|  |
| --- |
| DUOC UC - Escuela de informática y telecomunicaciones |
| Propuesta de Proyecto y Especificación de Requisitos de Software |
| *Proyecto: Espacio Huerta* |
|  |
| **Revisión*: [01]*** |
| **07/04/2023** |

**

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

Contenido

[Ficha del documento 3](#_Toc473208938)

[1. Introducción 4](#_Toc473208939)

[1.1. Propósito 4](#_Toc473208940)

[1.2. Ámbito del Sistema 4](#_Toc473208941)

[1.3. Definiciones, Acrónimos y Abreviaturas 4](#_Toc473208942)

[1.4. Referencias 4](#_Toc473208943)

[1.5. Visión General del Documento 4](#_Toc473208944)

[2. Descripción General 5](#_Toc473208945)

[2.1. Perspectiva del Producto 5](#_Toc473208946)

[2.2. Funciones del Producto 5](#_Toc473208947)

[2.3. Características de los Usuarios 5](#_Toc473208948)

[2.4. Restricciones 5](#_Toc473208949)

[2.5. Suposiciones y Dependencias 6](#_Toc473208950)

[2.6. Requisitos Futuros 6](#_Toc473208951)

[3. Requisitos Específicos 7](#_Toc473208952)

[3.1 Requisitos comunes de las interfaces 8](#_Toc473208953)

[3.1.1 Interfaces de usuario 8](#_Toc473208954)

[3.1.2 Interfaces de hardware 8](#_Toc473208955)

[3.1.3 Interfaces de software 8](#_Toc473208956)

[3.1.4 Interfaces de comunicación 8](#_Toc473208957)

[3.2 Requisitos funcionales 8](#_Toc473208958)

[3.3 Requisitos no funcionales 9](#_Toc473208959)

[3.3.1 Requisitos de rendimiento 9](#_Toc473208960)

[3.3.2 Seguridad 9](#_Toc473208961)

[3.3.3 Fiabilidad 10](#_Toc473208962)

[3.3.4 Disponibilidad 10](#_Toc473208963)

[3.3.5 Mantenibilidad 10](#_Toc473208964)

[3.3.6 Portabilidad 10](#_Toc473208965)

[3.4 Otros Requisitos 10](#_Toc473208966)

[4. Propuesta de Planificación 11](#_Toc473208967)

[4.1 Descripción general acerca de la Planificación 11](#_Toc473208968)

[4.1.2 Definición del Equipo de Trabajo 11](#_Toc473208969)

[4.1.3 Definición de Actividades principales del Proyecto 11](#_Toc473208970)

[4.1.4 Diagrama EDT 11](#_Toc473208971)

[4.1.5 Carta Gantt 11](#_Toc473208972)

[4.1.6 Resumen Costos del Desarrollo del Proyecto 11](#_Toc473208973)

[4.2 Plan de Control de Cambio 12](#_Toc473208974)

[5. Anexos 12](#_Toc473208975)

[5.1 Acta de Proyecto 12](#_Toc473208976)

[5.2 Matriz Especificación de Requerimientos 12](#_Toc473208977)

[5.3 Diagrama de Casos de Uso General 12](#_Toc473208978)

[5.4 Planilla Casos de Uso 12](#_Toc473208979)

[5.5 Prototipado de Software 13](#_Toc473208980)

[5.6 Resultado Análisis de Calidad Diagramas Modelamiento 13](#_Toc473208981)

[5.7 Resultado Análisis de Calidad Prototipado No funcional del Sistema 13](#_Toc473208982)

[5.8 Planilla entregables del Proyecto 13](#_Toc473208983)

[5.9 Matriz de Control de Cambios 13](#_Toc473208984)

[5.10 Matriz EDT. Planilla Detallada Cálculo de Esfuerzo 13](#_Toc473208985)

# Ficha del documento

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Revisión** | **Autor** | **Modificación** |
| *07-04-2023* | *1.1* | *Diego De La Vega* | *Complemento* |
| *9-05-2023* | *1.2* | *Diego De La Vega* | *Requisitos Funcionales* |
| *13-05-2023* | *1.3* | *Diego De La Vega* | *Requisitos no Funcionales, carta Gantt* |
| *24/06/2023* | *2.0* | *Diego De La Vega* | *Propuesta de planificación* |

