.0 (05/05/2019)	
Autores	• Carlos Escario Bajo	
Fuentes	<ul style="list-style-type: none"> • Iciar García-Ochoa • Pedro Escario Bajo 	
Dependencias	Ninguno	
Descripción	El sistema deberá almacenar la información correspondiente a <i>a las mascotas que se añaden al sistema</i> . En concreto:	
Datos específicos	<ul style="list-style-type: none"> • N° de identificación • Fecha de implantación • Nombre • Raza • Peligrosa • Fecha de nacimiento • Capa • Pelo • Sexo • Pasaporte animal • Aptitud • Última vacunación antirrábica • Fecha de la última revisión • Número de colegiado última revisión • Cicatrices • Numero de certificado 	
Tiempo de vida	Medio	Máximo
	PD	PD
Ocurrencias simultáneas	Medio	Máximo

	PD	PD
Importancia	PD	
Urgencia	PD	
Estado	PD	
Estabilidad	PD	
Comentarios	Ninguno	

IRQ-0003	Datos del dueño	
Versión	1.0 (05/05/2019)	
Autores	• Carlos Escario Bajo	
Fuentes	?	
Dependencias	Ninguno	
Descripción	El sistema deberá almacenar la información correspondiente a PD . En concreto:	
Datos específicos	<ul style="list-style-type: none"> • Nombre • Apellidos • Número de documento • Teléfono • Provincia • Municipio • Código Postal • País 	
Tiempo de vida	Medio	Máximo
	PD	PD
Ocurrencias simultáneas	Medio	Máximo
	PD	PD
Importancia	PD	
Urgencia	PD	
Estado	PD	
Estabilidad	PD	
Comentarios	Ninguno	

IRQ-0004	Saldo	
Versión	1.0 (05/05/2019)	
Autores	• Carlos Escario Bajo	
Fuentes	?	
Dependencias	Ninguno	
Descripción	El sistema deberá almacenar la información correspondiente a <i>al saldo en ethers o weis correspondiente a los usuarios veterinarios que usen el sistema</i> . En concreto:	
Datos específicos	<ul style="list-style-type: none"> • Cuenta ethereum • Saldo en weis • Número de colegiado 	
Tiempo de vida	Medio	Máximo
	PD	PD
Ocurrencias simultáneas	Medio	Máximo
	PD	PD
Importancia	importante	
Urgencia	PD	
Estado	PD	

Estabilidad	PD
Comentarios	Ninguno

5.2 Requisitos funcionales

Los requerimientos funcionales son declaraciones de los servicios que proveerá el sistema, de la manera en que éste reaccionará a entradas particulares. En algunos casos, los requerimientos funcionales de los sistemas también declaran explícitamente lo que el sistema no debe hacer.

FRQ-0001	Acceso al sistema
Versión	1.0 (05/05/2019)
Autores	• Carlos Escario Bajo
Fuentes	• Iciar García-Ochoa • Pedro Escario Bajo
Dependencias	Ninguno
Descripción	El sistema deberá <i>permitir el acceso a los usuarios a través de un formulario en el que se debe introducir un nombre de usuario y una contraseña. Si los datos introducidos no son correctos, el sistema deberá mostrar en la pantalla un mensaje de error.</i>
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	PD
Comentarios	Ninguno

FRQ-0002	Identificación de veterinarios
Versión	1.0 (05/05/2019)
Autores	• Carlos Escario Bajo
Fuentes	• Iciar García-Ochoa
Dependencias	Ninguno
Descripción	El sistema deberá <i>deberá mostrar el nombre del veterinario y el colegio en donde está inscrito, con solo introducir el número de Colegiado.</i>
Importancia	importante
Urgencia	puede esperar
Estado	en construcción
Estabilidad	PD
Comentarios	Ninguno

FRQ-0003	Traspaso de weis
Versión	1.0 (05/05/2019)
Autores	• Carlos Escario Bajo
Fuentes	• Carlos Escario Bajo
Dependencias	Ninguno
Descripción	El sistema deberá <i>traspasar weis cuando se haga una modificación de los datos de una mascota, desde el veterinario modificador hasta el veterinario que dio de alta a la mascota.</i>
Importancia	importante
Urgencia	puede esperar
Estado	en construcción

