|  |
| --- |
| DUOC UC - Escuela de informática y telecomunicaciones |
| Propuesta de Proyecto y Especificación de Requisitos de Software |
| *Proyecto: [PetShopiety]* |
|  |
| **Revisión*: [03]*** |
| **26/6/23** |

|  |
| --- |
| Planificación y Especificación de Requisitos según estándares; IEEE 830, ISO9000 y PMI. |

Contenido

[DUOC UC - Escuela de informática y telecomunicaciones 1](#_Toc137493859)

[Ficha del documento 3](#_Toc137493860)

[1. Introducción 3](#_Toc137493861)

[1.1. Propósito 3](#_Toc137493862)

[1.2. Ámbito del Sistema 4](#_Toc137493863)

[1.3. Definiciones, Acrónimos y Abreviaturas 4](#_Toc137493864)

[1.4. Referencias 4](#_Toc137493865)

[1.5. Visión General del Documento 4](#_Toc137493866)

[2. Descripción General 5](#_Toc137493867)

[2.1. Perspectiva del Producto 5](#_Toc137493868)

[2.2. Funciones del Producto 5](#_Toc137493869)

[2.3. Características de los Usuarios 5](#_Toc137493870)

[2.4. Restricciones 5](#_Toc137493871)

[2.5. Suposiciones y Dependencias 6](#_Toc137493872)

[2.6. Requisitos Futuros 6](#_Toc137493873)

[3. Requisitos Específicos 7](#_Toc137493874)

[3.1 Requisitos comunes de las interfaces 8](#_Toc137493875)

[3.1.1 Interfaces de usuario 8](#_Toc137493876)

[3.1.2 Interfaces de hardware 8](#_Toc137493877)

[3.1.3 Interfaces de software 8](#_Toc137493878)

[3.1.4 Interfaces de comunicación 8](#_Toc137493879)

[3.2 Requisitos funcionales 8](#_Toc137493880)

[3.3 Requisitos no funcionales 9](#_Toc137493881)

[3.3.1 Requisitos de rendimiento 9](#_Toc137493882)

[3.3.2 Seguridad 9](#_Toc137493883)

[3.3.3 Fiabilidad 10](#_Toc137493884)

[3.3.4 Disponibilidad 10](#_Toc137493885)

[3.3.5 Mantenibilidad 10](#_Toc137493886)

[3.3.6 Portabilidad 10](#_Toc137493887)

[3.4 Otros Requisitos 10](#_Toc137493888)

[4. Propuesta de Planificación 11](#_Toc137493889)

[4.1 Descripción general acerca de la Planificación 11](#_Toc137493890)

[4.1.2 Definición del Equipo de Trabajo 11](#_Toc137493891)

[4.1.3 Definición de Actividades principales del Proyecto 11](#_Toc137493892)

[4.2 Plan de Control de Cambio 11](#_Toc137493893)

# Ficha del documento

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Revisión** | **Autor** | **Modificación** |
| *07/04* |  |  |  |
| *14/05* |  |  |  |
| *26/06* |  |  |  |

Documento validado por las partes en fecha:

**Integrantes:**

|  |  |
| --- | --- |
| **Nombre Integrante del Equipo** | **Rol Definido** |
| *Antonia valencia* | *Diseñadora/Constructora* |
| *Jose Morales* | *Constructor/Diseñador* |

# 1. Introducción

Como introducción, nos vamos a basar en el caso de Un grupo de amigas que comenzara con la venta de productos para mascotas ya sean bandanas, correas e identificaciones, etc.… con el objetivo de ayudar a la fundación sin fines de lucro. Para este emprendimiento empezó a través de las redes sociales, pero al no tener todas las seguridades necesarias, necesitaban optar por un nuevo sistema el cual cumpla con todos los requisitos para la compra y venta de los productos

## 1.1. Propósito

El propósito para este proyecto, se basa en crear una página web la cual opte con un sistema de seguridad y una base de datos con el objetivo de cumplir con los requisitos solicitados y con los que contiene una tienda online.