Documento validado por las partes en fecha:

**Integrantes:**

|  |  |
| --- | --- |
| **Nombre Integrante del Equipo** | **Rol Definido** |
| *Raúl Valenzuela* | *Jefe De Proyecto* |
| *Diego De la Vega* | *Documentador* |
|  |  |
|  |  |
|  |  |

# 1. Introducción

## 1.1. Propósito

Este proyecto esta destinado a mejorar y optimizar la venta de productos de jardinería, para los vecinos de la comunidad, siendo como principal objetivo el desarrollar una plataforma web para la venta de productos como maceteros, tierra de horas, etc.

## 1.2. Ámbito del Sistema

* El Sistema tendrá como nombre Espacio Huerta
* El sistema tendrá como funcionalidades

1. El sistema permite el registro, modificación y eliminación de usuario
2. El sistema permite realizar compras descontando los productos en stock una vez creada la solicitud de venta, realizando los descuentos pertinentes a la venta del producto
3. El sistema permite el seguimiento de compras de una cuenta hora y fecha de la compra, fecha del despacho y recepción del cliente cerrando el pedido
4. El sistema permite a los usuarios registrados la opción de subscribirse o des suscribirse a la FSL

* Este proyecto esta destinado a ayudar a una FSL y a los vecinos de la comunidad generando un espacio virtual donde podrán generar ganancias con la venta de artículos y ayudar a los beneficiados por la fundación.  
  con este proyecto se espera optimizar la compra y venta de sus artículos proporcionando un sistema simple y amigable con el usuario

## 1.3. Definiciones, Acrónimos y Abreviaturas

* Fundación sin fines de lucro (FSL)

## 1.4. Referencias

## 1.5. Visión General del Documento

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

En este documento se describe el proyecto y sus funcionalidades contando con descripciones de las funcionalidades del mismo

# 2. Descripción General

## 2.1. Perspectiva del Producto

* El sistema en esta primera versión no interactúa con otro sistema es independiente

## 2.2. Funciones del Producto

* Las funciones del sistema son

1. Registra, almacena, modifica y elimina usuarios
2. Crea solicitudes de venta, registra la información y proporciona detalles de esta
3. Gestión de stock de la tienda
4. Gestión de suscripción a la FSL (añadir, modificar, eliminar)
5. Realizar validaciones de suscripción a la FSL

* El sistema podrá registrar, modificar y eliminar cuentas de posibles clientes, contara con un sistema de control de stock que facilitara la venta de artículos promocionados en la plataforma, contara con la posibilidad de gestión de suscripciones para la FSL

## 2.3. Características de los Usuarios

El sistema deberá ofrecer una interfaz de usuario intuitivo, fácil de aprender y sencillo de manejar. deberá presentar un alto grado de usabilidad. Lo deseable sería que un usuario nuevo se familiarizase con el sistema en una o dos horas.

## 2.4. Restricciones

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

* Lenguaje(s) de programación.
* Consideraciones acerca de la seguridad.

## 2.5. Suposiciones y Dependencias

* Se asume que los usuarios tengan el conocimiento de la FSL, los usuarios deberían tener el conocimiento básico para completar los formularios que se deban llenar para la compra o suscripción en la plataforma
* el sistema depende de el sistema de la FLS para validar si el usuario esta suscrito

## 2.6. Requisitos Futuros

* El sistema en un futuro deberá contar con una base de datos donde se podrá facilitar acceso y movimiento de datos relacionados con la plataforma y la FSL

# 3. Requisitos Específicos

RF1: Registro de usuarios

El sistema debe permitir a los usuarios registrarse proporcionando su nombre, dirección de correo electrónico y contraseña.

El sistema debe validar la dirección de correo electrónico para asegurar su formato correcto y evitar duplicados en la base de datos.

RF2: Autenticación de usuarios

El sistema debe permitir a los usuarios iniciar sesión utilizando su nombre de usuario y contraseña.

