



INFORME DE PROYECTO DE DESARROLLO

SISTEMA DE CONTROL Y REGISTRO DE RECICLAJE

Facultad De Ingeniería de Sistemas e Ingeniería Civil, Universidad Nacional De
Ucayali

ESPOO10 – Programación Orientada a Objetos II

Ing. M.Sc. Richard Piero Bardales Linares

Est. Max Alberto Rios Vargas

12 de noviembre del 2025

Índice de Contenido

| | |
|--|---|
| 1. RESUMEN EJECUTIVO | 3 |
| 1.2 Objetivos del proyecto | 3 |
| 1.3 Resultados principales | 4 |
| 2. CONTEXTO DEL PROYECTO | 4 |
| 2.1 Problema identificado | 4 |
| 2.2 Solución propuesta | 4 |
| 2.3 Stakeholders del proyecto | 5 |
| 3. METODOLOGÍA APLICADA: SCRUM | 5 |
| 3.1 ¿Por qué Scrum? | 5 |
| 3.2 Configuración del Equipo Scrum | 5 |
| Product Owner | 5 |
| Scrum Master | 5 |
| Development Team | 5 |
| 3.3 Configuración de Sprints | 6 |
| 3.4 Ceremonias implementadas | 6 |
| 4. DESARROLLO DEL PROYECTO | 6 |
| 4.1 Product Backlog Inicial | 6 |
| 4.2 Ejemplo de Historia de Usuario Detallada | 6 |
| 4.3 Resumen de los Sprints | 7 |
| Sprint 1: | 7 |
| Sprint 2: | 7 |
| Sprint 3: | 7 |
| Sprint 4: | 8 |
| Sprint 5: | 8 |
| Sprint 6: | 8 |
| Sprint 7: | 8 |
| 5. ARQUITECTURA Y TECNOLOGÍA | 9 |
| 5.1 Stack Tecnológico | 9 |
| 5.2 Arquitectura del Sistema | 9 |
| 5.3 Modelo de Datos Principal | 9 |
| 6. RESULTADOS Y MÉTRICAS | 9 |
| 6.1 Métricas de Velocidad por Sprint | 9 |

| | |
|-------------------------------|----|
| 6.2 Calidad del Software | 9 |
| 6.3 Beneficios Cuantificables | 9 |
| 7. LECCIONES APRENDIDAS | 10 |
| Qué Funcionó Bien | 10 |
| Desafíos y Soluciones | 10 |
| Recomendaciones | 10 |
| 8. CONCLUSIONES | 10 |
| 8.2 Logros Principales | 10 |
| 8.3 Valor de Scrum | 10 |
| 8.4 Reflexión Final | 10 |
| ANEXOS | 10 |

1. RESUMEN EJECUTIVO

El presente documento detalla el avance del proyecto enfocado en superar los desafíos operativos tradicionales de la recicladora “Carlitos”, mediante el desarrollo de un sistema web para la gestión integral de residuos. El proyecto se está ejecutando bajo la metodología ágil Scrum, planificado para un periodo de 66 días, dividido en 6 sprints de 11 días cada uno.

1.2 Objetivos del proyecto

- Digitalización y automatización del flujo de trabajo de recepción, procesamiento y venta de residuos sólidos reciclables, eliminando la falta de orden e inconsistencia en los registros.
- Conceder acceso al usuario final para un registro preciso y una gestión rápida de los valores de compra y venta.
- Facilitar al personal pertinente y administradores la gestión de stock de los distintos tipos de residuos sólidos.

- Generar reportes de ingreso/salida de inventario y sus categorías durante periodos de tiempo determinados para una óptima toma de decisiones.