## 1.2. Ámbito del Sistema

* El sistema debe tener un registro del cliente(usuario) para el acceso de los servicios de ventas
* El sistema le permita al cliente generar una compra, descontando los productos del stock una vez creada la venta y también realizar descuentos en caso de que se encuentre algún tipo de una promoción o de si el cliente .
* Para los usuarios registrados el sistema debe permitir ver un historial de ventas realizadas, además, del estado del despacho de las ventas.
* Para el despacho de los productos, el sistema también pueda entregar un código de seguimiento del producto hacia el despacho en el caso de los envíos a domicilio de clientes registrados, mostrando fecha y hora en que se toma el pedido, se despacha y es recibido por el cliente, cerrando el ciclo. sistema.
* Contar con una plataforma en internet para promover sus productos a usuarios registrados, que indique el stock real de productos disponibles
* Entregar el servicio venta en línea.
* El cliente tenga permitido monitorear sus propias.
* Mejorar el proceso de logística de sus productos a los clientes mediante el seguimiento del despacho.
* Permitir generar descuentos o promociones en la página principal
* Permitir subscribirse con una donación a la fundación sin fines de lucro, esta información ira directamente a la fundación. Con el fin de motivar a sus clientes y que se les pueda otorgar un 5% de descuento a los clientes que estén suscritos a la página.
* Debe llevar un carrito de compra el cual permitir agregar y eliminar las compras
* Tenga la opción de que la pagina pueda está en modo oscuro(a elección personal)
* Que tenga algún tipo de interacción con la base de datos:

1. Agregar
2. Editar
3. Eliminar
4. Obtener

* Mínimo de 3 tablas con el modelo Django
* Tenga un servicio API rest junto con Django para obtener todos los productos de la tienda
* Un control de usuario(login):
* Los mantenedores solo se podrán visualizar solo si se accede como un administrador
* Al realizarse una compra, el stock de los productos debe disminuirse, con un top max. De 0
* La pág. podrá mostrar los productos con un stock mayor a 0
* Mensajes:

1. Alertas
2. Errores

## 1.3. Definiciones, Acrónimos y Abreviaturas

Lo más importante y que también se podría considerar clave para el sistema solicitado es que tenga una base de batos con todo tipo de registro, los dato del cliente, los productos que el cliente haya comprado o mantenido en el carro, la localización de los productos ya sea para entrega a domicilió y que también tenga la opción de eliminar la suscripción del cliente y no quede ningún tipo de registro a la hora de cerrar sesión.

## 1.4. Referencias

En esta subsección se mostrará una lista completa de todos los documentos referenciados en la ERS.

## 1.5. Visión General del Documento

En esta subsección se describe brevemente los contenidos y la organización del resto de la ERS.

# 2. Descripción General

Como el sistema va a requerir una base de datos para “todo”, también se va a requerir un sistema de seguridad que pueda evitar todo tipo de hackeo o de algún tipo de error que pueda llegar. Tener el sistema. Para la seguridad para la base de datos se va a utilizar Oracle, el cual pueda contar con todo registro que vaya agregando o eliminando datos hacia el sistema sin requerimiento de una segunda copia de seguridad.

## 2.1. Perspectiva del Producto

Para iniciar con el proyecto empezara con la base de la pág. que sería la visualización de algunos productos, la opción de búsqueda, del tipo de producto(ya sea perro o gato), en la página se podrá visualizar además de productos y del nombre de la tienda, un reloj con la hora actual, acompañado del botón que dará el uso del modo oscuro(el cual eso ya seria a gusto del que llegue a usar la pág.). Después a través de un carrito de compra tendrá la opción de agregar los productos que el cliente tenga interés de comprar y a la vez tendrá la opción de eliminar en caso de que ya no lo necesite o encontró uno que le seria por así decirlo más útil Ya con las compras seleccionadas, deberá iniciar sesión a través de un correo(que exista) y de una contraseña que debe crear el usuario más el número de teléfono que también debe ser valido los botones de inicio de sesión o registro. Después se implementaría ya una base de datos más avanzada y con más seguridad para así evitar todo tipo de hackeo o de estafa y seguir los requisitos que va necesitar el sistema a futuro.