El sistema debe almacenar de forma segura las contraseñas de los usuarios.

El sistema debe proporcionar un mecanismo para recuperar la contraseña en caso de olvido, como el envío de un correo electrónico de restablecimiento de contraseña.

RF3: Catálogo de productos

El sistema debe mostrar un catálogo de productos que incluya imágenes, nombres, descripciones y precios de los productos disponibles para la venta.

El sistema debe permitir a los usuarios filtrar y ordenar los productos por categoría, precio, popularidad, etc.

El sistema debe mostrar la disponibilidad en tiempo real de cada producto, evitando que los usuarios realicen pedidos de productos agotados.

RF4: Carrito de compras

El sistema debe permitir a los usuarios agregar productos al carrito de compras, especificando la cantidad deseada.

El sistema debe permitir a los usuarios ver el contenido actualizado de su carrito de compras en cualquier momento.

El sistema debe permitir a los usuarios eliminar productos individuales o vaciar completamente su carrito de compras.

RF5: Proceso de pago

El sistema debe proporcionar una pasarela de pago segura que permita a los usuarios realizar transacciones utilizando tarjetas de crédito, PayPal u otros métodos de pago aceptados.

El sistema debe garantizar la seguridad de la información de pago, utilizando una capa de conexión segura (SSL/TLS) y cumpliendo con los estándares de seguridad de la industria.

El sistema debe enviar un correo electrónico de confirmación de pedido al usuario después de que se haya completado el proceso de pago.

RF6: Gestión de pedidos

El sistema debe permitir a los usuarios ver el historial de pedidos y el estado actual de sus pedidos, incluyendo la información de envío y el número de seguimiento, si está disponible.

El sistema debe enviar notificaciones por correo electrónico al usuario cuando haya actualizaciones importantes sobre el estado de su pedido, como el envío o la entrega.

El sistema debe permitir a los usuarios realizar devoluciones y solicitar reembolsos, siguiendo una política de devolución predefinida.

## 3.1 Requisitos comunes de las interfaces

RCI1: Diseño intuitivo

La interfaz debe ser fácil de entender y utilizar para los usuarios, con una disposición clara y coherente de los elementos visuales y de navegación.

La interfaz debe seguir los principios de diseño de usabilidad, como la visibilidad de las acciones disponibles, la consistencia en la ubicación de los controles y la retroalimentación adecuada al realizar acciones.

RCI2: Responsividad

La interfaz debe ser adaptable a diferentes dispositivos y tamaños de pantalla, como computadoras de escritorio, tablets y smartphones.

La interfaz debe responder y ajustarse automáticamente para proporcionar una experiencia de usuario óptima en cada dispositivo.

RCI3: Interacción intuitiva

La interfaz debe permitir a los usuarios interactuar de forma natural y familiar, utilizando elementos interactivos estándar y convencionales.

### 3.1.1 Interfaces de usuario

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

### 

### Sitio web, Escala de tiempo Descripción generada automáticamente

### 3.1.2 Interfaces de hardware

Dado que la aplicación será desarrollada como una aplicación web, no se requiere una interacción directa con interfaces de hardware específicas. Sin embargo, se espera que los usuarios accedan a la aplicación utilizando dispositivos como computadoras personales, laptops, tabletas o teléfonos inteligentes. La aplicación debe ser compatible con los estándares de hardware comunes para estos dispositivos, como pantalla, teclado, mouse/touchpad y conexión a Internet.

### 3.1.3 Interfaces de software

Navegadores web: La aplicación debe ser compatible con los navegadores web más comunes, como Google Chrome, Mozilla Firefox, Microsoft Edge, Safari, etc. Se deben seguir las mejores prácticas de desarrollo web y asegurarse de que la aplicación funcione correctamente en diferentes navegadores y versiones.

Sistema operativo: La aplicación debe ser compatible con diferentes sistemas operativos, como Windows, macOS, Linux, iOS y Android. Se deben seguir los estándares de desarrollo multiplataforma para garantizar el correcto funcionamiento en los sistemas operativos compatibles.

