# **ERS – Proyecto *Level-Up Gamer***

**Asignatura:** FullStack\_2

**sección:** 003D   
 **Evaluación:** Parcial 2 – Forma B  
 **Proyecto:** Tienda Online *Level-Up Gamer*

**Integrantes del equipo:**

Benjamín Mella

Bryan Saavedra

**Docente:** Josué Nahum Oteiza Soto

## **1. Introducción**

### **1.1 Propósito**

El presente documento describe la **Especificación de Requisitos del Software (ERS)** del proyecto **Level-Up Gamer**, junto con la **propuesta personalizada** desarrollada para la Evaluación Parcial N°2.  
El sistema corresponde a una tienda online dirigida al público gamer, desarrollada con **React**, **Bootstrap** y **JavaScript**, e incluye pruebas unitarias para garantizar la funcionalidad de los componentes frontend mediante **Jasmine y Karma**.

**2. objetivos**

* Modernizar el frontend integrando un framework JavaScript (React).
* Desarrollar componentes reutilizables con gestión de estado y propiedades.
* Asegurar un diseño responsivo con Bootstrap.
* Implementar pruebas unitarias que validen la lógica y comportamiento de los componentes.

## **3. Alcance**

El sistema permite:

* Registro e inicio de sesión de usuarios.
* Navegación por catálogo de productos.
* Visualización del detalle de cada producto.
* Gestión de stock.
* Interacción mediante interfaz moderna y responsiva.

## **4. Requisitos Funcionales y No Funcionales**

| **ID** | **Tipo** | **Descripción** |
| --- | --- | --- |
| RF-01 | Funcional | Registro de usuario con validación de correo y contraseña. |
| RF-02 | Funcional | Inicio y cierre de sesión. |
| RF-03 | Funcional | Catálogo de productos dinámico y filtrable. |
| RF-04 | Funcional | Visualización de detalle y stock de producto. |
| RF-05 | Funcional | Carrito de compras con actualización de total. |
| RNF-01 | No funcional | Diseño responsivo implementado con Bootstrap. |
| RNF-02 | No funcional | Tiempo de carga inferior a 3 segundos. |
| RNF-03 | No funcional | Código modular y mantenible con React Components. |

## **5. Reglas de Negocio**

* El correo debe contener “@” y dominio válido.
* Contraseña entre 4 y 12 caracteres.
* El stock y precios no pueden ser negativos.
* Los formularios deben validar todos los campos antes de enviar.

## 

## 6. **Propuesta Personalizada**

El proyecto se modernizó usando **React** como framework principal.

* Se crearon componentes **Header**, **Footer**, **Producto**, **Carrito**, y **LoginForm**.
* Cada componente gestiona su propio **estado (state)** y recibe **props** desde un componente principal App.js.
* Se aplicó **Bootstrap 5** para lograr un diseño responsivo, adaptando el layout a pantallas de escritorio.
* El routing se gestionó mediante **React Router DOM**, permitiendo navegación fluida entre vistas (/home, /productos, /login).

## 7. **Herramientas utilizadas**

|  |  |
| --- | --- |
| **React** | : Framework JavaScript principal para el desarrollo del frontend. |

|  |  |
| --- | --- |
| **Bootstrap 5** | : Framework CSS utilizado para lograr diseño responsivo. |

|  |  |
| --- | --- |
| **Node.js / npm** | : Entorno de ejecución y gestor de dependencias del proyecto. |

|  |  |
| --- | --- |
| **Git y GitHub** | : Control de versiones y repositorio colaborativo. |

|  |  |
| --- | --- |
| **AWS** | : Plataforma utilizada para desplegar y alojar la aplicación web. |

|  |  |
| --- | --- |
| **Visual Studio Code** | : Entorno de desarrollo utilizado por el equipo. |

|  |  |
| --- | --- |
| **Jasmine y Karma** | : Herramientas de testing unitario para los componentes frontend. |

## **8. Documento de Cobertura de Testing**

## Las pruebas unitarias se realizaron sobre la aplicación **Level-Up Gamer**, desarrollada con el framework **React**, desplegada en un servidor **AWS EC2** El objetivo fue validar el comportamiento, la lógica y la correcta renderización de los componentes frontend, asegurando su estabilidad en un entorno de producción real.

Entorno de pruebas:

|  |  |
| --- | --- |
| **Framework frontend** | : React (Vite + Node.js) |

|  |  |
| --- | --- |
| **Framework CSS** | : Bootstrap (para diseño responsivo) |

|  |  |
| --- | --- |
| **Framework de testing** | : Jasmine + Karma |

|  |  |
| --- | --- |
| **Servidor** | : AWS EC2 |

|  |  |
| --- | --- |
| **Ruta del proyecto** | /level-up/level-up-React/level-up |

|  |  |
| --- | --- |
| **Comando de ejecución** | : npm run test:karma |

Evidencia

Interfaz de usuario gráfica, Texto

El contenido generado por IA puede ser incorrecto.

## 9: Propuesta personalizada

La actualización de la tienda online **Level-Up Gamer** incluyó no solo la migración del frontend a **React**, sino también su **implementación en la nube mediante AWS**, asegurando mayor disponibilidad y escalabilidad.

**Integración AWS:**

* Se creó una **instancia EC2** con sistema operativo Ubuntu para alojar la aplicación React.
* Se instaló **Node.js**, **npm** y las dependencias del proyecto para ejecutar el build de producción.
* Se configuró el servidor con **Nginx** como proxy inverso para servir el contenido estático del build (/dist).
* El acceso público se habilitó mediante la dirección IPv4 pública de la instancia.
* Se utilizó **AWS Security Groups** para permitir tráfico HTTP/HTTPS.
* Los archivos de la aplicación también pueden alojarse en un **bucket S3** con permisos públicos y CDN mediante **CloudFront** (alternativa opcional).

**Ventajas de usar AWS:**

* Escalabilidad automática para soportar tráfico variable.
* Disponibilidad 24/7 sin depender del equipo local.

## 10: Conclusión

El desarrollo del proyecto **Level-Up Gamer** permitió aplicar de manera integral los conocimientos adquiridos en **Desarrollo Fullstack II**, mediante la implementación de un **frontend moderno con React**, complementado con **Bootstrap** para lograr un diseño totalmente **responsivo y adaptable**.

Durante esta fase, el equipo consolidó una propuesta personalizada orientada a la escalabilidad y calidad del software, integrando pruebas unitarias con **Jasmine y Karma** para validar el correcto funcionamiento de los componentes, asegurando su estabilidad y confiabilidad.

Además, se incorporó la **plataforma AWS** para el despliegue de la aplicación, garantizando mayor **disponibilidad**, **rendimiento** y **profesionalismo** en el entorno de ejecución, simulando un escenario real de implementación en la nube.

En conjunto, el proyecto cumple con los requerimientos establecidos por la pauta de evaluación, demostrando un desarrollo técnico completo, organizado y alineado con las buenas prácticas del desarrollo frontend moderno.