**Descripción de los servicios SOAP y REST**

**Introducción**

El diseño del sistema debe aprovechar las características específicas de **SOAP** y **REST** para optimizar las operaciones según su nivel de complejidad y necesidad de seguridad.

Las funcionalidades deben estar alineadas a los diferentes casos de uso y driver funcionales, puede realizar las respectivas modificaciones para cumplir el objetivo del proyecto.

**Servicios SOAP**

SOAP es ideal para operaciones transaccionales y complejas que requieren alta seguridad, estructura estricta y validaciones avanzadas.

***Funcionalidades***

1. **Gestión de Préstamos de Libros**
   * **Registrar Préstamo**
     + *Input*: ID del usuario, ID del libro, fecha de préstamo, fecha de devolución.
     + *Output*: Confirmación del préstamo (ID de la transacción, fecha y estado).
     + *Detalles*:
       - Validación de disponibilidad del libro.
       - Actualización del inventario (marcar como prestado).
       - Registro del historial de préstamos del usuario.
     + *Seguridad*: Transacción autenticada y encriptada.
   * **Registrar Devolución**
     + *Input*: ID de la transacción de préstamo.
     + *Output*: Confirmación de devolución y estado actualizado del inventario.
     + *Detalles*:
       - Actualizar el estado del libro a "disponible".
       - Calcular multas en caso de retraso (si aplica).
     + *Seguridad*: Validación del usuario y la transacción correspondiente.
2. **Generación de Reportes Detallados**
   * **Reporte de Préstamos Activos**
     + *Input*: Rango de fechas opcional.
     + *Output*: Listado de préstamos activos, incluyendo:
       - ID del préstamo, nombre del usuario, título del libro, fechas de préstamo y devolución, estado.
     + *Detalles*:
       - Soporte para filtros avanzados por usuario, fechas, categorías de libros, etc.
   * **Reporte de Historial de Usuarios**
     + *Input*: ID del usuario.
     + *Output*: Historial completo de préstamos y devoluciones del usuario, incluyendo multas.
     + *Detalles*:
       - Permite exportar los datos en formatos como PDF o Excel.
3. **Gestión de Usuarios**
   * **Registrar Usuario**
     + *Input*: Información del usuario (nombre, correo, tipo de usuario: estudiante/profesor).
     + *Output*: Confirmación del registro (ID del usuario generado).
     + *Detalles*:
       - Verificación de datos únicos como correo electrónico.
   * **Suspender Usuario**
     + *Input*: ID del usuario.
     + *Output*: Confirmación de suspensión y motivo.
     + *Detalles*:
       - Actualización del estado del usuario (suspendido).
       - Restricción inmediata de acceso a préstamos.

**Servicios REST**

REST es ideal para consultas rápidas, operaciones con menos requisitos de seguridad estricta y alta interacción con la interfaz de usuario.

**Funcionalidades**

1. **Consulta de Disponibilidad de Libros**
   * **Buscar Libro por ID o Título**
     + *Input*: ID o título del libro.
     + *Output*: Detalle del libro (disponibilidad, ubicación en la biblioteca, categoría).
     + *Endpoint*: GET /libros/{id} o GET /libros?titulo=nombre.
     + *Detalles*:
       - Respuesta optimizada para consumo rápido por aplicaciones cliente.
   * **Filtrar Libros por Categoría**
     + *Input*: Categoría o rango de publicación.
     + *Output*: Listado de libros filtrados por criterio.
     + *Endpoint*: GET /libros?categoria=categoria\_nombre.
2. **Gestión de Inventario de Libros**
   * **Listar Libros**
     + *Input*: Ninguno.
     + *Output*: Listado completo de libros con sus detalles básicos.
     + *Endpoint*: GET /libros.
   * **Actualizar Información del Libro**
     + *Input*: ID del libro, nuevos valores para los atributos (título, autor, categoría).
     + *Output*: Confirmación de la actualización.
     + *Endpoint*: PUT /libros/{id}.
3. **Gestión de Usuarios**
   * **Listar Usuarios**
     + *Input*: Ninguno.
     + *Output*: Listado básico de usuarios (ID, nombre, correo, tipo).
     + *Endpoint*: GET /usuarios.
   * **Actualizar Información del Usuario**
     + *Input*: ID del usuario, nuevos datos a actualizar (nombre, correo, etc.).
     + *Output*: Confirmación de la actualización.
     + *Endpoint*: PUT /usuarios/{id}.
4. **Consulta del Historial de Préstamos (Modo Simplificado)**
   * *Input*: ID del usuario.
   * *Output*: Listado resumido de préstamos (libro, fecha de préstamo, estado).
   * *Endpoint*: GET /usuarios/{id}/prestamos.
   * *Detalles*:
     + Optimizado para carga rápida en interfaces de usuario.