Servidor web: La aplicación se ejecutará en un servidor web que admita tecnologías web como HTML, CSS y JavaScript. Se pueden utilizar frameworks y tecnologías específicas Django.

**Base de datos: La aplicación requerirá una interfaz con la base de datos Oracle para almacenar y recuperar datos.**

### 3.1.4 Interfaces de comunicación

**La aplicación web requerirá interfaces de comunicación para interactuar con diferentes actores y componentes. A continuación, se describen las interfaces de comunicación relevantes:**

**Protocolo HTTP: La comunicación entre el cliente y el servidor web se realizará utilizando el protocolo HTTP (Hypertext Transfer Protocol). Esto permitirá la transferencia de datos entre el navegador web del cliente y el servidor que aloja la aplicación.**

**Correo electrónico: La aplicación enviará correos electrónicos para notificaciones, confirmaciones de compra, recibo de compra, entre otros. Se requerirá una interfaz de comunicación con un servidor de correo electrónico, utilizando protocolos estándar como SMTP**

**Integración de pasarelas de pago: Si la aplicación admite pagos en línea, será necesario establecer interfaces de comunicación con pasarelas de pago como PayPal o web Pay**

**Comunicación en tiempo real: Si se requiere comunicación en tiempo real, como el seguimiento del despacho**

**API de servicios externos: Si la aplicación se integra con servicios externos, como proveedores de logística para el despacho de productos, será necesario utilizar las API proporcionadas por esos servicios. Estas interfaces permitirán la comunicación con los servicios externos para obtener información actualizada sobre el estado del despacho.**

## 

## 3.2 Requisitos funcionales

**Registro de clientes:**

La aplicación debe permitir a los usuarios registrarse como clientes proporcionando información básica, como nombre, dirección, correo electrónico y contraseña.

Se deben realizar validaciones para garantizar la integridad de los datos ingresados durante el registro.

**Gestión de productos:**

La aplicación debe mostrar un catálogo de productos disponibles para la venta, incluyendo maceteros, tierra de hojas, flores y arbustos.

Los clientes registrados deben poder ver el stock real de los productos.

Los administradores deben tener la capacidad de agregar, editar y eliminar productos del catálogo.

**Venta en línea:**

Los clientes registrados deben poder seleccionar los productos deseados y agregarlos a una lista de compras.

La aplicación debe calcular el total de la venta, aplicar descuentos o promociones si corresponde, y descontar los productos del stock una vez creada la venta.

Se debe generar una confirmación de compra para que el cliente pueda revisar los detalles y realizar el pago.

**Monitoreo de compras:**

Los clientes registrados deben poder ver un historial de las compras realizadas, incluyendo detalles como la fecha, los productos adquiridos y el estado del despacho.

La aplicación debe proporcionar actualizaciones en tiempo real sobre el estado del despacho, incluyendo la fecha y hora en que se toma el pedido, se despacha y es recibido por el cliente.

**Suscripción a donaciones:**

Los clientes registrados deben tener la opción de suscribirse a una donación periódica a la fundación sin fines de lucro.

La información de suscripción debe enviarse directamente a la fundación.

Como incentivo, se otorgará un 5% de descuento en el total de la venta a todos los clientes suscritos.

**Descuentos y promociones:**

La aplicación debe permitir a los administradores crear y gestionar descuentos o promociones para productos específicos.

Los descuentos deben aplicarse automáticamente al total de la venta si corresponde.

**Mantenedores:**

La aplicación debe proporcionar funcionalidades para administrar la información relacionada con clientes, usuarios, productos, promociones o descuentos.

Los administradores deben poder agregar, editar y eliminar registros en estos mantenedores.

**Acceso basado en perfiles:**

La aplicación debe mostrar opciones y funcionalidades de acuerdo con los perfiles de cada usuario.

Los administradores deben tener acceso a funcionalidades adicionales, como la gestión de productos y mantenedores.

