**UNIVERSIDAD NACIONAL DE UCAYALI**

**FACULTAD DE INGENIERÍA DE SISTEMAS E INGENIERÍA CIVIL ESCUELA PROFESIONAL DE INGENIERÍA SISTEMAS**



**SISTEMA DE GESTIÓN DE INVENTARIO DE REPUESTOS**

**CURSO : PROGRAMACIÓN ORIENTADA A OBJETOS II**

**DOCENTE : ING. MG. RICHARD LINARES BARDALES**

**GRUPO : “A”**

**CICLO : “VI”**

**INTEGRANTES : QUISPE CASTRO, SERGIO JUNIOR**

**PUCALLPA-PERÚ 2024**

**SISTEMA DE GESTIÓN DE INVENTARIO DE REPUESTOS**

**Versión 1.1.2 Fecha: [14/01/2025]**

**CONTROL DE VERSIONES**

|  |  |  |  |
| --- | --- | --- | --- |
| **Versión** | **Fecha** | **Responsable** | **Descripción** |
| 1.1.2 | 17/01/2025 | Sergio Junior Quispe Castro | Versión intermedia del documento |

**TABLA DE CONTENIDO**

1. **Visión del Proyecto  
   1.1. Descripción General  
   1.2. Objetivos Principales  
   1.3. Stakeholders**
2. **Alcance y Objetivos  
   2.1. Módulos del Sistema  
   2.2. Gestión de Clientes  
   2.3. Gestión de Vendedores  
   2.4. Gestión de Gerentes  
   2.5. Gestión del Personal Administrativo**
3. **Arquitectura del Sistema  
   3.1. Stack Tecnológico  
   3.2. Requisitos No Funcionales**
4. **Historias de Usuario  
   4.1. Gestión de Usuarios  
   4.2. Gestión de   
   4.3. Reportes y Análisis  
   4.4. Gestión de Pagos  
   4.5. Gestión de Productos  
   4.6. Gestión de Roles y Permisos**
5. **Notas Técnicas  
   5.1. Definition of Ready (DoR)  
   5.2. Definition of Done (DoD)  
   5.3. Priorización y Estimación**
6. **Anexos  
   6.1. Diagramas  
   6.2. Mockups  
   6.3. Repositorio de GitHub  
   6.4. Manual de Instalación y Configuración**
7. **Aprobaciones**
8. **VISIÓN DEL PROYECTO**
   1. **DESCRIPCIÓN GENERAL**

Un Sistema de Gestión de Inventario de Repuestos es una solución integral diseñada para agilizar los procesos manuales y administrativos de un almacén dedicado al almacenamiento de repuestos y productos relacionados. Este sistema facilita la gestión eficiente de inventarios, atención al cliente y facturación directa con la SUNAT mejorando la productividad y proporcionando una experiencia de usuario sencilla.

* 1. **OBJETIVOS PRINCIPALES**
* Optimización de inventarios
* Mejorar la experiencia de facturación
* Automatización de procesos administrativos
  1. **STAKEHOLDERS**
* Personal administrativo (Usuarios primarios)
* Clientes (Beneficiarios indirectos)
* Departamento de TI (Soporte y mantenimiento)

1. **ALCANCE Y OBJETIVOS**
   1. **MÓDULOS DEL SISTEMA**
      1. **Gestión de clientes**

* Registro de datos
* Generación de facturas
  + 1. **Gestión de usuarios**
* Registro de datos
* Generación de clientes, productos y facturaciones.
  + 1. **Gestión de productos**
* Planificación de inventario
* Gestión de productos y precios
  + 1. **Gestión de Facturación**
* Gestión de facturación SUNAT.

1. **ARQUITECTURA DEL SISTEMA**
   1. **STACK TECNOLÓGICO**

**Frontend:**

* Bootstrap para diseño y componentes visuales.
* HTML, CSS y JavaScript para la estructura y estilos.
* Thymeleaf (para integraciones con Spring Boot).

**Backend:**

* Lenguaje Java
* Spring Boot para la lógica de negocio y la API REST.
* Spring Data JPA para el acceso a datos.
* Lombok para reducir el código repetitivo.

**Base de Datos**:

* MySQL para almacenamiento de datos.

**Infraestructura:**

* CI/CD con GitHub Actions
  1. **REQUISITOS NO FUNCIONALES**

**Disponibilidad:**

* Disponibilidad del sistema en horario laborales (7:00 a.m. - 5:00 p.m.).

**Rendimiento:**

* Tiempo de respuesta < 1.5 segundos
* Manejo de al menos 2 usuarios concurrentes.

**Seguridad:**

* Autenticación de usuario administrativo.

**Escalabilidad:**

* Capacidad para integrar nuevos módulos funcionales sin afectar el rendimiento.

1. **HISTORIAS DE USUARIO**
   1. **EPIC 1: Gestión de Usuarios**
      1. **HU-001: Registro** **de Usuarios**

**Como** administrador **Quiero** registrar nuevos usuarios  **Para** realizar facturas en SUNAT, gestionar productos, gestionar categorías y gestionar clientes

**Criterios de Aceptación:**

1. El formulario debe incluir:
   * Datos personales (nombre completo, correo, usuario, contraseña)
   * Contacto (teléfono)
2. Validar que el usuario no esté duplicado
3. Las contraseñas deben ser cifradas
4. Mostrar mensaje de confirmación al registrar con éxito.

