# **ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE (ERS)**

## **Sistema de Subastas Online**

Versión: 1.0  
Fecha: 08 de Enero de 2025  
Equipo de Desarrollo: Diony Peguero y Benjamin  
Asignatura: DSY1104 - Desarrollo Fullstack II

## **1. INTRODUCCIÓN**

### **1.1 Propósito del Documento**

Este documento establece los requisitos de software para el desarrollo de una plataforma de subastas online que permite a los usuarios participar en subastas en tiempo real, gestionar sus pujas y realizar seguimiento de envíos de productos ganados.

### **1.2 Alcance del Sistema**

El sistema "Subastas Online" es una aplicación web que facilita la participación en subastas virtuales, proporcionando una experiencia interactiva y segura para los usuarios. La plataforma incluye gestión de usuarios, sistema de pujas, seguimiento de envíos y administración de presupuestos.

## **2. DESCRIPCIÓN GENERAL DEL SISTEMA**

### **2.1 Perspectiva del Producto**

La plataforma de Subastas Online es un sistema web independiente que opera como una tienda online especializada en subastas en tiempo real. El sistema se enfoca en proporcionar una experiencia de usuario fluida y moderna utilizando tecnologías web actuales.

### **2.2 Características Principales**

* Autenticación de usuarios con registro y login seguro
* Sistema de subastas en tiempo real con pujas automáticas y manuales
* Gestión de presupuesto con sistema de recarga
* Seguimiento de envíos con códigos de rastreo
* Interfaz responsive compatible con dispositivos móviles y desktop
* Sistema de notificaciones para informar estado de pujas y envíos

### **2.3 Usuarios del Sistema**

* Usuarios registrados: Participantes en subastas con capacidad de pujar y gestionar envíos
* Sistema automatizado: Bots que simulan competencia en las subastas

## **3. REQUERIMIENTOS ESPECÍFICOS**

### **3.1 Requerimientos Funcionales**

#### **RF001 - Gestión de Usuarios**

* Descripción: El sistema debe permitir registro y autenticación de usuarios
* Prioridad: Alta
* Criterios de Aceptación:
  + Registro con nombre, email y contraseña
  + Validación de email único en el sistema
  + Login con credenciales válidas
  + Sesión persistente durante navegación
  + Cierre de sesión seguro

#### **RF002 - Sistema de Subastas**

* Descripción: Gestión completa del ciclo de vida de subastas
* Prioridad: Alta
* Criterios de Aceptación:
  + Visualización de subastas activas
  + Participación en pujas con presupuesto disponible
  + Temporizador en tiempo real
  + Sistema de pujas automáticas (bots)
  + Notificación de resultados

#### **RF003 - Gestión de Presupuesto**

* Descripción: Control de fondos disponibles para pujas
* Prioridad: Media
* Criterios de Aceptación:
  + Presupuesto inicial de $5,000 CLP
  + Recarga de presupuesto hasta $100,000 CLP
  + Descuento automático por pujas realizadas
  + Validación de fondos suficientes

#### **RF004 - Sistema de Envíos**

* Descripción: Seguimiento y gestión de productos ganados
* Prioridad: Media
* Criterios de Aceptación:
  + Configuración de datos de envío
  + Generación de códigos de seguimiento
  + Timeline de estados de envío
  + Búsqueda por código de seguimiento

#### **RF005 - Interfaz de Usuario**

* Descripción: Experiencia de usuario intuitiva y responsive
* Prioridad: Alta
* Criterios de Aceptación:
  + Diseño responsive para móviles y desktop
  + Navegación intuitiva entre secciones
  + Feedback visual en tiempo real
  + Accesibilidad web básica

### **3.2 Requerimientos No Funcionales**

#### **RNF001 - Rendimiento**

* El sistema debe cargar páginas en menos de 3 segundos
* Actualización de pujas en tiempo real (máximo 1 segundo de delay)
* Soporte para al menos 10 usuarios concurrentes

#### **RNF002 - Usabilidad**

* Interfaz intuitiva que no requiera entrenamiento previo
* Compatibilidad con navegadores Chrome, Firefox, Safari y Edge
* Diseño responsive para dispositivos móviles

#### **RNF003 - Confiabilidad**

* Disponibilidad del sistema del 95% durante horarios de operación
* Persistencia de datos en localStorage para sesiones
* Manejo graceful de errores con mensajes informativos

#### **RNF004 - Seguridad**

* Validación de datos en cliente y servidor
* Protección contra inyección de código
* Autenticación segura de usuarios

## **4. HERRAMIENTAS Y TECNOLOGÍAS**

### **4.1 Tecnologías Frontend**

#### **4.1.1 Lenguajes de Programación**

* HTML5: Estructura semántica de páginas web
* CSS3: Estilos y diseño visual responsive
* JavaScript ES6+: Lógica de negocio del lado cliente y validaciones

#### **4.1.2 Frameworks y Librerías**

* Bootstrap 5.3.8: Framework CSS para diseño responsive
* Font Awesome 6.0: Librería de iconos vectoriales
* Bootstrap JavaScript: Componentes interactivos (modales, tooltips, etc.)