## 3.3 Requisitos no funcionales

### 3.3.1 Requisitos de rendimiento

Tiempo de respuesta:

La aplicación debe tener un tiempo de respuesta rápido para todas las acciones y operaciones. Se espera que la mayoría de las operaciones se completen en un tiempo máximo de 10 segundos

Capacidad de usuarios concurrentes:

La aplicación debe ser capaz de manejar simultáneamente un alto número de usuarios activos. Debe poder soportar al menos 2000 usuarios concurrentes sin degradar significativamente el rendimiento.

Tiempo de carga de páginas:

Las páginas de la aplicación deben cargarse rápidamente para minimizar la espera del usuario. Se debe priorizar la optimización del tiempo de carga, asegurándose de que las páginas se muestren en un tiempo máximo de 2 segundos

Rendimiento de la base de datos:

Las operaciones de lectura y escritura en la base de datos deben realizarse de manera eficiente para minimizar el tiempo de respuesta.

Se deben optimizar las consultas y utilizar índices apropiados para mejorar el rendimiento de las consultas relacionadas con la venta, el stock de productos y el historial de compras.

### 3.3.2 Seguridad

La aplicación debe ser segura y proteger los datos confidenciales de los usuarios, como contraseñas y detalles de pago.

Se deben implementar medidas de seguridad adecuadas, como cifrado de datos, protección contra ataques de fuerza bruta y prevención de vulnerabilidades conocidas.

### 3.3.3 Fiabilidad

Tolerancia a fallos:

La aplicación debe ser capaz de manejar errores y fallos de manera adecuada, minimizando su impacto en el funcionamiento general.

Se deben implementar mecanismos de recuperación y redundancia para garantizar la continuidad del servicio en caso de fallos.

Respaldos de datos:

Se debe realizar respaldo regularmente de la base de datos de la aplicación para asegurar la integridad y disponibilidad de los datos.

Los respaldos deben ser almacenados en un lugar seguro y se deben realizar pruebas periódicas de restauración para verificar su validez.

Manejo de errores:

La aplicación debe ser capaz de detectar y manejar errores de manera adecuada, proporcionando mensajes de error claros y acciones correctivas cuando sea necesario.

Se deben implementar mecanismos de registro de errores para facilitar la identificación y resolución de problemas.

Mantenimiento preventivo:

Se debe realizar un mantenimiento preventivo regular para garantizar el buen funcionamiento de la aplicación a largo plazo.

Esto incluye la aplicación de parches de seguridad, actualizaciones de software y optimizaciones de rendimiento.

### 3.3.4 Disponibilidad

La aplicación debe estar disponible y accesible para los usuarios en un alto porcentaje de tiempo. Se espera que la disponibilidad sea de al menos el 99% en un período de 24 horas, excluyendo períodos de mantenimiento programado

### 3.3.5 Mantenibilidad

Los administradores del sistema son responsables de realizar el mantenimiento preventivo, como aplicar actualizaciones de seguridad, monitorear el rendimiento del sistema y realizar copias de seguridad regulares de los datos. Para esto se realizarán pruebas a la página. El Mantenimiento se realizará 1 vez cada mes para asegurar la fiabilidad del sistema

### 3.3.6 Portabilidad

Se debe minimizar la dependencia de componentes específicos del servidor para facilitar la portabilidad. Esto implica que la lógica y funcionalidad del sistema estén principalmente implementadas en el cliente, reduciendo la carga en el servidor.

Porcentaje de código dependiente del servidor: Se debe minimizar la dependencia de componentes específicos del servidor para facilitar la portabilidad.

Uso de un determinado lenguaje por su portabilidad: Se debe utilizar un compilador o plataforma de desarrollo que permita generar ejecutables compatibles con múltiples sistemas operativos y arquitecturas.

Uso de un determinado compilador o plataforma de desarrollo: Se debe evitar la dependencia de características específicas de un sistema operativo en la aplicación. La aplicación debe ser compatible con diferentes sistemas operativos, como Windows, macOS o Linux.

## 3.4 Otros Requisitos

Cumplimiento normativo:

La aplicación web debe cumplir con las regulaciones y normativas legales vigentes, como la protección de datos personales y la privacidad de los usuarios.