## 2.2. Funciones del Producto

Al ingresar a la página, automáticamente se podrá visualizar alguno de los productos que ya tiene la tienda, junto con el carrito de compras el cual estará listando los productos que se agreguen o eliminen, contara con un botón que lo mandara a una vista para el ingreso o el registro ya sea tenga cuenta en la página o no. También se visualizara tipo de productos ya sea para perros o gatos y en el caso de buscar un producto en específico, también contara con la opción de búsqueda de producto. Para el ingreso los únicos datos solicitados será el correo del cliente y la contraseña que asigno, en el caso de no tener ninguna cuenta en la pág. Tendrá la opción de un registro donde solo le pedirá algunos datos como el correo, el teléfono, el Rut y la fecha de nacimiento

## 2.3. Características de los Usuarios

* Registro: El usuario tendrá un registro con sus datos que ya relleno a la hora de iniciar la sesión en la pág.
* Base de datos: Tendrá una base de todo lo que haya buscado o comprado en la página.
* Registro de producto: ya sea un producto comprado o aun guardado en el carrito de compras y donde llega a usar un descuento.
* Al realizar una compra, se actualizara el stock del producto
* Recibirá todo tipo de mensaje o alertas ya sea de alguna información registrada que sea incorrecta, o de alguna compra fallida de tal producto

## 2.4. Restricciones

Esta subsección describirá aquellas limitaciones que se imponen sobre los desarrolladores del producto:

• Políticas de la empresa.

• Limitaciones del hardware.

• Interfaces con otras aplicaciones.

• Operaciones paralelas.

• Funciones de auditoría.

• Funciones de control.

• Lenguaje(s) de programación.

• Protocolos de comunicación.

• Requisitos de habilidad.

• Criticidad de la aplicación.

• Consideraciones acerca de la seguridad.

## 2.5. Suposiciones y Dependencias

Como se nombró, anteriormente, la pág., se le a agregado la función Django y un api Request, para que su registro sea más fácil a la hora de guardar y a la hora de eliminar, junto con que el usuario ahora tendrá la implementación de agregar, eliminar y editar un producto. También se a implementado el sistema de base de datos Cetecom, el cual permite la creación de tablas y permite el poder agregar productos con los datos necesarios que sería el precio, el nombre del producto y la descripción.

## 2.6. Requisitos Futuros

Esta subsección esbozará futuras mejoras al sistema, que podrán analizarse e implementarse en un futuro.

# 3. Requisitos Específicos

* Carousel: Permitir la visualización de las imágenes
* Modal: Ventana emergente que esta encima de la pg. Principal y pide de manera adicional la acción del usuario.
* Card: Contiene la función relacionada como el título, la imagen del producto, el texto y el enlace
* Image: Visualización del producto o relacionada a la idea de la pág.(mascotas)
* Navbar: Barra de navegación el cual contendrá los botones que dirigen a las demás vistas.
* Sea una web responsiva, que se adapte a distintos tamaños de pantalla.
* El carrito debe llevar la funcionalidad de:

1. Agregar productos
2. Editar los productos
3. Listar los productos.Contener un carrito de compras(Que tengan las funcionalidades en un localStorage).

* Llegue a tener una visualizacion de un reloj que este a tiempo real
* 3) Que se pueda validar todo tipo de formulario existentes de la web, como java script y jquery.
* Deben consumir un servicio web (API)
* Agregar funcionalidades a los elementos html con jquery, para que exista un cambio en ellos, sin que se recargue la página web.
* agregar una clase a un elemento html, que ustedes hayan desarrollado en la hojade estilos de css, y asi cambiar la apariencia del elemento desde java script.
* Debe tener la funcionalidad de que permita el modo oscuro o modo claro en la pagina web, osea que pueda cambiar el tono de los colores.

## 3.1 Requisitos comunes de las interfaces

Descripción detallada de todas las entradas y salidas del sistema de software.