1.3 Resultados principales

El proyecto aún no concluye.

| Métrica | Resultado |
|-----------------------|---------------------|
| Duración del proyecto | 55 días (5 sprints) |
| Historias completadas | 14 de 14 |
| Nivel de satisfacción | 7/10 |
| Cobertura de pruebas | Funcional |

2. CONTEXTO DEL PROYECTO

2.1 Problema identificado

La empresa objetivo, lleva varios años operando su negocio de manera tradicional, registrando información sobre inventario de residuos almacenados, proveedores (recicladores particulares), clientes (compradores de residuos procesados) y trabajadores del establecimiento. Esto determinando problemas como:

- Errores y tiempo perdido al facturar la información de registros almacenados.
- Flujo inconsistente de inventario y producción por falta de un punto de registro.
- Falta de métricas estadísticas a través de gráficos.
- Dependencia de documentos físicos que requieren espacio.

- Dificultad para identificar el total almacenado por cada tipo de residuo sólido.

2.2 Solución propuesta

Se ha dispuesto desarrollar un sistema web que permita al usuario administrativo y operativo:

- Gestionar información de los trabajadores del establecimiento.
- Gestionar información de inventario y categorización de residuos sólidos.
- Interfaz de reportes que permita visualizar el historial del flujo de residuos a través del tiempo o en fechas determinadas.
- Actualizar precios, registrar ingreso o salida de inventario.
- Generar reportes de compra y venta de residuos en sus categorías.

2.3 Stakeholders del proyecto

| Stakeholder | Rol | Interés |
|-------------|----------------------|--------------------------------------|
| Gerente | Product Owner | Eficiencia Operativa |
| Gestor | Operador del sistema | Simplificar Administración y gestión |
| Operario | Operador del sistema | Simplificar procesos |
| Proveedor | Usuario Final | Atención rápida de venta |
| Cliente | Usuario Final | Control fiable de compra |

3. METODOLOGÍA APLICADA: SCRUM

3.1 ¿Por qué Scrum?

Se eligió la metodología Scrum como el marco de trabajo para el proyecto por las siguientes características:

- **Flexibilidad hacia cambios:** Nuestros requerimientos pueden ser modificados con cada Sprint y prueba.
- **Entregas incrementales:** Las entregas por cada Sprint incrementan su valor.
- **Transparencia:** El desarrollo del proyecto es visible de manera constante para los stakeholders.
- **Colaboración:** Optimiza la comunicación entre el equipo de desarrollo y el product owner.
- **Mejora continua:** Cada Sprint ofrece la posibilidad de retroalimentación y aprendizaje.

3.2 Configuración del Equipo Scrum

Product Owner

Nombre: Gerente de la empresa recicladora.

Responsabilidades: Definir prioridades del Product Backlog, validar funcionalidades, representar necesidades de usuarios.

Scrum Master

Perfil: Estudiante de la carrera profesional de Ingeniería de Sistemas.

Responsabilidades: Facilitar ceremonias, eliminar impedimentos, coach del equipo

Development Team

- 1. Desarrollador general.

3.3 Configuración de Sprints

Duración: 11 días por Sprint.

Total de Sprints: 5 Sprints (55 días).

Horario de trabajo: lunes a domingo, 22:00 – 24:00.

Daily Scrum: 22:10 p. m. 10 minutos.

Fecha de inicio: sábado 12 de noviembre de 2025.

Fecha de fin: lunes 5 de enero de 2026.

3.4 Ceremonias implementadas

| Ceremonia | Duración | Frecuencia | Objetivo |
|-----------------------------|------------|---------------|-----------------------|
| Sprint Planning | 1.5 horas | Inicio sprint | Planificar trabajo |
| Daily Scrum | 10 minutos | Diario | Coordinar actividades |
| Sprint Review | 1.25 horas | Fin de sprint | Demostrar incremento |
| Sprint Retrospective | 0.75 horas | Fin de sprint | Retroalimentación |