**Estimación: 3 puntos  
Prioridad: Alta**

* 1. **EPIC 2: Gestión de Ventas y Productos**
     1. **HU-002: Proceso de registro de Ventas**

**Como** administrador

**Quiero** registrar una venta en la SUNAT

**Para** llevar control de los productos vendidos y el cliente que compró

Criterios de Aceptación:

1. Ingresar cantidad vendida
2. Ingresar datos del cliente (nombre, teléfono)
3. Calcular y mostrar el total automáticamente

Estimación: 5 puntos  
Prioridad: Alta

* + 1. **HU-003: Actualización de Stock de Productos**

**Como** administrador  
**Quiero** actualizar el stock de productos  
**Para** mantener la disponibilidad correcta de los repuestos en el inventario

Criterios de Aceptación:

1. Permitir buscar productos por nombre, marca o código
2. Permitir ingresar nuevas cantidades para actualizar el stock
3. Mostrar mensaje de éxito o error según corresponda

Estimación: 4 puntos  
Prioridad: Alta

* + 1. **HU-004: Reporte de Productos**

**Como** administrador  
**Quiero** un reporte detallado de inventario   
**Para** monitorear la disponibilidad de productos

Criterios de Aceptación:

1. Mostrar productos con bajo stock
2. Generar alertas de productos agotados
3. Permitir filtro por categorías de productos

Estimación: 6 puntos  
Prioridad: Media

* 1. **EPIC 3: Gestión de Productos**
     1. **HU-005: Gestión de Categorías de Productos**

**Como** administrador  
**Quiero crear y administrar categorías de productos**  
**Para** organizar los repuestos en el sistema

Criterios de Aceptación:

1. Crear nuevas categorías con nombre y descripción
2. Editar categorías existentes
3. Eliminar categorías que no tengan productos asociados
4. Validar que no se repita el nombre de una categoría

Estimación: 3 puntos  
Prioridad: Baja

* 1. **EPIC 4: Gestión de Productos**
     1. **HU-006: Gestión de Productos**

**Como** administrador  
**Quiero asignar usuarios**   
**Para** controlar su nivel de participación

**Criterios de Aceptación**:

1. Mostrar una lista de productos
2. Permitir modificación de datos
3. Validación de stock

Estimación: 6 puntos  
Prioridad: Alta

**Notas Técnicas**

**Definition of Ready (DoR):**

* Historia claramente descrita
* Criterios de aceptación definidos
* Estimación realizada
* Dependencias identificadas
* Recursos necesarios disponibles

**Definition of Done (DoD):**

* Código implementado y probado
* Pruebas automatizadas escritas
* Documentación actualizada
* Code review completado
* Desplegado en ambiente de pruebas
* Aprobado por Product Owner

**Priorización:**

* Alta: Crítico para el MVP
* Media: Importante pero no bloqueante
* Baja: Deseable, pero puede esperar

**Estimación:**

* **1-2 puntos**: Validaciones simples.
* **3-5 puntos**: Funcionalidades medianas.
* **8 puntos**: Funciones complejas.
* **13+ puntos**: Requieren dividirse en tareas más pequeñas.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID HU** | **Historia de Usuario** | **EPIC** | **Estimación (Puntos)** | **Prioridad** |
| HU-001 | Registro de Usuarios | Gestión de Usuarios | 3 | Alta |
| HU-002 | Proceso de registro de ventas en la SUNAT | Gestión de ventas | 5 | Alta |
| HU-003 | Actualización de stock de productos | Gestión de productos | 4 | Alta |
| HU-004 | Reporte de productos | Gestión de productos | 6 | Media |
| HU-005 | Gestión de categorías y productos | Gestión de productos | 3 | Baja |
| HU-006 | Gestión de productos | Gestión de productos | 6 | Alta |

1. **ANEXOS**
   1. **DIAGRAMAS**

* **Arquitectura del sistema:** El sistema de gestión para la tienda de muebles está diseñado bajo la arquitectura Modelo-Vista-Controlador (MVC) en un entorno monolítico. Esta elección se basa en la simplicidad, el control centralizado y la adecuación del diseño a las necesidades del proyecto, que no requiere la división en microservicios debido al número limitado de módulos involucrados.

La arquitectura MVC divide claramente las responsabilidades de la siguiente manera:

* + **Modelo**: Gestiona la lógica de negocio y los datos, abarcando entidades como productos, tipos de productos, clientes, empleados y ventas.
  + **Vista**: Ofrece la interfaz gráfica que permite a los usuarios interactuar con el sistema, garantizando una experiencia de usuario intuitiva y organizada.
  + **Controlador**: Actúa como intermediario, procesando las solicitudes de los usuarios, interactuando con el modelo y seleccionando las vistas correspondientes para la respuesta.
* **Justificación de la Arquitectura:**
  + **Reducción de costos:** Al adoptar una arquitectura monolítica, se eliminan los gastos relacionados con la gestión de varios servicios independientes.
  + **Mayor seguridad:** La arquitectura MVC favorece una separación clara entre la lógica de negocio y la presentación, lo que disminuye las posibilidades de ataques directos.
  + **Simplicidad en el mantenimiento:** La estructura ordenada de MVC facilita la expansión futura del sistema, permitiendo agregar nuevos roles o módulos sin complicaciones.
* **Modelo de datos**

Tabla

Descripción generada automáticamente

Imagen que contiene biombo, con baldosas, listo, parado

Descripción generada automáticamente

* 1. **MOCKUPS**
* Interfaces principales

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Sitio web

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Tabla

Descripción generada automáticamente Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

**GITHUB**

* CÓDIGO FUENTE
* INFORME
* BD (SCRIPT)

**APROBACIONES**

Product Owner Scrum Master

Fecha: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Fecha: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tech Lead Stakeholder Principal

Fecha: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Fecha: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_