



# FACULTAD DE INGENIERÍA

---

CARRERA DE INGENIERÍA DE SISTEMAS  
COMPUTACIONALES

## INFORME PROYECTO

### **Autores:**

Flores Castillo, Frank Junior Alberto  
Garcia Moreno, Jose Luiz  
Polo Niquin, Milagros Yeraldin  
Rodriguez Leiva, Sebastian Enrique

### **Curso:**

Técnicas de Programación Orientada Objetos

### **Docente del Curso:**

Jorge Ricardo Perez Vigil

Trujillo – Perú

2025-1

# INSTRUCCIONES

En grupos de Proyecto:

- Documentar criterios de aceptación de cada requerimiento.
- Implementar haciendo uso de Java los requerimientos declarados y personalizados en historias de usuario (15 requerimientos)

## DESARROLLO

### 1. Control de requerimientos y criterios de aceptación

REQUERIMIENTO	HISTORIA DE USUARIO	CRITERIO DE ACEPTACIÓN	ESTADO ACTUAL
REQ-CL-1	Como administrador, quiero registrar y gestionar la información completa de los clientes (tipo, nombre/razón social, dirección, teléfono, correo y DNI/RUC), para evitar duplicación de datos y generar luego cotizaciones, órdenes e informes de manera precisa.	Permite registrar clientes con todos los campos obligatorios y valida formato de DNI/RUC.	Implementado
REQ-CL-2	Como administrador, quiero editar clientes existentes, manteniendo unicidad de DNI/RUC, para corregir o actualizar datos sin crear duplicados.	Permite editar un cliente sin duplicar su DNI/RUC.	Implementado
REQ-CL-3	Como administrador, quiero buscar clientes por nombre, DNI o RUC en tiempo real, para localizar rápidamente la ficha de un cliente.	Implementa buscador en tiempo real que filtra por nombre, DNI o RUC.	Implementado
REQ-SRV-1	Como administrador, quiero crear servicios con nombre, descripción, categoría, precio y plantilla de informe, para estructurar el catálogo de la empresa.	Permite crear servicios con todos los campos obligatorios.	Implementado
REQ-SRV-2	Como administrador, quiero validar que no se dupliquen servicios por nombre y categoría, para mantener integridad en el catálogo.	Bloquea creación de servicios con nombre y categoría ya existentes.	Implementado
REQ-SRV-3	Como administrador, quiero activar o desactivar servicios, para controlar su disponibilidad en ventas.	Permite cambiar estado activo/inactivo y refleja disponibilidad en el módulo de ventas.	Implementado
REQ-SRV-4	Como administrador, quiero mostrar solo servicios activos en el módulo de ventas, para evitar vender servicios deshabilitados.	El listado de servicios en ventas filtra únicamente aquellos activos.	En progreso
REQ-VD-1	Como administrador, quiero seleccionar un cliente existente para	El formulario de nueva venta muestra	Implementado

