

# ARQUITECTURA DE SOFTWARE

## 1RA INSTANCIA

### MICROSERVICIOS + SAGAS

#### Entregable:

En base al proyecto asignado los microservicios participantes del proceso principal asignado a cada grupo, debe presentarse las metas detalladas en la tabla inferior, aplicando al menos 3 patrones de Diseño con la combinación de 2 estilos de Arquitectura (MicroServicios + Hexagonal), debe mantenerse la aplicación de **Clean Code** y principios **SOLID**.

#### Rúbrica de Evaluación

Tarea	Puntaje	Observaciones
Login + Manejo de Sesiones + JWT (Json Web Token)	10	<ul style="list-style-type: none"><li>Login (1 Pts.)</li><li>Manejo de Sesiones (7 Pts.): Se controlará la bitácora de los usuarios con los CRUDs que realicen en el sistema, mismo que debe registrarse en los atributos de auditoría de las tablas en la BDD</li><li>JWT (2 Pts.)</li></ul>
UI Interactiva y Funcional para el proceso Principal (en Razor Pages) consumiendo diferentes microservicios <b>Ver Anexo 1</b>	25	<ul style="list-style-type: none"><li>Incluir al menos 2 buscadores para agregar los registros temporales previos a la Inserción (8 Pts.)</li><li>Debe permitir llamar de manera usable a otra UI en caso de no existir un registro necesario para la relación de la transacción (5 Pts.)</li><li>Debe permitir agregar, modificar, eliminar, editar, etc los registros que se enviarán en el microservicio que concentra la Base de Datos con la tabla Muchos a Muchos (9 Pts.)</li><li>Debe contener datos sumariados (2 Pts.)</li><li>Prever que cuando se quiera “Guardar” y generar la transacción debe pedirse un mensaje de confirmación (1 Pts.)</li></ul>
Transacción en Microservicios del Proceso Principal (ya definida para cada equipo).	15	Puede usarse sagas coreografiadas u orquestadas, se verificará el Funcionamiento adecuado de la saga (15 Pts.).
Generación de comprobante/recibo (reporte) de la transacción principal <b>Ver Anexo 2</b>	15	Es requisito que se encuentre desarrollado en un Microservicio Independiente exclusivo para reportes. <ul style="list-style-type: none"><li>Debe estar personalizado en base a la línea gráfica del proyecto, logo, colores, etc (3 Pts.)</li><li>Debe contener datos generales de la transacción, datos a detalle de la transacción,</li></ul>

		<p>datos sumariados, fecha y hora de generación, usuario que generó. (10 Pts.)</p> <ul style="list-style-type: none"> <li>• Debe generarse de manera automática una vez que se realizó la Inserción con la transacción Muchos a Muchos (2 Pts.)</li> </ul>
Operaciones adicionales a las transacciones realizadas con microservicios	15	<ul style="list-style-type: none"> <li>• Delete (2 Pts.): Solo anulación o baja del registro en el microservicio que contenga la tabla principal de la transacción</li> <li>• Select (3 Pts.) con refresco. Solo mostrar datos generales, 1 registro por transacción.</li> <li>• Ver detalles (10 Pts.): Seleccionado un registro en la UI (de los datos desplegados en el Select), abre un modal mostrando los datos completos de la transacción del Proceso Principal en modo solo lectura</li> </ul>
Defensa y presentación	20	Preguntas a cada miembro del grupo sobre patrones de diseño, estilos arquitectónicos, clean code, principios SOLID y código del proyecto
PLUS		Para aquellos equipos que quieran desarrollar un plus, será considerado como tal el Manejo de OutBox + Idempotencia hasta 10 Pts.

Cuidar la ortografía en la UI, por cada error ortográfico se penalizará con 1 punto

**IMPORTANTE:** Para confirmar la calificación, cada uno de los miembros del equipo debe generar la evidencia en el LMS subiendo de manera comprimida los microservicios desarrollados

**Fecha de presentación y Defensa: 18/12/2025**

## ANEXO 1

### EJEMPLO DE UI - INSERCIÓN PARA UNA VENTA

Para este ejemplo, se debe trabajar con varios microservicios:

- Cliente: Solo para elegir su PK y así generar la relación
- Venta: Para realizar la inserción tomando en cuenta que debe relacionarse con un cliente
- Producto: Para elegir los productos que participan de dicha venta y que luego actualizarán sus saldos
- DetalleVenta: La que relaciona los datos de la venta con el producto

The UI mockup is titled "REGISTRO DE VENTA" and includes the following elements and annotations:


- Header:** "Venta" (top left), window controls (top right).
- Date:** "Fecha: 23/06/2030" (labeled "Dato Automático No Editable").
- Client Search:** "CI/NIT Cliente" (labeled "Buscador de Clientes") with a search icon. A note states: "Si no se encuentra un cliente registrado, permite el registro de otro de manera usable".
- Client Label:** "Cliente/Razón Social" (labeled "Busca y agrega productos para la venta").
- Product Search:** "Producto a Agregar" (labeled "Busca y agrega productos para la venta").
- Product Table:**

Cantidad	Descripción	Precio Unitario Bs.	Importe Bs.	Opción
1	Lapiz Negro HB	5,00	5,00	-
20	Hojas Bond Tamaño Carta	0.50	10,00	-
2	Borrador de Tinta y de Carbón	8,00	16,00	-

(labeled "Permite edición de cantidades")
- Total:** "Total Bs: 31,00" (labeled "Quita un producto de la Venta antes de confirmar la misma").
- Action:** "Guardar" button (labeled "Registra la venta, donde se modifica al menos 3 tablas de microservicios diferentes. Genera el Recibo de manera Automática (abrirlo en otra pestaña)").
- Footer Note:** "Dato sumariado, se actualiza con la adición, edición o eliminación del 1 producto previo a la venta. Se puede agregar otros como ser: Efectivo Pagado, Cambio, etc".

## ANEXO 2

### EJEMPLO DE COMPROBANTE LUEGO DEL REGISTRO DE UNA VENTA

	<b>COMPROBANTE DE VENTA</b>		
Fecha: 23/06/2030			
CI/NIT: 4545451226      Razón Social: JUAN PEREZ VARGAS			
Cantidad	Descripción	Precio Unitario Bs.	Importe Bs.
1	Lapiz Negro HB	5,00	5,00
20	Hojas Bond Tamaño Carta	0.50	10,00
2	Borrador de Tinta y de Carbón	8,00	16,00
			<b>Total Bs: 31,00</b>
<b>Son Treinta y Un 00/100 Bolivianos</b>			
23/06/2030 - 10:15:45 - cajero1			