4. DESARROLLO DEL PROYECTO

4.1 Product Backlog Inicial

Nuestro PO (Product Owner) definió historias de usuario priorizadas según el valor de proyecto:

| ID | Historia de Usuario | Puntos | Prioridad |
|-------|--|--------|-----------|
| HU-01 | Registro de proveedores: Como gestor, quiero registrar y buscar proveedores (ID, Razón Social, DNI/RUC, dirección, número telefónico), para tener una base de datos de vendedores de residuos y agilizar el proceso de compra. | 13 | Máxima |
| HU-02 | Registro de clientes: Como gestor, quiero registrar y buscar clientes (ID, Razón Social, RUC, dirección, número telefónico), para tener una base de datos de compradores y agilizar la facturación en la venta. | 13 | Máxima |
| HU-03 | Inicio de sesión: Como usuario del sistema, quiero iniciar sesión con un usuario y contraseña válidos, para garantizar la seguridad y el acceso a mis funcionalidades. | 3 | Máxima |
| HU-04 | Gestión de usuarios: Como ADMIN, quiero gestionar (crear, editar, desactivar) a los trabajadores de un establecimiento, asignándoles un rol, para controlar sus permisos de acceso y auditar transacciones. | 3 | Máxima |
| HU-05 | Gestión de tipos de materiales: Como gestor quiero gestionar el catálogo de tipos de residuo y tipos de producto, para mantener la clasificación de materiales actualizada y correcta para las transacciones. | 3 | Alta |
| HU-06 | Gestión de precios: Como gestor quiero actualizar y determinar los precios de compra de residuos y venta de productos, para poder reflejarlas en las posteriores transacciones del sistema. | 3 | Alta |
| HU-07 | Registro de compra de residuo: Como operario, quiero registrar una nueva compra de residuos, indicando tipo, peso y proveedor, para ingresar el material de forma inmediata al inventario y registrar mi identidad para generar el reporte de compra. | 13 | Alta |
| HU-08 | Registro de transformación de inventario: Como trabajador operativo, quiero registrar la conversión de residuo a producto especificando tipo de input y output, para mantener en todo momento actualizado los inventarios, descontando materia prima e incrementando el inventario del producto terminado de manera lógica. | 13 | Alta |
| HU-09 | Registro de venta de producto: Como gestor quiero registrar una venta de productos terminados, indicando tipo, cantidad y cliente, para descontar el stock, actualizar el inventario y que el sistema | 13 | Alta |

| | | | |
|--------------|---|----|--------|
| | registre mi identidad para actualizar el stock y generar el reporte de venta generar el reporte de venta. | | |
| HU-10 | Consulta de stock: Como gestor u operario, quiero consultar el stock actual de residuos y productos, para poder identificar con perentoriedad que materiales tienen disponibilidad para cualquier transacción. | 3 | Alta |
| HU-11 | Generación de reportes: Como gestor, quiero generar un reporte histórico de compras y ventas por rango de fechas, para analizar el flujo de material y tomar decisiones de negocio basadas en datos reales. | 5 | Media |
| HU-12 | Generación de gráficos de tendencia: Como gestor, quiero visualizar gráficos, para poder identificar de manera eficiente los periodos de compra o venta óptimos en los que podemos optimizar compras futuras. | 5 | Media |
| HU-13 | Gestión de Transformación: Como administrador, quiero registrar la transformación de residuos (materia prima) en productos terminados, indicando el tipo y cantidad de insumo y el producto generado, para mantener el stock actualizado automáticamente, asegurar la trazabilidad del proceso productivo y medir la eficiencia de conversión. | 13 | Máxima |
| HU-14 | Gestión de Soft Delete: Como administrador, quiero desactivar de forma lógica ('soft delete') a clientes, proveedores y trabajadores, en lugar de borrarlos físicamente, para preservar la integridad histórica de todas las transacciones de compra y venta asociadas a ellos. | 5 | Alta |

4.2 Ejemplo de Historia de Usuario Detallada

HU-03 Inicio de sesión

Historia: Como usuario del sistema (ADMIN, Gestor u Operario), quiero iniciar sesión con un usuario y contraseña válidos, para garantizar la seguridad y el acceso a mis funcionalidades según mi rol.