#### **4.1.3 Herramientas de Desarrollo**

* Visual Studio Code: Editor de código principal
* Chrome DevTools: Depuración y testing
* Git: Control de versiones
* GitHub: Repositorio remoto y colaboración

### **4.2 Herramientas de Diseño y Documentación**

* Visual Paradigm: Diagramas
* Microsoft Word: Documentación técnica
* Draw.io: Diagramas de flujo y arquitectura

## **5. PROPUESTAS DEL PROYECTO**

### **5.1 Arquitectura del Sistema**

#### 

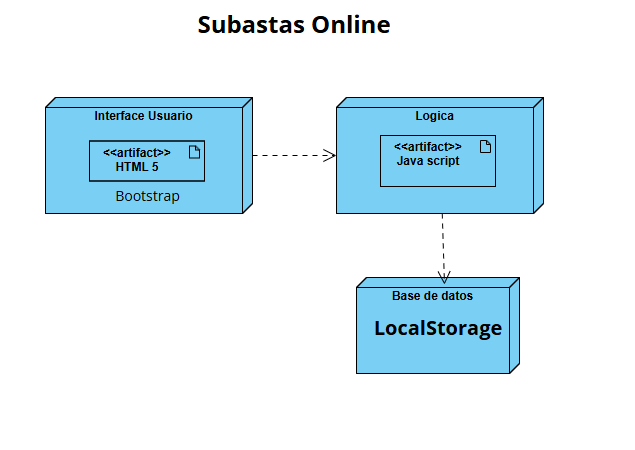
#### 

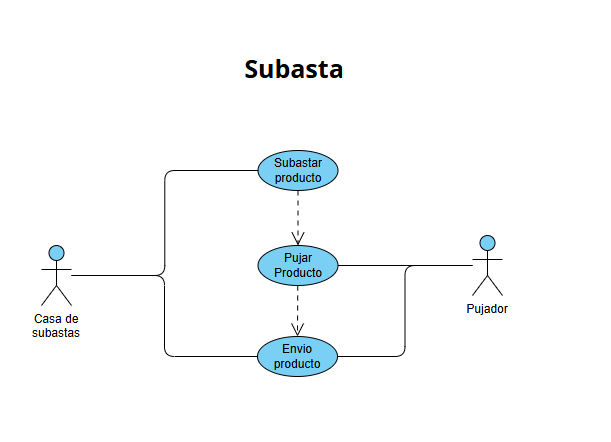
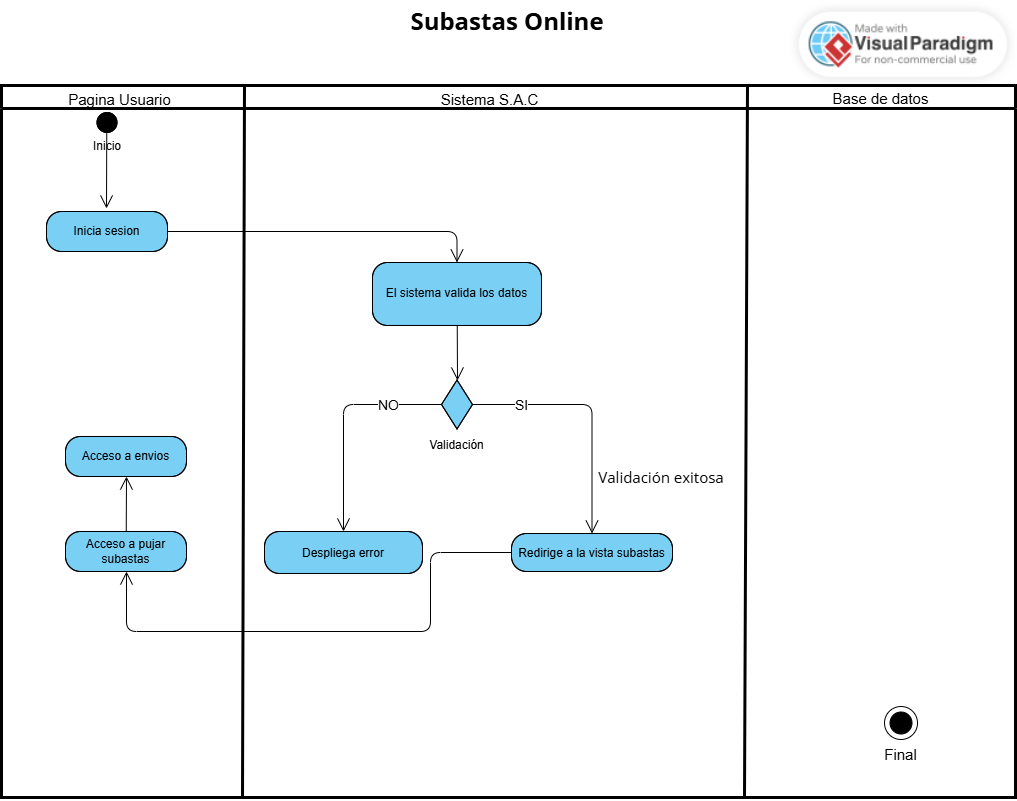
#### 

