| DUOC UC - ESCUELA DE INFORMÁTICA Y TELECOMUNICACIONES |
| --- |
| Especificación de Requisitos de Software |
| *Proyecto: Tienda Duoc Fragancias* |
|  |
| **Revisión*: [00.01]*** |
| **Integrantes: Mayckol Mardones, Rodrigo Vargas, Francisco Vera** |

| Especificación de Requisitos según estándar de IEEE 830. |
| --- |

**Contenido**

[**Ficha del documento 3**](#_tj7f1s25lmsz)

[**1. Introducción 4**](#_hre5h7l31v4h)

[1.1.](#_fnzfmprrjkwp) Propósito 4

[1.2.](#_msbuc4p7nrvt) Ámbito del Sistema 4

[1.3.](#_ymhxa0k2txdt) Definiciones, Acrónimos y Abreviaturas 4

[1.4.](#_qwwsb37uk8al) Referencias 4

[1.5.](#_kklvc6k4r34i) Visión General del Documento 4

[**2.**](#_18l3hfflsks2) **Descripción General 5**

[2.1.](#_7ks0vn2ztvlv) Perspectiva del Producto 5

[2.2.](#_k3o9lu1gquq8) Funciones del Producto 5

[2.3.](#_aairfogv4hi7) Características de los Usuarios 5

[2.4.](#_ppbf2ki6y192) Restricciones 5

[2.5.](#_wrzemkgbzwx3) Suposiciones y Dependencias 6

[2.6.](#_gq0hejegqyb6) Requisitos Futuros 6

[**3.**](#_f1nt5ped0v3c) **Requisitos Específicos 7**

[3.1](#_rbc9074iugjn) Requisitos comunes de las interfaces 8

[*3.1.1*](#_whw69zqwck06) *Interfaces de usuario 8*

[*3.1.2*](#_zbbr3a6bgrtn) *Interfaces de hardware 8*

[*3.1.3*](#_g9me1ngoohbz) *Interfaces de software 8*

[*3.1.4*](#_sfiuoybysbqb) *Interfaces de comunicación 8*

[3.2](#_v5920c99bgyo) Requisitos funcionales 9

[3.3](#_6ne8u3v8umi4) Requisitos no funcionales 9

[*3.3.1*](#_jk7lt9gao2c8) *Requisitos de rendimiento 9*

[*3.3.2*](#_m1rf3w3r677x) *Seguridad 10*

[*3.3.3*](#_sdctroqjs5o5) *Fiabilidad 10*

[*3.3.4*](#_x7gbl0gwdu1q) *Disponibilidad 10*

[*3.3.5*](#_4m381ihczlzz) *Mantenibilidad 10*

[*3.3.6*](#_dznsiqxf0ssq) *Portabilidad 10*

[3.4](#_jf91qc94r3nt) Otros Requisitos 10

# Ficha del documento

| **Fecha** | **Revisión** | **Autor** | **Modificación** |
| --- | --- | --- | --- |
| *13-09-2025* | *01* | *Rodrigo Vargas* | *Apartados 1 y 2* |
|  |  |  |  |

Documento validado por las partes en fecha:

| Por el cliente |  | Por la empresa suministradora |
| --- | --- | --- |
| [Firma] |  | [Firma] |
| Sr./Sra. |  | Sr./Sra. |

# 1. Introducción

En esta sección se proporcionará una introducción a todo el documento de Especificación de Requisitos Software (ERS). Consta de varias subsecciones: propósito, ámbito del sistema, definiciones, referencias y visión general del documento.

## 1.1. Propósito

El propósito de este documento es definir los requisitos, tanto funcionales como no funcionales de una página web que permitirá a los usuarios comprar perfumes en una tienda online dedicada exclusivamente para ellos. El presente documento va dirigido a todo el personal involucrado en la creación, desarrollo, implementación y mantenimiento del proyecto, con el fin de entregar una guía útil para la realización del mismo.

## 1.2. Ámbito del Sistema

El sistema llevará por nombre **Duoc Fragancias**

**Qué hará el sistema:**

* Permitirá a todos los usuarios de Duoc Fragancias realizar compras mediante la plataforma web online proveída.
* Permitirá a todos los usuarios generar un carrito de compras con todos los productos de su elección, entregando distintas opciones para agregar, modificar y eliminar dichos productos según sus necesidades.

**Qué NO hará el sistema:**

* En la primera versión, el sistema no incluirá un sistema de rastreo de envíos para cada compra realizada.
* En la primera versión, el sistema no incluirá un botón para exportar las boletas a formatos comunes.
* En la primera versión, el sistema será únicamente online, por lo que no aceptará pagos en ningún formato que no sea digital.

## 1.3. Definiciones, Acrónimos y Abreviaturas

ERS: Especificación de Requerimientos de Software.

## 1.4. Referencias

No se han utilizado documentos adicionales como referencia para la realización del presente.