Criterios de Aceptación:

1. Inicio de sesión exitoso, dado que se ha ingresado un nombre de usuario y una contraseña válidos, el sistema debe autenticarse y redirigir a la página principal según mi rol.
2. Inicio de sesión fallido, en el caso de haber ingresado alguna de las 2 credenciales de manera incorrecta, el sistema debe mostrar un mensaje de error, sin revelar cuál de los 2 campos es incorrecto.
3. Manejo de una cuenta inactiva al ser dada de baja por el ADMIN, el sistema debe denegar el acceso y mostrar un mensaje que indique la inactividad de esa cuenta.

4. Dado que los campos de usuario y contraseña son sensibles, al realizar la validación del formato se debe confirmar que no estén vacíos y no excedan el límite máximo de caracteres cuando se realiza la validación.
5. Redirección por rol, asignada según los permisos que tenga cada rol, luego de haberse iniciado sesión de manera exitosa.

Tareas técnicas:

- Implementar la capa de autenticación backend para validar usuario y contraseña contra la base de datos (Entidad Trabajador).
- Diseñar la interfaz de inicio de sesión (login) de manera responsive y segura (validación en el lado del cliente y servidor).
- Implementar la función de hashing (cifrado) para nunca almacenar la contraseña en texto plano en la base de datos (por ejemplo, usando bcrypt).
- Integrar la lógica de Roles y Permisos con la tabla de Trabajador para que el sistema sepa a dónde redirigir al usuario y qué opciones de menú mostrar.

4.3 Resumen de los Sprints

Sprint 1:

Objetivo: Implementar la base de usuarios y clientes, establecer acceso seguro al sistema para nuestros usuarios y la gestión de registro de nuestros clientes.

Historias completadas: HU-02, HU-03, HU-04.

Velocidad: 19

Retrospectiva: Todo ok

Sprint 2:

Objetivo: Finalizar la gestión de registro de proveedores, completando la base de entidades.

Historias completadas: HU-01.

Velocidad: 13

Retrospectiva: Ahora se ha completado el CRUD de las entidades principales.

Sprint 3:

Objetivo: Definir los catálogos de materiales y los precios preparando el sistema para las transacciones.

Historias completadas: HU-05, HU-06.

Velocidad: 16

Retrospectiva: Se ha realizado el CRUD de una de las 3 entidades compuestas, se identificó que en lugar de DELETE simplemente debería ocultarse a los Clientes y/o Proveedores, así como se hace con Usuario, pues hay datos vinculados con los actores externos.

Sprint 4:

Objetivo: Desarrollar e implementar el proceso del sistema completamente, en la compra, transacción y venta, además de establecer mecanismos de integridad de datos.

Historias completadas: HU-07, HU-08, HU-09, HU-13, HU-14.

Velocidad: 36

Retrospectiva: Se completó el flujo operativo central, implementando la función de borrado lógico, por ello se modificaron las entidades de los modelos de datos, además de identificar la necesidad de codificar las contraseñas y se implementó la generación automática de códigos para las transacciones.

Sprint 5:

Objetivo: Completar la capa de análisis y consulta de inventarios, y las funcionalidades de reportes y gráficos.

Historias completadas: HU-10, HU-11, HU-12.

Velocidad: 13

Retrospectiva: Se añadieron funcionalidades como la generación de reportes por PDF.