Estabilidad	PD
Comentarios	Ninguno

FRQ-0004	Navegación
Versión	1.0 (05/05/2019)
Autores	• Carlos Escario Bajo
Fuentes	• Pedro Escario Bajo
Dependencias	Ninguno
Descripción	El sistema deberá <i>permitir en todo momento la navegación por el sistema a través de las opciones que se muestren en la pantalla de inicio o en un menú situado en la cabecera de la aplicación.</i>
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	PD
Comentarios	Ninguno

FRQ-0005	Campos no modificables en la Modificación.
Versión	1.0 (05/05/2019)
Autores	• Carlos Escario Bajo
Fuentes	• Iciar García-Ochoa
Dependencias	Ninguno
Descripción	El sistema deberá <i>impedir en la opción de Modificación de datos de la mascota, que el usuario cambie los datos asociados al veterinario identificador y los datos de nacimiento correspondientes a la mascota sobre la cual se están realizando modificaciones.</i>
Importancia	vital
Urgencia	puede esperar
Estado	en construcción
Estabilidad	PD
Comentarios	Ninguno

FRQ-0006	Campos no modificables en la Baja de mascotas
Versión	1.0 (05/05/2019)
Autores	• Carlos Escario Bajo
Fuentes	• Iciar García-Ochoa
Dependencias	Ninguno
Descripción	El sistema deberá <i>impedir en la opción de Baja de mascotas que el usuario cambie los datos asociados al veterinario identificador.</i>
Importancia	importante
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	PD
Comentarios	Ninguno

FRQ-0007	Consultar webs disponibles
Versión	1.0 (05/05/2019)

Autores	• Carlos Escario Bajo
Fuentes	• Carlos Escario Bajo
Dependencias	Ninguno
Descripción	El sistema deberá <i>visualizar un listado con el saldo de weis que tiene cada veterinario,</i>
Importancia	importante
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	PD
Comentarios	Ninguno

FRQ-0008	Validación de campos
Versión	1.0 (05/05/2019)
Autores	• Carlos Escario Bajo
Fuentes	• Pedro Escario Bajo
Dependencias	Ninguno
Descripción	El sistema deberá <i>mostrar en pantalla un mensaje que informa de aquellos campos que son obligatorios y no han sido rellenados en el momento de enviar el formulario.</i>
Importancia	vital
Urgencia	inmediatamente
Estado	pendiente de verificación
Estabilidad	PD
Comentarios	Ninguno

5.3 Requisitos no funcionales

Son aquellos requerimientos que no se refieren directamente a las funciones específicas que entrega el sistema, sino a las propiedades emergentes de éste como la fiabilidad, la respuesta en el tiempo y la capacidad de almacenamiento. De forma alternativa, definen las restricciones del sistema como la capacidad de los dispositivos de entrada/salida y la representación de datos que se utiliza en la interface del sistema, etc.

NFR-0001	Visualización
Versión	1.0 (05/05/2019)
Autores	• Carlos Escario Bajo
Fuentes	• Carlos Escario Bajo
Dependencias	Ninguno
Descripción	El sistema deberá <i>ajustar la visualización de las interfaces gráficas en función del modelo de desarrollo responsive. Es decir será el cliente quien detecte el tipo de dispositivo desde el que se está accediendo a la web.</i>
Importancia	quedaría bien
Urgencia	puede esperar
Estado	en construcción
Estabilidad	PD
Comentarios	Ninguno

NFR-0002	Moneda de pago
Versión	1.0 (05/05/2019)

Autores	• Carlos Escario Bajo
Fuentes	?
Dependencias	Ninguno
Descripción	El sistema deberá <i>permitir el pago en weis, siendo la conversión en ethers de 1 ether = 1×10^{18} wei.</i>
Importancia	importante
Urgencia	PD
Estado	en construcción
Estabilidad	PD
Comentarios	Ninguno