## 1.5. Visión General del Documento

En la primera sección del documento se proporciona una introducción, donde presentamos la intención del documento, su público objetivo y el propósito del proyecto en su forma implementada.

En los siguientes puntos se detallarán algunos factores que afectan al producto, la importancia que tiene este software para los distintos actores que harán uso del mismo, considerando sus necesidades y otras características relevantes. En la siguiente instancia, se incluirán sus requisitos, tanto funcionales como no funcionales, requisitos específicos, entre otros.

# 2. Descripción General

Algunos de los factores por los que se ve afectado el producto son los siguientes:

• Alta demanda del mercado objetivo.

• Cumplimiento de normas y regulaciones ambientales.

• Estructura tecnológica y escalabilidad futura.

• Rapidez y/o eficiencia en los procesos a realizar dentro de la aplicación.

## 2.1. Perspectiva del Producto

El sistema propuesto puede ser relacionado a otros ya existentes en el mercado, aunque en este caso en específico se ha desarrollado como iniciativa para aprender sobre e-commerce y ser líderes innovadores en nuestra institución y carrera.

## 2.2. Funciones del Producto

En su primera versión, este sistema plantea las siguientes funcionalidades para los usuarios **Clientes**:

* Visita y navegación por las distintas páginas que conforman la plataforma web de Duoc Fragancias.
* Creación de carrito de compras para previsualizar tanto presupuestos como compras en sí.

En su primera versión, este sistema plantea las siguientes funcionalidades para los usuarios **Administradores**:

* Visita y navegación por las distintas secciones de las páginas de administración que conforman la plataforma web de Duoc Fragancias.
* Creación/Edición/Eliminación de nuevos usuarios con sus distintos roles.
* Creación/Edición/Eliminación de nuevos productos con sus respectivos atributos.

## 

## 2.3. Características de los Usuarios

Existirán 2 tipos de perfiles de usuarios, en este caso corresponden a los siguientes:

1. **Usuario Administrador:** Debe poseer conocimientos en utilización de computadores a nivel intermedio, ya que el uso de sus respectivas funciones en el aspecto laboral no suponen tareas de mayor complejidad.
2. **Usuario Cliente:** Debe poseer conocimientos básicos en utilización de computadores e internet. El sistema está adecuado a un nivel básico de comprensión lectora y matemática simple en aspectos de compra/venta y la disposición de la información.

## 2.4. Restricciones

* Durante las primeras 2 entregas, este proyecto no será desplegable en otras plataformas que no sean tipo Github Pages, dado que consta meramente de FrontEnd
* El sistema actual no se integra con otros que sean complejos, únicamente consume datos de una interfaz local, la cual corresponde al archivo [regiones-comunas.js](http://regiones-comunas.js).
* Al utilizar LocalStorage para almacenar datos, no se permiten operaciones en paralelo al momento de utilizar la primera versión de la plataforma, es por ello que la información tanto de los productos como de los usuarios estará sujeta al navegador en uso en ese momento.
* Por la misma razón que el punto previo, todo lo sujeto a auditorías queda fuera de discusión, dado que no existe una forma de realizar logs o alguna forma de bitácoras en un servidor.
* El desarrollo del proyecto está restringido al siguiente conjunto de tecnologías:
  + HTML5
  + BootStrap
  + CSS
  + JavaScript

## 2.5. Suposiciones y Dependencias

Despliegue en plataformas: Durante la primera y segunda versión del proyecto, se asume que el sistema será desplegado en Github Pages, lo que implica tener un sistema sin Backend. Esto debería cambiar para el momento en que se desarrolle la versión 3 del sistema.

APIs en el navegador de los clientes: Al depender de LocalStorage para almacenar los datos de productos y usuarios, se asume que los usuarios clientes dispondrán de un navegador compatible con esta funcionalidad y no borrarán los datos de navegación.

## 

## 2.6. Requisitos Futuros

Dentro de los requisitos futuros, y como ya se ha mencionado en puntos anteriores del documento, se implementará BackEnd para la gestión de los datos, tanto de productos como de usuarios, con el fin de poder realizar operaciones en paralelo, así como cubrir aspectos de usabilidad, mantenibilidad, y escalabilidad. El fin de este requisito es gestionar de mejor manera los datos en una plataforma dedicada para ello, resultando en un manejo de inventario óptimo. Adicionalmente, esto contribuirá a un mejor sistema de login, con almacenamiento de credenciales seguras y sesiones de usuario. Como último requisito futuro debemos mencionar la integración de un panel de control para los clientes, donde se les permita ver su historial de compras, editar su información personal e incorporar un apartado de seguimiento de compra.

3. Requisitos Específicos

Esta sección contiene los requisitos a un nivel de detalle suficiente como para permitir a los diseñadores diseñar un sistema que satisfaga estos requisitos, y que permita al equipo de pruebas planificar y realizar las pruebas que demuestren si el sistema satisface, o no, los requisitos. Todo requisito aquí especificado describirá comportamientos externos del sistema, perceptibles por parte de los usuarios, operadores y otros sistemas. Esta es la sección más larga e importante de la ERS. Deberán aplicarse los siguientes principios:

• El documento debería ser perfectamente legible por personas de muy distintas formaciones e intereses.