Se deben implementar medidas de cumplimiento, como la obtención del consentimiento del usuario para el manejo de sus datos personales y la adopción de políticas de privacidad claras.

# 4. Propuesta de Planificación

## 4.1 Descripción general acerca de la Planificación

1. [Fase de análisis y documentación (1 mes):
   * Persona 1: Documentación de requisitos y diseño de la plataforma web.
2. Fase de desarrollo (2 meses):
   * Persona 2: Programación y desarrollo de la plataforma web.
3. Fase de pruebas y ajustes (1 mes):
   * Ambos miembros del equipo: Pruebas exhaustivas, corrección de errores y ajustes necesarios.

Consideraciones adicionales:

* Mantener una comunicación constante y coordinada entre los miembros del equipo.
* Establecer hitos y plazos intermedios para monitorear el progreso.
* Contar con un entorno de desarrollo adecuado.
* Considerar el uso de metodologías ágiles para una entrega incremental.

### 4.1.2 Definición del Equipo de Trabajo

1. Diego De La Vega: Encargada de Documentación y Diseño.
   * Responsabilidades:
     + Recolectar y analizar los requisitos del sistema.
     + Elaborar el Documento de Especificación de Requisitos de Software (ERS).
     + Realizar el diseño de la interfaz de usuario.
     + Crear diagramas de flujo y prototipos de la plataforma web.
2. Raul Valenzuela: Encargada de Programación.
   * Responsabilidades:
     + Desarrollar la plataforma web basándose en los requisitos y el diseño establecidos.
     + Aplicar buenas prácticas de desarrollo de software.
     + Escribir código limpio y eficiente.
     + Realizar pruebas unitarias y revisiones de código.

### 4.1.3 Definición de Actividades principales del Proyecto

1. Fase de Inicio:
   * Definición del alcance, objetivos y stakeholders del proyecto.
   * Elaboración del acta de inicio.
2. Fase de Planificación:
   * Desarrollo del plan de proyecto, incluyendo EDT, recursos y cronograma.
   * Estimación de tiempos, costos y riesgos.
3. Fase de Ejecución:
   * Implementación del plan y coordinación de tareas.
   * Gestión de recursos y comunicación efectiva.
4. Fase de Monitoreo y Control:
   * Seguimiento del avance y desempeño del proyecto.
   * Control de cambios y gestión de riesgos.
5. Fase de Cierre:
   * Evaluación del cumplimiento de objetivos.
   * Documentación de lecciones aprendidas y entrega de resultados.

## 4.2 Plan de Control de Cambio

| **Número de Cambio** | **Descripción del Cambio** | **Fecha de Solicitud** | **Estado del Cambio** | **Prioridad** | **Responsable** |
| --- | --- | --- | --- | --- | --- |
| 1 | Actualización de vista | 20/5/23 | Completo | Alta | Raúl Valenzuela |
| 2 | Actualización de documentos | 25/5/23 | Completo | Baja | Diego de la Vega |
| 3 | Portar la app en Django | 31/5/23 | En progreso | Alta | Raúl Valenzuela |
| 4 | Enlazar las páginas de la plataforma web | 24/6/23 | En progreso | Media | Diego De La Vega |

1. Actualización de vista: Se realizaron modificaciones en la interfaz de usuario para mejorar la experiencia del usuario, incluyendo cambios en la disposición de elementos, colores y estilos visuales.
2. Actualización de documentos: Se llevaron a cabo modificaciones en los documentos del proyecto, como el Documento de Especificación de Requisitos y el Plan de Proyecto, para reflejar los cambios y ajustes realizados en el alcance y en la planificación del proyecto.
3. Portar la app en Django: Se está trabajando en la migración de la aplicación existente a la plataforma de desarrollo Django. Esto implica adaptar el código y las funcionalidades existentes para que sean compatibles con el entorno Django, aprovechando sus ventajas y características.
4. Enlazar las páginas de la plataforma web: Actualmente se está trabajando en la interconexión y navegabilidad entre las distintas páginas de la plataforma web. Esto implica establecer enlaces adecuados para permitir una transición fluida entre las diferentes secciones y funcionalidades del sitio.