

Formato de Propuesta de Proyecto Final

1. Integrantes del equipo

Cif	Nombre Completo	Rol
23020011	Emmanuel Leonardo Aguilar Novoa	Analista y Modelador UML
22020500	Diego Armando Urbina Aviles	Desarrollador de Dominio
23010363	Mauricio Javier Lacayo McConnell	Desarrollador de Interfaces OpenXava y Pruebas

2. Información general del proyecto

Título del proyecto	Sistema Integral de Gestión y Seguimiento de Paquetes con Facturación
Tipo de Sistema	Aplicación de gestión
Breve descripción (máximo 150 palabras)	Sistema web desarrollado con OpenXava que permite gestionar el ciclo completo de paquetería: registro de clientes, sucursales y paquetes, asignación a embarques, seguimiento de estado y ubicación, así como facturación y control de pagos. Cada paquete tiene un código de tracking, peso y tipo de envío (AÉREO/MARÍTIMO) desde los que se calcula el costo según una tabla de tarifas vigente. El sistema genera facturas con detalles por paquete, calcula impuestos, total y saldo pendiente en función de los pagos registrados. Además, permite registrar reclamos de clientes sobre paquetes específicos y llevar el historial de movimientos (cambios de estado y sucursal). Todo se organiza en entidades persistentes y vistas CRUD para facilitar reportes y consultas operativas y administrativas.

3. Definición del problema

Situación problemática	Las empresas de paquetería suelen manejar de forma dispersa la información de clientes, paquetes, embarques y facturación. Esto provoca problemas como: dificultad para conocer el estado real de cada envío, riesgo de cobrar dos veces o de olvidar facturar paquetes entregados, falta de control sobre tarifas vigentes y pagos pendientes, y poca trazabilidad de reclamos de los clientes. La ausencia de un
------------------------	--

	sistema integrado genera reprocesos, errores en la atención y poca visibilidad del negocio.
Justificación	Un sistema que unifique la gestión de paquetes, inventario en sucursales, embarques y facturación mejora la trazabilidad, reduce errores de cobro y facilita la atención al cliente. Al registrar todos los movimientos y pagos en un solo lugar, se pueden obtener reportes de ingresos, paquetes en tránsito, saldos pendientes y reclamos abiertos, lo que apoya la toma de decisiones y la transparencia del servicio.
Objetivo General	Diseñar e implementar un sistema integral que permita registrar y rastrear paquetes, administrar su movimiento entre sucursales y embarques, y emitir facturas con control de pagos y reclamos asociados.
Objetivos Específicos (mínimo 3)	<ol style="list-style-type: none"> 1. Gestionar información de clientes, sucursales, categorías y tarifas de envío (alta, baja, modificación y consulta). 2. Registrar paquetes con cliente, peso, tipo de envío, estado y ubicación (sucursal/ubicación de almacén), manteniendo el historial de movimientos. 3. Administrar embarques que agrupen paquetes entre sucursales, vinculados a un transportista y a un estado de envío. 4. Generar facturas por cliente con detalles por paquete, calcular subtotal, impuestos, total y saldo pendiente en función de los pagos.

4. Alcance del sistema

- CRUD de clientes, sucursales y categorías de servicio.
- Registro y gestión de paquetes con tracking, peso, tipo de envío, costo calculado, estado y banderas de “facturado”.
- Gestión de ubicaciones de almacén (pasillo, estante, nivel) por sucursal y asociación de paquetes a dichas ubicaciones.
- Gestión de embarques (origen, destino, fecha de salida, transportista, estado) y asociación de paquetes a cada embarque.
- Tabla de tarifas por tipo de envío y fecha de vigencia para calcular el costo por libra.
- Generación de facturas con detalles por paquete y cálculo automático de subtotal, impuesto e importe total.
- Registro de pagos por factura, con método de pago y referencia de transacción, y cálculo de saldo pendiente.

- Registro y seguimiento de reclamos de clientes sobre paquetes, con descripción del problema, estado y resolución.
- Historial de movimientos por paquete (fecha/hora, estado, sucursal, comentario).
- Búsqueda y consulta de paquetes, facturas y reclamos por código, cliente y estado, aprovechando las vistas de lista de OpenXava.

5. Actores y usuarios del sistema

Actor / Usuario	Descripción / Rol en el sistema
Administrador	Configura sucursales, categorías, tarifas, transportistas y usuarios; revisa reportes generales.
Operador de sucursal	Registra paquetes, actualiza estados y movimientos, asigna paquetes a embarques y ubicaciones.
Cajero / Facturador	Genera facturas, agrega detalles por paquete, registra pagos y consulta saldos pendientes.
Agente de atención	Registra y actualiza reclamos de clientes, consulta historial de paquetes y facturas.
Cliente/Consultante	Consulta el estado e historial de sus paquetes y, en su caso, datos básicos de facturas.