• Deberán referenciarse aquellos documentos relevantes que poseen alguna influencia sobre los requisitos.

• Todo requisito deberá ser unívocamente identificable mediante algún código o sistema de numeración adecuado.

• Lo ideal, aunque en la práctica no siempre realizable, es que los requisitos posean las siguientes características:

* **Corrección**: La ERS es correcta si y sólo si todo requisito que figura aquí (y que será implementado en el sistema) refleja alguna necesidad real. La corrección de la ERS implica que el sistema implementado será el sistema deseado.
* **No ambiguos**: Cada requisito tiene una sola interpretación. Para eliminar la ambigüedad inherente a los requisitos expresados en lenguaje natural, se deberán utilizar gráficos o notaciones formales. En el caso de utilizar términos que, habitualmente, poseen más de una interpretación, se definirán con precisión en el glosario.
* **Completos**: Todos los requisitos relevantes han sido incluidos en la ERS. Conviene incluir todas las posibles respuestas del sistema a los datos de entrada, tanto válidos como no válidos.
* **Consistentes**: Los requisitos no pueden ser contradictorios. Un conjunto de requisitos contradictorios no es implementable.
* **Clasificados**: Normalmente, no todos los requisitos son igual de importantes. Los requisitos pueden clasificarse por importancia (esenciales, condicionales u opcionales) o por estabilidad (cambios que se espera que afecten al requisito). Esto sirve, ante todo, para no emplear excesivos recursos en implementar requisitos no esenciales.
* **Verificables**: La ERS es verificable si y sólo si todos sus requisitos son verificables. Un requisito es verificable (testeable) si existe un proceso finito y no costoso para demostrar que el sistema cumple con el requisito. Un requisito ambiguo no es, en general, verificable. Expresiones como a veces, bien, adecuado, etc. Introducen ambigüedad en los requisitos. Requisitos como “en caso de accidente la nube tóxica no se extenderá más allá de 25Km" no es verificable por el alto costo que conlleva.
* **Modificables**: La ERS es modificable si y sólo si se encuentra estructurada de forma que los cambios a los requisitos pueden realizarse de forma fácil, completa y consistente. La utilización de herramientas automáticas de gestión de requisitos facilitan enormemente esta tarea.
* **Trazables**: La ERS es trazable si se conoce el origen de cada requisito y se facilita la referencia de cada requisito a los componentes del diseño y de la implementación. La trazabilidad hacia atrás indica el origen (documento, persona, etc.) de cada requisito. La trazabilidad hacia delante de un requisito R indica que componentes del sistema son los que realizan el requisito R.

## 3.1 Requisitos comunes de las interfaces

### 3.1.1 Interfaces de usuario

* UI-001: Página HTML principal con menú simple
* UI-002: Página de productos con lista básica
* UI-003: Página de carrito simple
* UI-004: Página de contacto con formulario
* UI-005: Diseño que funcione en computadora

### 3.1.2 Interfaces de hardware

* HW-001: Funcione en navegador web normal
* HW-002: No necesita touch, solo mouse

### 3.1.3 Interfaces de software

* SW-001: Solo archivos HTML/CSS/JS
* SW-002: No necesita conexiones externas

### 3.1.4 Interfaces de comunicación

* COM-001: Solo localhost, no necesita internet
* COM-002: No necesita seguridad HTTPS

## 3.2 Requisitos funcionales

En la siguiente sección se detallarán los principales requisitos funcionales a realizar durante esta primera versión del proyecto:

Sección de clientes

* **RF-01: Gestión de usuarios:** El cliente debe poder registrarse en la página web ingresando sus datos (rut, nombre, apellidos, correo, región, comuna, etc.).
* **RF-02: Gestión de carrito de compra:** El cliente debe poder agregar productos a su carrito de compras desde la página de detalles del producto. Dentro del carrito, podrá ver un resumen indicando el total de productos que contiene, junto a su precio total e individual por cada tipo distinto de producto. También se le permitirá editar y eliminar productos del mismo.

Sección de administración

* **RF-01: Gestión de productos:** El administrador debe tener acceso a un panel de control que le permita interactuar con los distintos productos creados, permitiendo agregar nuevos productos y editar características de los ya existentes, así como modificar la cantidad de inventario por cada producto.

## 3.3 Requisitos no funcionales

### 3.3.1 Requisitos de rendimiento

En la siguiente sección se detallan algunos de los requisitos de rendimiento abarcados en esta primera versión del proyecto.

* La carga de las páginas no debe exceder un máximo de 3 segundos al navegar.
* El sistema deberá ser capaz de soportar usuarios en simultáneo navegando por la página sin que esta se caiga.

### 3.3.2 Seguridad

### 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.