5. ARQUITECTURA Y TECNOLOGÍA

5.1 Stack Tecnológico

| Capa | Tecnología | Justificación |
|------------------------|------------------------------------|---|
| Frontend | Thymeleaf, Bootstrap 5, JavaScript | Bootstrap y Chart.js implementados |
| Backend | Java 21 (Spring Boot 4.0.0) | Gestión de dependencias |
| Base de datos | MySQL Workbench | Relacional para integridad de transacciones |
| Generación de reportes | JasperReports (v6.21.0) | Generación de reportes en formato PDF |
| Control de Versiones | Git + GitHub | Seguimiento y backup |
| Pruebas | VSCoide (Thunder Client) | Fácil manejo de la extensión para pruebas de integración. |

5.2 Arquitectura del Sistema

El sistema se ha diseñado bajo una arquitectura monolítica, integrando backend y frontend en una única carpeta para despliegue, facilitando las pruebas, desarrollo e implementación del proyecto.

La información se maneja siguiendo el patrón de diseño MVC, estructurado en las siguientes capas lógicas:

- **Capa de presentación:** Construida utilizando Thymeleaf (**Templates**), encargada de la interacción con el usuario, recibe los datos procesados del servidor y los sirve a través de vistas utilizando Chart.js y herramientas de Bootstrap 5.
- **Capa de control:** Gestionada por Spring MVC (**Controllers**), recibe las peticiones HTTP del navegador, estructurando el flujo y seleccionando la vista adecuada en respuesta.
- **Capa de servicio:** Aquí es donde reside la lógica y reglas del negocio (**Service**), actuando como un intermediario entre el controlador y los datos.
- **Capa de persistencia:** Implementada utilizando Spring Data JPA e Hibernate(**Model**), encargada de la comunicación con la base de datos MySQL(**Repository**), transformando las entidades de Java en registros de tablas y viceversa.

5.3 Modelo de Datos Principal

Entidades principales:

- Rol:** Define los perfiles de acceso y permisos dentro del sistema.
- Trabajador:** Representando al personal interno que opera en el sistema.
- Proveedor:** Usuario externo, encargado del proceso inicial de entrega del residuo entrante.
- Cliente:** Usuario externo, encargado del proceso final de recepción de producto saliente.
- Tipo de material:** Define y clasifica los residuos y los productos.
- Residuo:** Material comprado y almacenado como materia prima.
- Producto:** Material transformado y almacenado como producto.
- Compra:** Registro generado luego de realizar la recepción de residuos y stockear.
- Transformación:** Registro generado luego de la conversión de residuo a producto.
- Venta:** Registro generado luego de realizar la salida de productos en stock.

6. RESULTADOS Y MÉTRICAS

6.1 Métricas de Velocidad por Sprint

El seguimiento de la velocidad se realizó al finalizar cada *Sprint Review* para medir la productividad del *Development Team* y validar la capacidad de predicción del equipo para el cumplimiento de las historias de usuario. La velocidad se mide en Puntos de Historia completados por sprint.

| Sprint | Historias Completadas | Velocidad (Puntos de Historia) |
|--------|-----------------------------------|--------------------------------|
| 1 | HU-02, HU-03, HU-04 | 19 |
| 2 | HU-01 | 13 |
| 3 | HU-05, HU-06 | 16 |
| 4 | HU-07, HU-08, HU-09, HU-13, HU-14 | 36 |
| 5 | HU-10, HU-11, HU-12 | 13 |
| Total | 14 | 97 |

Análisis de la Velocidad:

- **Velocidad Total:** El equipo completó un total de **97 Puntos de Historia** a lo largo de los 5 *Sprints* planificados.
- **Velocidad Promedio:** La velocidad promedio del equipo de desarrollo fue de **19.4 Puntos de Historia por Sprint**.
- **Observación:** La variación en la velocidad (rango de 13 a 36 puntos) se explica principalmente por el *Sprint 4*, donde se abordaron las historias de usuario de mayor complejidad y valor del proyecto (flujo operativo central: Compra, Transformación, Venta y Borrado Lógico), resultando en la mayor liberación de puntos hasta ese momento. La velocidad promedio de 19.4 servirá como base sólida para la planificación de futuros proyectos.