6. Requerimientos funcionales y no funcionales

Requerimientos funcionales:

- **RF1.** Registrar y actualizar clientes, incluyendo datos de identificación, contacto y dirección.
- **RF2.** Registrar y actualizar sucursales y ubicaciones de almacén (pasillo, estante, nivel).

- **RF3.** Registrar paquetes con tracking, cliente, sucursal actual, ubicación detallada, peso, tipo de envío y estado inicial.
- **RF4.** Calcular el costo de envío de un paquete en función del tipo de envío y la tarifa vigente (precio por libra).
- **RF5.** Registrar movimientos de paquetes (timestamp, estado, sucursal, comentario) y mostrar el historial ordenado.
- **RF6.** Gestionar embarques que agrupen paquetes entre sucursales, asociados a un transportista y a un estado de envío.
- **RF7.** Generar facturas por cliente con detalles de paquetes no facturados, calculando subtotal, impuesto y total.
- **RF8.** Registrar pagos de una factura (monto, fecha, método de pago y referencia) y calcular el saldo pendiente.
- **RF9.** Registrar y actualizar reclamos de clientes sobre paquetes, incluyendo estado del reclamo y resolución.
- **RF10.** Consultar y filtrar paquetes, facturas, embarques y reclamos por distintos criterios (cliente, estado, fechas, sucursal) y exportar listados usando las opciones de reporte de OpenXava.

Requerimientos no funcionales:

- **RNF1.** Implementar una arquitectura por capas (Dominio, Servicios/Lógica de negocio, Persistencia, Presentación) para favorecer mantenibilidad y pruebas.
- **RNF2.** Utilizar JPA/Hibernate y la infraestructura de OpenXava para la persistencia en una base de datos relacional.
- **RNF3.** Aplicar validaciones de negocio: transición coherente de estados de envío y reclamo, códigos de tracking únicos, no facturar dos veces el mismo paquete.
- **RNF4.** Mantener registro consistente de fechas y horas utilizando una misma zona horaria.
- **RNF5.** Proveer una interfaz web basada en módulos de OpenXava con listas, formularios y acciones estándar para reducir el tiempo de desarrollo.

7. Modelo inicial del dominio

Clases

- **Cliente**: nombreCompleto:String, cedula:String, telefono:String, email:String, direccion:String
- **Sucursal**: id:String, nombre:String, direccion:String, horario:String
- **Categoría**: id:String, nombre:String, descripcion:String
- **UbicacionAlmacen**: id:String, pasillo:String, estante:String, nivel:String, sucursal:Sucursal
- **Paquete**: id:String, trackingProveedor:String, descripcion:String, cliente:Cliente, ubicacionActual:Sucursal, ubicacionDetallada:UbicacionAlmacen, tipoEnvio:TipoEnvio, pesoLibras:BigDecimal, costoEnvio:BigDecimal, estado:EnvioEstado, embarque:Embarque, facturado:boolean, historial>List<Movimiento>
- **Movimiento**: timestamp:LocalDateTime, estado:EnvioEstado, sucursal:Sucursal, descripcion:String
- **Transportista**: id:String, nombre:String, codigoATA:String, contacto:String
- **Embarque**: id:String, numeroVueloContenedor:String, fechaSalida:LocalDate, origen:Sucursal, destino:Sucursal, estado:EnvioEstado, transportista:Transportista, paquetes:Collection<Paquete>
- **Tarifa**: id:String, tipoEnvio:TipoEnvio, fechaVigenciaInicio:LocalDate, fechaVigenciaFin:LocalDate, precioPorLibra:BigDecimal
- **Factura**: id:String, anio:int, numero:int, fecha:LocalDate, cliente:Cliente, detalles:Collection<DetalleFactura>, observaciones:String, pagada:boolean, pagos:Collection<Pago>
- **DetalleFactura**: paquete:Paquete, cantidad:int, precio:BigDecimal, importe:BigDecimal
- **Pago**: id:String, fechaPago:LocalDate, factura:Factura, monto:BigDecimal, metodo:MetodoPago, referenciaTransaccion:String
- **Reclamo**: id:String, fechaReclamo:LocalDate, cliente:Cliente, paquete:Paquete, descripcionProblema:String, estado:EstadoReclamo, resolucion:String

