**DUOC UC - Escuela de informática y telecomunicaciones**

Especificación de Requisitos del Software

Proyecto Huerto-Hogar

**Revisión*: [02]***

Integrantes: Aaron Gorigoitia

Agustín Aceval

Franco Alarcón

Docente: Josue Nahum Oteiza Soto

Sección: 001D

Fecha: 27-10-2025

**Integrantes:**

| **Nombre Integrante del Equipo** |
| --- |
| *Franco Alarcon* |
| *Aaron Gorigoitia* |
| *Agustin Aceval* |

**1.1 Propósito**

El propósito de este sistema es ofrecer una plataforma web que conecte a las personas con productos naturales y orgánicos de forma simple y directa. Los usuarios podrán explorar el catálogo, registrar sus datos, realizar compras y llevar un control de lo que adquieren, todo desde la misma tienda online.

Además, el sistema incluirá herramientas para que los administradores gestionen productos y perfiles de clientes, manteniendo la información organizada y el funcionamiento eficiente.

**1.2 Descripción general del proyecto**

El proyecto consiste en el desarrollo de una plataforma web de comercio electrónico que conecta directamente a los productores de alimentos orgánicos y naturales con los consumidores finales en Chile.

La tienda online busca ofrecer a los clientes una experiencia de compra sencilla, rápida y segura, permitiéndoles explorar un catálogo de productos frescos, conocer información de la empresa y contactar directamente a la tienda.

**El sistema se compone de dos perfiles principales:**

**Cliente:** podrá registrarse, iniciar sesión, navegar por el catálogo de productos, añadir artículos al carrito, consultar información de la empresa, ver un mapa de tiendas y enviar mensajes a través de un formulario de contacto.

**Administrador:** contará con un panel de administración que le permitirá gestionar los productos (crear, modificar, eliminar) y los usuarios registrados en la plataforma.

El desarrollo se realizará utilizando React como framework principal de JavaScript, implementando componentes reutilizables con gestión de estados y propiedades. Se mantendrá el diseño responsivo con Bootstrap y se implementarán pruebas unitarias exhaustivas para garantizar la calidad del código.

**1.3 Alcance**

Nuestra plataforma web contará con dos módulos principales:

**Módulo Público**: accesible para los clientes para poder explorar los productos, registrarse, iniciar sesión.

**Módulo Administrativo:** El sistema está enfocado en un sistema escalable para que en futuras entregas tenga otros sistemas como métodos de pago.

**2. Requerimientos Del Proyecto**

**2.1 Requerimientos funcionales**

**RF-01**: Registro y autenticación de usuarios.

**RF-02**: Visualización de catálogo de productos orgánicos y frescos.

**RF-03**: Visualización de detalle de producto.

**RF-04:** Añadir productos al carrito de compras.

**RF-05**: Validación de stock antes de confirmar el pedido.

**RF-06**: Generación de boleta digital al finalizar la compra.

**RF-07**: Envío de consultas mediante formulario de contacto.

**RF-08**: Presentación de información institucional (misión, visión, mapa de sucursales).

**RF-09**: Blog con artículos sobre vida saludable y alimentación orgánica.

**RF-10**: Panel de administración para gestionar productos y usuarios.

**2.2 Requerimientos no funcionales**

**RNF-01**: Desarrollo con React, HTML5, CSS3, Bootstrap y JavaScript.

**RNF-02**: Diseño responsivo compatible con computadores, tablets y móviles.

**RNF-03**: Interfaz intuitiva y de fácil navegación.

**RNF-04**: Tiempo de carga inferior a 3 segundos por página.

**RNF-05**: Uso de componentes React reutilizables para modularidad.

**RNF-06**:Validación de datos en formularios (correo, contraseñas, RUN, códigos promocionales).

**RNF-07**: Compatibilidad con navegadores Chrome, Firefox y Edge.

**RNF-08**: Implementación de pruebas unitarias con Jasmine y Karma.

**RNF-09:** Cobertura mínima del 80% en pruebas de componentes críticos.

**2.3 Restricciones**

Las restricciones son limitaciones técnicas, operativas o de negocio que afectan el desarrollo del sistema:

**Tecnológicas:**

* El frontend debe desarrollarse con React, HTML5, CSS3 y JavaScript.
* El proyecto debe versionarse en repositorio Github.
* Debe implementarse pruebas unitarias con Jasmine y Karma.

**Operativas:**

El sistema de descuentos y promociones debe estar validado con los datos ingresados por el usuario (edad, correo, código promocional).

**De alcance:**

* La primera versión se centrará en catálogo, carrito y pedidos, con proyección a integraciones futuras (métodos de pago, base de datos y backend).

