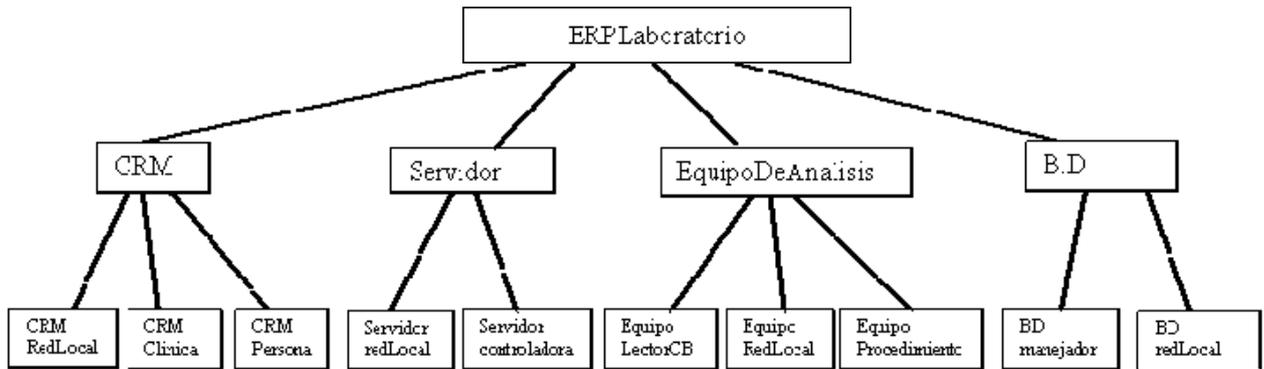


Punto I

**Para el planteamiento de la solución propuesta por ustedes:**

Requerimientos:

- Ingreso clientes:  
R1: registrar usuario en b.d.
  
- Clasificación y precio del examen:  
R2: clasificar y codificar tipos de examen (Según ministerio de protección social)
  
- Procedimiento examen:  
R3 enviar código de barras del cliente a la maquina.
  
- Resultados examen:  
R4: enviar los resultados del cliente de la maquina al servidor  
R5: actualizar los datos de los clientes.  
R6: entregar resultados.



Allocation:

	CRM	servidor	equipoAnalysis	B.D
R1	x	x		x
R2		x		x
R3		x	x	x
R4		x	x	
R5		x		x
R6	x	x		

1. Analice el particionamiento y asignación de requerimientos planteado, la propuesta de subsistemas y sub-subsistemas, en términos de **acoplamiento** (tip: revise cada subsistema y la razón por la cual tiene requerimientos asignados).
  
2. De acuerdo al análisis del punto anterior, replantee el diagrama de particionamiento, explicando cómo **mejora el acoplamiento**.

## Punto II

Para el siguiente “código a evaluar” planteado en su trabajo, escriba el **conjunto de casos de prueba** para **cubrimiento de código** (basado en tablas de decisión; distinga cada caso de prueba):

```
public static String agregarContacto(String loginUsuario, String loginContacto){
    Usuario contactoNuevo = buscarUsuarioLogin(loginContacto);
    Usuario usuario = buscarUsuarioLogin(loginUsuario);
    if(contactoNuevo != null){
        Usuario contacto = usuario.buscarContactoLogin(loginContacto);
        if(contacto == null){
            ObjetoServicio.agregarContacto(usuario,contactoNuevo);
            usuario.agregarContacto(contactoNuevo);
            return "El contacto se ha agregado exitosamente";

        }else{
            return "Usted ya tiene este contacto entre sus contactos!";
        }
    }else{
        return "El contacto que intenta agregar no existe";
    }
}
```