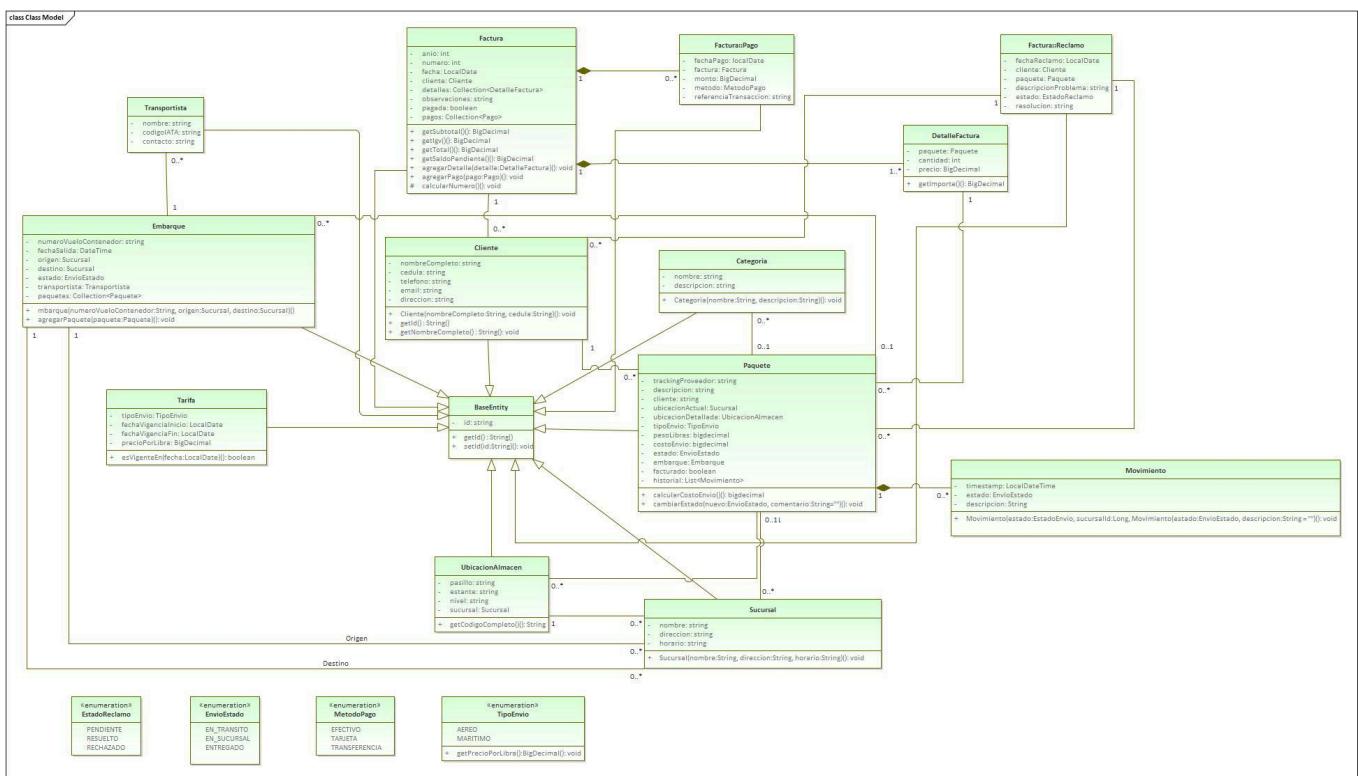
Enumeraciones

- **TipoEnvio**: AEREO, MARITIMO
- **EnvioEstado**: EN_TRANSITO, EN_SUCURSAL, ENTREGADO
- **EstadoReclamo**: PENDIENTE, RESUELTO, RECHAZADO

- **MetodoPago:** EFECTIVO, TARJETA, TRANSFERENCIA

Relaciones (resumen, opcional para texto):

- Cliente 1 — 0..* Paquete
- Cliente 1 — 0..* Factura
- Cliente 1 — 0..* Reclamo
- Sucursal 1 — 0..* UbicacionAlmacen
- UbicacionAlmacen 0..1 — 0..* Paquete
- Sucursal 1 — 0..* Paquete (como ubicaciónActual)
- Sucursal 1 — 0..* Embarque (origen)
- Sucursal 1 — 0..* Embarque (destino)
- Transportista 1 — 0..* Embarque
- Embarque 0..1 — 0..* Paquete
- Paquete 1 — 0..* Movimiento (composición)
- Tarifa (según TipoEnvio y fecha) se usa para calcular costo de Paquete (relación lógica)
- Factura 1 — 1..* DetalleFactura (composición)
- Paquete 1 — 0..* DetalleFactura
- Factura 1 — 0..* Pago (composición)
- Factura 1 — 0..* Reclamo (vía Paquete y Cliente, para atención al cliente)
- Reclamo 1 — 1 Cliente, 1 — 1 Paquete



8. URL del repositorio

<https://github.com/diegourbiaviles1/POO-proyecto.git>

9. Cronograma estimado de desarrollo

Etapa	Actividad	Semana #
Análisis	Definición de requerimientos y modelo de clases	11
Diseño	Estructura de entidades y vistas en OpenXava	12
Desarrollo	Implementación de módulos CRUD	13, 14
Pruebas	Validación y corrección de errores	15
Presentación	Exposición final y documentación	16

10. Recursos necesarios

- Github