### 3.1.1 Interfaces de usuario

* Inicio: Lo principal que tenga la pág. y los productos tanto recientes como los más pedidos
* Catálogo de productos: Se vea como para que esta catalogado cada producto que en este caso sería si es para perros o gatos
* Inicio: Si ya tiene algún tipo de usuario en la pág., que ingrese su correo y su contra
* Registro: Si no tiene una cuenta registrada, puede crearse solo pidiéndole algunos datos personales
* Perfil de usuario: Se visualiza algunos datos(que no serán los mas personales) y el tipo de producto que mas compra
* Contacto: Una visualización donde se vea a quien uno se puede llegar a contactar y la dirección de la tienda

### 3.1.2 Interfaces de hardware

Especificar las características lógicas para cada interfaz entre el producto y los componentes de hardware del sistema. Se incluirán características de configuración.

### 3.1.3 Interfaces de software

Indicar si hay que integrar el producto con otros productos de software.

* Para cada producto de software debe especificarse lo siguiente:
* Descripción del producto software utilizado
* Propósito del interfaz
* Definición del interfaz: contiendo y formato

### 3.1.4 Interfaces de comunicación

Describir los requisitos del interfaz de comunicación si hay comunicaciones con otros sistemas y cuáles son los protocolos de comunicación.

## Requisitos funcionales

* Registro de usuarios: Los usuarios antes de empezar una compra, deben ingresar algún tipo de cuenta con información requerida que sería un correo, una contraseña y un número de teléfono
* Inicio de sesión: Los usuarios que ya estén registrados pueden acceder solo ingresando el correo y la contraseña
* Búsqueda de productos: Los usuarios pueden buscar y filtrar productos según la categoría, ya sea perro o gato
* Detalle de producto: El usuario podrá visualizar las caracts. Del producto, el precio la descripción, etc…
* Carrito de compras: A la hora de una compra, el usuario puede guardar los productos que está interesado en comprar a través de un carrito de compra, e incluso puede llegar a eliminar si ya no le llega a interesar
* Proceso del pago: Y a con los productos seleccionados para la compra, el usuario debe registrar un método de pago y algunos tanto como el tipo de tarjeta, los códigos y contraseña que eso ya seria información confidencial
* Gestión de pedidos: El usuario ya con los productos pagados, tendrá la opción de tener un envió a alguna dirección que el ingrese ya sea su propio despacho o el de alguien más, o si prefiere ir a retirar a la misma tienda
* Opiniones y valoraciones: el Usuario puede dejar una opinión acerca del producto para así informar al siguiente usuario que esté interesado en tal producto o una opinión cerca de la tienda en si
* Notificaciones: También estará la opción de que a los usuarios ya registrados, a través de sus correos, les llegue algún tipo de información o beneficio que les pueda llegar a gustar, tanto como ofertas o beneficios en si
* Información: La pág. también contara con las preguntas frecuentes que leguen a realizar los usuarios, las políticas de devoluciones, privacidad y de la tienda y algún contacto en caso de que necesiten comunicarse con alguien

## 3.3 Requisitos no funcionales

### 3.3.1 Requisitos de rendimiento

Especificación de los requisitos relacionados con la carga que se espera tenga que soportar el sistema. Por ejemplo, el número de terminales, el número esperado de usuarios simultáneamente conectados, número de transacciones por segundo que deberá soportar el sistema, etc.

Todos estos requisitos deben ser mesurables. Por ejemplo, indicando “el 95% de las transacciones deben realizarse en menos de 1 segundo”, en lugar de “los operadores no deben esperar a que se complete la transacción”.

### 3.3.2 Seguridad

Especificación de elementos que protegerán al software de accesos, usos y sabotajes maliciosos, así como de modificaciones o destrucciones maliciosas o accidentales. Los requisitos pueden especificar:

* Empleo de técnicas criptográficas.
* Registro de ficheros con “logs” de actividad.
* Asignación de determinadas funcionalidades a determinados módulos.
* Restricciones de comunicación entre determinados módulos.
* Comprobaciones de integridad de información crítica.

### 3.3.3 Fiabilidad

Especificación de los factores de fiabilidad necesaria del sistema. Esto se expresa generalmente como el tiempo entre los incidentes permisibles, o el total de incidentes permisible.