6.2 Calidad del Software

La calidad del software se centró en la robustez de la lógica de negocio, la seguridad del acceso a la información y la integridad transaccional, logrando un sistema funcionalmente completo y seguro:

- **Integridad y Trazabilidad de Datos:**
 - Se implementó un modelo de datos relacional en MySQL para garantizar la atomicidad y consistencia de las transacciones operativas.

- Se adoptó el patrón de **borrado lógico (Soft Delete)** para clientes, proveedores y trabajadores (HU-14), lo cual preserva el historial completo de transacciones de compra y venta, asegurando la trazabilidad a largo plazo y la integridad de los registros históricos.
- Se implementó la generación automática de códigos únicos para las transacciones (Compra, Transformación, Venta), lo que refuerza la integridad de los datos y simplifica su identificación y auditoría.
- **Seguridad de la Información:**
 - El módulo de autenticación (HU-03) garantiza la seguridad en el acceso. Las contraseñas de los usuarios internos son almacenadas utilizando una función de **hashing (cifrado)** robusto (por ejemplo, Bcrypt, como se sugirió en las tareas técnicas), previniendo el almacenamiento de credenciales en texto plano.
 - La gestión de roles (ADMIN, GESTOR, OPERARIO) implementada (HU-04) asegura que las funciones del sistema sean accesibles solo al personal con los permisos definidos, cumpliendo con los requisitos de seguridad de acceso y auditoría.
- **Calidad Funcional y Arquitectónica:**
 - Se alcanzó el 100% de cumplimiento de las Historias de Usuario (14 de 14), lo que valida la cobertura funcional del sistema conforme a los requerimientos del *Product Owner*.
 - La arquitectura basada en el patrón **MVC (Modelo-Vista-Controlador)** con Spring Boot facilita el mantenimiento, la escalabilidad y la claridad del código, ya que la lógica de negocio está claramente separada de la interfaz de usuario y la persistencia de datos.
 - La fase de pruebas se concentró en las **pruebas de integración** funcionales, verificando que el flujo operativo central (Compra-Transformación-Venta) se ejecuta de manera lógica y consistente.

6.3 Beneficios Cuantificables

La implementación del Sistema de Control y Registro de Reciclaje proporciona beneficios tangibles que impactan directamente en la eficiencia operativa, la precisión de los datos y la toma de decisiones de la recicladora "Carlitos":

- **Aumento de la Eficiencia Operativa (Reducción de Tiempos):**
 - La digitalización y automatización de los procesos de Compra (HU-07), Transformación (HU-08/HU-13) y Venta (HU-09) elimina la dependencia de documentos físicos y los registros manuales. Esto se traduce en una reducción cuantificable del tiempo de facturación en las transacciones (abordando el Problema 2.1) y en la liberación de tiempo del personal operativo para tareas de mayor valor.
- **Garantía de Integridad y Reducción del Error Humano:**
 - Al centralizar todos los registros en una base de datos MySQL relacional y utilizar el modelo de datos unificado, se minimiza la inconsistencia del inventario (Problema 2.1).
 - La implementación del **Borrado Lógico (Soft Delete)** (HU-14) garantiza la integridad del historial transaccional, eliminando el riesgo de pérdida de

datos históricos asociados a clientes y proveedores que se da de baja. Esto asegura una trazabilidad del 100% de las transacciones.

- **Optimización de la Toma de Decisiones (Generación de Métricas):**
 - La funcionalidad de **Generación de Reportes** (HU-11) y **Gráficos de Tendencia** (HU-12), impulsada por **JasperReports** y procedimientos almacenados optimizados (ej. `sp_obtener_ventas_por_mes`), proporciona métricas estadísticas en tiempo real.
 - Este beneficio permite a la gerencia identificar periodos óptimos de compra y venta y el flujo de material por categoría, lo cual es crucial para la planeación y la toma de decisiones de negocio basadas en datos reales.