#### 

#### 

#### **5.1.1 Arquitectura Actual (Fase 1)**



**5.1.2 Caso de uso****5.1.2 Diagrama de procesos**

### **5.2 Metodología de Desarrollo**

#### **5.2.1 Enfoque Iterativo**

* Iteración 1: Estructura HTML y Bootstrap
* Iteración 2: Funcionalidad JavaScript y validaciones
* Iteración 3: Integración de componentes y testing
* Iteración 4: Optimización y documentación

#### **5.2.2 Prácticas de Desarrollo**

* Código limpio: Nomenclatura clara y comentarios descriptivos
* Separación de responsabilidades: HTML estructura, Bootstrap presentación, JS funcionalidad
* Diseño Responsivo

### **5.3 Estrategia de Implementación**

#### **5.3.1 Fases de Desarrollo**

Fase 1: Frontend Básico (Actual)

* Estructura HTML5 semántica
* Estilos CSS personalizados
* Funcionalidad JavaScript básica
* Sistema de validaciones frontend

Fase 2: Backend Integration (Futura)

* API REST con Node.js
* Base de datos Oracle
* React

Fase 3: Optimización (Futura)

* Validación dual
* Estructura de micro servicios

## **6. MODELO DE DATOS**

### **6.1 Estructura de Datos Actual (LocalStorage)**

#### **6.1.1 Usuarios**

JavaScript

{

id: Number,

nombre: String,

correo: String,

contraseña: String,

presupuesto: Number,

fechaRegistro: Date

}

#### **6.1.2 Productos Ganados**

JavaScript

{

id: Number,

nombre: String,

imagen: String,

precio: Number,

fechaGanada: Date,

estado: String,

codigoSeguimiento: String

}

### **6.2 Modelo de Base de Datos Futuro (Oracle)**

#### **6.2.1 Entidades Principales**

* USUARIO: Información de usuarios registrados
* PRODUCTO: Catálogo de productos en subasta
* SUBASTA: Gestión de subastas activas
* PUJA: Registro de pujas realizadas
* ENVIO: Información de envíos
* HISTORIAL\_ENVIO: Tracking detallado

## **7. INTERFACES DE USUARIO**

### **7.1 Páginas Principales**

#### **7.1.1 Página de Login/Registro (Subasta.html)**

* Formulario de inicio de sesión
* Formulario de registro de usuarios
* Validaciones en tiempo real
* Video promocional
* Footer informativo

#### **7.1.2 Página Principal (Inicio.html)**

* Dashboard de subastas activas
* Sistema de pujas en tiempo real
* Gestión de presupuesto
* Historial de pujas ganadas
* Modal de perfil de usuario

#### **7.1.3 Página de Envíos (Envio.html)**

* Configuración de datos de envío
* Seguimiento de paquetes
* Timeline de estados
* Búsqueda por código
* Historial de envíos

### **7.2 Componentes Interactivos**

* Modales: Perfil, confirmaciones, formularios
* Cards: Productos, estadísticas, envíos
* Formularios: Con validación en tiempo real
* Notificaciones: Alerts y toasts
* Timeline: Seguimiento visual de envíos

## **8. PLAN DE TESTING**

### **8.1 Testing Funcional**

* Registro/Login de usuarios
* Flujo completo de subastas
* Sistema de pujas y presupuesto
* Configuración y seguimiento de envíos
* Navegación entre páginas

### **8.2 Testing de Usabilidad**

* Responsive design en múltiples Navegadores
* Accesibilidad básica
* Cross-browser compatibility

### **8.3 Testing de Validaciones**

* Formularios con datos inválidos
* Límites de presupuesto
* Códigos de seguimiento inexistentes
* Sesiones expiradas

## **9. CRONOGRAMA DE DESARROLLO**

### **9.1 Semana 1**

* Estructura HTML básica
* Diseño Bootstrap
* Setup de repositorio Git

### **9.2 Semana 2**

* Funcionalidad JavaScript
* Sistema de validaciones
* Integración de componentes

### **9.3 Semana 3**

* Optimización de código
* Documentación ERS

### **9.4 Semana 4**

* Presentación final
* Entrega
* Evaluación de resultados

## **10. RIESGOS Y MITIGACIONES**

### **10.1 Riesgos Técnicos**

* Compatibilidad de navegadores: Testing extensivo en múltiples browsers
* Pérdida de datos: Backup regular de localStorage
* Performance: Optimización de código

### **10.2 Riesgos de Desarrollo**

* Integración: Testing continuo de componentes

## **11. CONCLUSIONES**

### **11.1 Beneficios Esperados**

* Experiencia de usuario moderna con tecnologías web actuales
* Funcionalidad completa de subastas y gestión de envíos
* Base sólida para futuras expansiones con backend
* Código mantenible siguiendo mejores prácticas

### **11.2 Próximos Pasos**

* Implementación de backend con Node.js y Oracle
* Sistema de pagos real
* Notificaciones push
* Mobile app nativa
* Sistema de calificaciones y reviews