### 3.3.4 Disponibilidad

Especificación de los factores de disponibilidad final exigidos al sistema. Normalmente expresados en % de tiempo en los que el software tiene que mostrar disponibilidad.

### 3.3.5 Mantenibilidad

Identificación del tipo de mantenimiento necesario del sistema.

Especificación de quien debe realizar las tareas de mantenimiento, por ejemplo usuarios, o un desarrollador.

Especificación de cuándo debe realizarse las tareas de mantenimiento. Por ejemplo, generación de estadísticas de acceso semanales y mensuales.

### 3.3.6 Portabilidad

Especificación de atributos que debe presentar el software para facilitar su traslado a otras plataformas u entornos. Pueden incluirse:

* Porcentaje de componentes dependientes del servidor.
* Porcentaje de código dependiente del servidor.
* Uso de un determinado lenguaje por su portabilidad.
* Uso de un determinado compilador o plataforma de desarrollo.
* Uso de un determinado sistema operativo.

## 3.4 Otros Requisitos

Cualquier otro requisito que no encaje en otra sección.

# 4. Propuesta de Planificación

## 4.1 Descripción general acerca de la Planificación

[Insertar una descripción de cómo se abordará el trabajo en cuanto a los días totales estimados y las personas involucradas en su ejecución, las buenas prácticas y condiciones necesarias a considerar para implementar para su buen término]

### 4.1.2 Definición del Equipo de Trabajo

El equipo de trabajo, básicamente se separaba en alguien se encargaba de lo teórico(tanto de informe o algo en específico) y uno construía. Cada uno estaba al tanto de los avances, lo que se llegaba a agregar, lo que se iba cambiando, lo que se iba eliminando para el proyecto para así estar al tanto de todo. Cuando uno tenía duda o algún tipo de consulta de algo, el otro integrante buscaba o le explicaba la pregunta presentada, ya que, es una manera para que así ambos sepan tanto, lo que hay en el proyecto, o de la materia en general

### 4.1.3 Definición de Actividades principales del Proyecto

La actividades principales para este proyecto, creo que era cada vez que el profe enseñaba algo lo cual uno sabía que después lo iba a pedir para una siguiente entrega, era saber, como después uno lo agregaba en el proyecto. Este tipo de actividad era la principal porque era saber o averiguar cómo agregar lo que se aprendía en clase y obteniendo el resultado de que eso no llegara a tener algún error o llegue a cambiar algo la pág., lo cual eso se requería ver paso por paso, las pruebas de si funcionaba o no, o que habría que llegar a cambiar para que el proyecto llegara a. funcionar con lo que se iba implementando

## 4.2 Plan de Control de Cambio

[Se recomienda primero describir los tipos de cambio que se podrán resolver y sus alcances]

[Insertar Tabla de Control de Cambios]

[ Obs.

Insertar Descripción de los aspectos del desarrollo en los que se permitirá aplicar cambios como parte del Desarrollo del Software definiendo sus alcances y limitaciones asociadas.

El control de cambios es una actividad paralela al desarrollo del proyecto que responde a eventos que surgen del mismo, sea por requerimientos propios del usuario o por mejoras o correcciones detectadas por el mismo equipo del proyecto.

 Se describe de manera independiente de las demás fases de la metodología pues puede ser aplicada indistintamente a proyectos en marcha o proyectos ya implementados, y porque es necesario resaltar su importancia y no relegarla como una actividad posterior al desarrollo, sino reconocerla como una actividad que debe estar definida, presente y es crítica desde el inicio del proyecto.  Deberá describir que tipo aspectos Funcionalidades y no funcionales se podrán modificar con cambio, en que instancia de proyecto se podrán aplicar y que motivos los validarían para ser aplicables y en qué caso no será posible aplicar cambios.

Luego esto se debe complementar con la observación de que en el anexo encontrarán la Planilla de Control de Cambio con los Tipos de Cambio que podrán aplicarse en la cual posteriormente se debe completar la planilla al ejecutarse la instancia. ]