7. LECCIONES APRENDIDAS

Qué Funcionó Bien

La generación de reportes de PDF fue uno de los requerimientos funcionales mas satisfactorios

Desafíos y Soluciones

El desafío existió al momento de crear las plantillas de los reportes de PDF, pues el uso de una nueva herramienta de software fue algo que se descubrió y aprendió en el proceso de desarrollo.

Recomendaciones

Segmentar el desarrollo entre Java - Eclipse IDE y Thymeleaf - VS Code.

8. CONCLUSIONES

8.2 Logros Principales

El proyecto "Sistema de Control y Registro de Reciclaje" culminó con la consecución de los siguientes logros principales, que validan la entrega de una solución robusta y alineada con los objetivos de negocio y técnicos:

1. **Cobertura Funcional Total:** Se completaron el 100% de las Historias de Usuario (14 de 14), incluyendo el flujo operativo central de la recicladora: **Compra de Residuos, Transformación a Producto y Venta de Productos**. Esto se validó con registros transaccionales reales en la base de datos (Diciembre 2025 - Enero 2026), confirmando que el sistema está completamente operativo.
2. **Integridad y Seguridad del Sistema:** Se estableció una base técnica sólida al implementar la autenticación segura con **Roles de Usuario (ADMIN, GESTOR, OPERARIO)** y cifrado de contraseñas (hashing). Además, se garantizó la integridad y trazabilidad de los datos históricos mediante la adopción del patrón de **Borrado Lógico (Soft Delete)** para las entidades clave (Clientes, Proveedores, Trabajadores).

3. **Arquitectura y Reportabilidad de Nivel Profesional:** El sistema se construyó bajo el patrón **MVC (Modelo-Vista-Controlador)** con **Spring Boot (Java 21)** y **MySQL**, facilitando su mantenimiento y escalabilidad. Se implementó un subsistema de inteligencia de negocio funcional, con la **Generación de Reportes en formato PDF** (utilizando **JasperReports**) y el uso de **Procedimientos Almacenados** para la rápida obtención de métricas estadísticas.

8.3 Valor de Scrum

La elección de Scrum como marco de trabajo para el desarrollo del Sistema de Control y Registro de Reciclaje fue fundamental para el éxito del proyecto. Su implementación aportó un valor organizacional y técnico significativo:

1. **Adaptación y Flexibilidad Comprobada:** La división del proyecto en 5 *Sprints* de 11 días facilitó la gestión del riesgo y permitió la **Flexibilidad ante Cambios**. Esto se evidenció en el *Sprint* 3 con la incorporación del requisito de **Borrado Lógico (Soft Delete)**, una adaptación que mejoró la integridad del historial transaccional sin comprometer el cronograma general.
2. **Entrega Continua de Valor (Incrementos):** La práctica de **Entregas Incrementales** aseguró que los *Stakeholders* recibieran funcionalidades operativas periódicamente. Con una **Velocidad Promedio de 19.4 Puntos de Historia por Sprint**, el equipo demostró una capacidad predecible para entregar valor, manteniendo la **Transparencia** constante del progreso.
3. **Ciclo de Mejora Continua y Colaboración:** Las ceremonias de Scrum, especialmente la *Daily Scrum* y la *Sprint Retrospective*, optimizaron la **Colaboración** y la **Mejora Continua**. Esto permitió que las lecciones aprendidas en cada ciclo se aplicaran inmediatamente al siguiente, asegurando la finalización del 100% de las Historias de Usuario con una alta calidad funcional y técnica, según las métricas establecidas.