NFR-0003	Tiempo de sesión
Versión	1.0 (05/05/2019)
Autores	• Carlos Escario Bajo
Fuentes	• Carlos Escario Bajo
Dependencias	Ninguno
Descripción	El sistema deberá <i>controlar que el tiempo que puede estar una sesión inactiva, será como máximo de 15 minutos.</i>
Importancia	importante
Urgencia	puede esperar
Estado	en construcción
Estabilidad	PD
Comentarios	Ninguno

NFR-0004	Usuarios autorizados
Versión	1.0 (05/05/2019)
Autores	• Carlos Escario Bajo
Fuentes	• Carlos Escario Bajo
Dependencias	Ninguno
Descripción	El sistema deberá <i>tener almacenados los usuarios de la aplicación en un fichero en formato JSON, en donde se indique su numero de colegiado, nombre de usuario, contraseña y número de colegiado.</i>
Importancia	vital
Urgencia	inmediatamente
Estado	pendiente de validación
Estabilidad	PD
Comentarios	Ninguno

NFR-0005	Número de cuentas
Versión	1.0 (05/05/2019)
Autores	• Carlos Escario Bajo
Fuentes	• Carlos Escario Bajo
Dependencias	Ninguno
Descripción	El sistema deberá <i>tener definidas por defecto al menos 100 cuentas en el servidor Ganaché.</i>
Importancia	vital

Urgencia	inmediatamente
Estado	en construcción
Estabilidad	PD
Comentarios	Ninguno

NFR-0006	Autenticacion JWT
Versión	1.0 (05/05/2019)
Autores	• Carlos Escario Bajo
Fuentes	• Carlos Escario Bajo
Dependencias	Ninguno
Descripción	El sistema deberá <i>tener un sistema de autenticación mediante JWT, que se almacena como variable en la SessionStorage.</i>
Importancia	importante
Urgencia	puede esperar
Estado	en construcción
Estabilidad	PD
Comentarios	Ninguno

6 Matriz de rastreabilidad

TRM-0001	FRQ-0001	FRQ-0002	FRQ-0003	FRQ-0004	FRQ-0005	FRQ-0006	FRQ-0007	FRQ-0008	NFR-0001	NFR-0002	NFR-0003	NFR-0004	NFR-0005	NFR-0006
OBJ-0001	-	↑	-	-	-	-	-	↑	-	-	-	-	-	-
OBJ-0002	-	-	-	-	-	-	-	↑	-	-	-	-	-	-
OBJ-0005	-	↑	↑	-	↑	-	↑	↑	-	-	-	-	-	-
OBJ-0006	-	-	-	-	-	-	-	↑	-	-	-	-	-	-
OBJ-0007	-	-	-	-	-	-	-	↑	-	-	-	-	-	-
OBJ-0008	-	-	-	-	-	↑	-	↑	-	-	-	-	-	-
OBJ-0009	-	-	-	-	-	-	↑	-	-	-	-	-	-	-
OBJ-0011	↑	-	-	-	-	-	-	-	-	-	-	↑	-	↑
OBJ-0012	-	-	-	-	-	-	-	-	↑	-	-	-	-	-
OBJ-0013	-	-	↑	-	-	-	-	-	-	↑	-	-	-	-
OBJ-0014	-	-	-	↑	-	-	-	-	-	-	-	-	-	-
OBJ-0015	-	-	-	-	-	-	-	-	-	-	↑	-	-	-

Matriz de rastreabilidad: PetChain. Matriz de rastreabilidad

7 Glosario de términos

CSS3: Cascade Style Sheet ver.3 (Hojas de estilo en cascada)

HTML5:Hyper Text Markup Language versión 5 (Lenguaje de marcado en hipertexto).

Twitter Bootstrap:Framework CSS de código abierto desarrollado por la compañía Twitter.

ECMA Script 6:Especificación del lenguaje de programación Javascript, realizado por el consorcio ECMA.

jQuery:Biblioteca open source de javascript y multiplataforma.

MVC:Modelo Vista Controlador (Model View Controller)

NoSQL:Gestores de bases de datos que almacenan la información en forma de objetos y no emplean el lenguaje SQL para la realización de consultas.

JSON:Notación de objetos Javascript (JavaScript Object Notation).

URL:Localizador de recursos uniforme (Uniform Resource Locator)

JWT:JavaScript Web Token