	realizar la venta, para asociar la transacción a su informe técnico.	selector de cliente con búsqueda.	
REQ-VD-2	Como administrador, quiero validar que el cliente tenga DNI/RUC correcto antes de la venta, para asegurar datos fiscales válidos.	Muestra error si el DNI/RUC no cumple formato.	Implementado
REQ-VD-3	Como administrador, quiero agregar servicios a la cotización posteriormente al informe técnico, recalculando el total, para preparar la venta correctamente.	Al añadir un servicio, actualiza el subtotal y total dinámicamente.	Implementado
REQ-VD-4	Como administrador, quiero que el sistema genere un código de venta y lo asocie al usuario que la realiza, para trazabilidad interna.	Asigna un ID único a la venta y registra el usuario.	En progreso
REQ-VD-5	Como administrador, quiero validar existencia de servicios y datos completos antes de guardar la venta, para evitar registros incompletos.	Impide guardar si hay campos vacíos o servicios no disponibles.	Pendiente
REQ-FAC-1	Como administrador, quiero emitir factura tras confirmación de cotización, para generar el informe legal.	Permite emitir factura solo si la cotización está confirmada; puede anularse en caso de impago.	Implementado
REQ-FAC-2	Como administrador, quiero validar que el cliente tenga DNI/RUC correcto antes de emitir factura, para asegurar datos fiscales válidos.	Muestra error si el DNI/RUC es inválido.	Implementado
REQ-FAC-3	Como administrador, quiero que se asigne número de factura conforme a la serie fiscal, para cumplir normativa.	Genera automáticamente el siguiente número en la serie fiscal configurada.	Implementado
REQ-FAC-4	Como administrador, quiero calcular subtotal, IGV y total automáticamente, para evitar errores de cálculo.	Muestra desglose de subtotal, IGV al 18% y total.	Implementado
REQ-FAC-5	Como administrador, quiero generar un PDF con formato fiscal y datos del cliente, para entregar el comprobante al cliente.	Crea y almacena PDF con encabezado fiscal, datos de cliente, ítems y totales.	En progreso
REQ-FAC-6	Como administrador, quiero registrar la factura con su ID, número y fecha, para mantener trazabilidad interna.	Almacena registro en la BD con ID, número, fecha y ruta del PDF.	Pendiente
REQ-FAC-7	Como administrador, quiero mostrar error si faltan datos fiscales al emitir factura, para evitar facturas incompletas.	Bloquea emisión y muestra mensaje de campos obligatorios faltantes.	Pendiente
REQ-TEC-1	Como administrador, quiero seleccionar plantilla de informe técnico adecuada, para documentar resultados de servicios.	Ofrece selector de plantilla según tipo de servicio.	Implementado
REQ-TEC-2	Como administrador, quiero autocompletar campos básicos (cliente, venta, fecha), para agilizar creación de informe.	Completa automáticamente datos de cliente, venta y fecha al cargar plantilla.	Implementado

REQ-TEC-4	Como administrador, quiero generar un PDF corporativo del informe, para compartirlo con el cliente.	Crea PDF con el estilo corporativo definido.	En progreso
REQ-TEC-5	Como administrador, quiero registrar metadatos del informe (idInforme, fecha, usuario, ruta PDF), para auditabilidad.	Guarda metadatos en BD al generar el informe.	Pendiente
REQ-TEC-6	Como administrador, quiero descargar o enviar por correo el informe, para compartirlo fácilmente.	Ofrece botón de descarga y opción de envío por SMTP.	Pendiente
REQ-CTR-2	Como administrador, quiero ver tabla "Servicios más vendidos", para identificar tendencias.	Muestra tabla ordenada por cantidad o importe.	Implementado
REQ-CTR-3	Como administrador, quiero filtros por fecha, cliente y servicio sin recarga, para explorar datos dinámicamente.	Filtra sin refrescar la página (AJAX).	En progreso
REQ-SEG-1	Como administrador, quiero que solo administradores accedan a ciertos módulos, para seguridad.	Control de roles; bloquea acceso a rutas si no es administrador.	Implementado

## 2. Relación implementación Java y requerimientos realizados

REQUERIMIENTO	HISTORIA DE USUARIO	NOMBRE ARCHIVO JAVA	ESTADO ACTUAL
REQ-CL-1...4	Gestión de Clientes: registrar/editar/buscar/historial de clientes.	Cliente.java	Ver tabla arriba
REQ-SRV-1...4	Gestión de Servicios: crear, validar duplicados, activar/desactivar y filtrar activos.	Servicio.java	Ver tabla arriba
REQ-VD-1...5	Venta Directa: seleccionar cliente, validar datos, agregar servicios, generar código, validar antes de guardar.	OrdenServicio.java	Ver tabla arriba
REQ-FAC-1...7	Emisión de Facturas: emisión condicional, validaciones, cálculos, generación y registro de PDF. (validación adicional por DNI/RUC → VerificacionFactura.java)	Factura.java	Ver tabla arriba
REQ-TEC-1...6	Generación de Informes Técnicos: plantillas, autocompletar, edición, PDF, metadatos, descarga/envío. (lógica de negocio de informe → InformeLegal.java)	EstructuralInforme.java	Ver tabla arriba
REQ-SEG-1...3	Seguridad y Control de Accesos: control de roles.	SeguridadService.java	Ver tabla arriba