**3. Herramientas y Tecnologías**

**3.1 Lenguajes de programación**

* HTML: Para la creación de la estructura semántica de las páginas web.
* CSS: Para el diseño visual, aplicaciones de estilos personalizados y adaptación responsive.
* JavaScript: Para la validación de formularios, interactividad y manipulación dinámica del contenido.

**3.2 Frameworks y librerías**

* React: Para el desarrollo de componentes frontend reutilizables con gestión de estados y propiedades.
* Bootstrap: Para agilizar diseño responsivo.
* Jasmine: Framework para pruebas unitarias de componentes JavaScript.
* Karma: Test runner para ejecutar pruebas en múltiples navegadores.

**3.3 Plataformas y herramientas adicionales**

* Git y Github: Control de versiones y repositorio remoto colaborativo.
* Visual Studio Code: Entorno de desarrollo principal.
* Google Chrome: Navegador para prueba y depuración.
* Herramientas de validación HTML/CSS: Para asegurar el cumplimiento de estándares web.

**4. Propuestas del Proyecto**

**4.1 Enfoque general**

El proyecto se abordará bajo un enfoque incremental e iterativo, priorizando primero las funcionalidades esenciales:

* Catálogo de productos desarrollado con componentes React.
* Carrito de compras con gestión de estado.
* Gestión de pedidos con validaciones.

Se prioriza la usabilidad, accesibilidad y seguridad del sistema, asegurando una experiencia de compra sencilla, moderna y confiable para los clientes. Se implementará un proceso de testing robusto con pruebas unitarias para garantizar la calidad del código.

**4.2 Planificación y cronograma**

* Fase 1: Diseño de estructura y navegación web.
* Fase 2: Implementación de catálogo y carrito con gestión de estado.
* Fase 3: Validaciones en formularios y flujo de pedidos.
* Fase 4: Implementación de pruebas unitarias con Jasmine y Karma.
* Fase 5: Módulo administrativo y blog.
* Fase 6: Pruebas finales y despliegue

**4.3 Estrategias de implementación**

* **Modularidad:** Separación clara de componentes frontend.
* **Testing:** Pruebas unitarias y de integración continua.
* **Responsividad:** Diseño adaptable a todo tipo de dispositivo.
* **Control de versiones:** Uso constante de commits y ramas en GitHub.

**4.4 Estrategia de Testing**

La estrategia de testing se centrará en:

* Configuración del entorno de pruebas con Jasmine y Karma.
* Creación de al menos 10 pruebas unitarias para los componentes críticos.
* Uso de mocks y stubs para dependencias.
* Análisis de cobertura de código.
* Validación de diseño responsivo en distintos dispositivos.

**4.5 Validación y manejo de riesgos**

* Pruebas manuales y automáticas en cada iteración.
* Registro de errores y soluciones.
* Revisión continua de objetivos y prioridades del proyecto.

**5. Conclusiones**

El proyecto HuertoHogar representa un avance significativo hacia la digitalización del comercio de productos orgánicos y naturales en Chile. Su desarrollo permite fortalecer el vínculo entre productores y consumidores mediante una plataforma web moderna, accesible y confiable.

La implementación con React, Bootstrap y JavaScript garantiza un sistema escalable, modular y de fácil mantenimiento. Gracias al uso de componentes reutilizables, la aplicación mantiene una estructura ordenada y adaptable a futuras ampliaciones. Además, las pruebas unitarias aseguran la calidad del código y la estabilidad de las principales funcionalidades, minimizando errores en etapas posteriores.

El enfoque incremental e iterativo adoptado favorece una entrega continua de valor, facilitando la validación temprana de requerimientos y la mejora progresiva de la experiencia del usuario. En conjunto, el sistema sienta las bases para una tienda online robusta, eficiente y alineada con las tendencias tecnológicas actuales.

**5.1 Resumen de las expectativas para el proyecto y próximos pasos.**

* **Implementación de una base de datos centralizada:** Permitir la persistencia de información de usuarios, productos y pedidos en la nube.
* **Desarrollo del backend:** Crear un servidor que gestione peticiones, autenticación y comunicación con la base de datos mediante APIs RESTful o GraphQL.
* **Integración de métodos de pago seguros:** Incorporar plataformas como WebPay o MercadoPago, garantizando transacciones seguras y confiables.
* **Optimización de rendimiento y seguridad:** Aplicar buenas prácticas en cacheo, encriptación y protección de datos personales (RGPD/Ley 19.628).
* **Despliegue y monitoreo continuo:** Publicar la aplicación en un entorno productivo (por ejemplo, Vercel o Netlify) y establecer herramientas de seguimiento de errores y métricas de uso.
* **Expansión funcional:** Agregar secciones dinámicas como blog, reseñas de usuarios y promociones automatizadas basadas en comportamiento de compra.