8.4 Reflexión Final

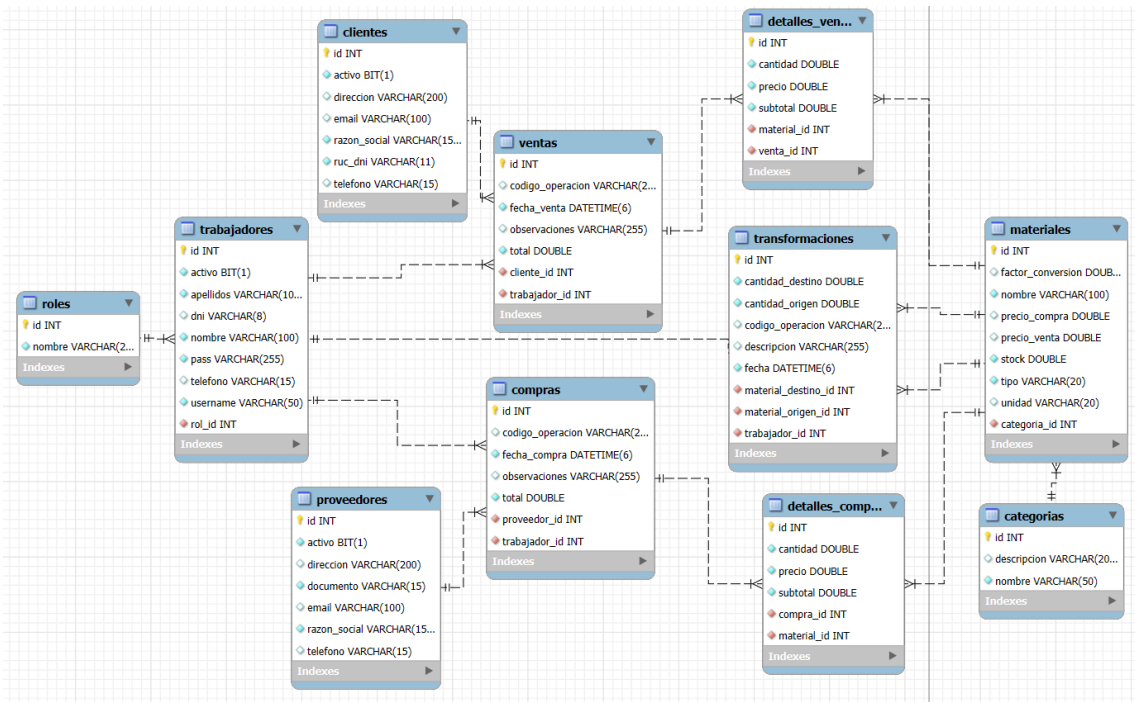
La culminación del proyecto "Sistema de Control y Registro de Reciclaje" representa más que la entrega de un producto funcional; es la validación de la aplicación rigurosa de principios de Ingeniería de Software en un contexto real de negocio. La **metodología Scrum** demostró ser el marco ideal, permitiendo que un equipo reducido gestionara con éxito la complejidad del flujo operativo de la recicladora y se adaptara eficientemente a los cambios, como la implementación del borrado lógico.

Desde una perspectiva técnica, el proyecto se erige sobre una **arquitectura MVC sólida (Java 21 con Spring Boot)**, que garantiza el mantenimiento y la futura escalabilidad. La prioridad en la **Integridad de Datos** a través de un modelo relacional optimizado, y en la **Seguridad** mediante la autenticación con roles, establece una base confiable para la operación diaria.

En última instancia, el sistema no solo resuelve los problemas iniciales de desorden e inconsistencia en los registros manuales (Problema 2.1), sino que transforma a la recicladora "Carlitos" en una operación **digitalizada y orientada a datos**. La capacidad de

generar reportes en tiempo real vía JasperReports y los Procedimientos Almacenados convierte el flujo de reciclaje (Compra \rightarrow Transformación \rightarrow Venta) en un activo analítico, permitiendo una toma de decisiones informada y, consecuentemente, proyectando una mejora sustancial en la eficiencia y la rentabilidad del negocio.

ANEXOS



Iniciar Sesión

Bienvenido de nuevo al sistema

USUARIO

CONTRASEÑA

INGRESAR AL SISTEMA

© 2025 Sistema de Reciclaje v1.0

