

ARQUITECTURA DEL SISTEMA: CAPA DE NEGOCIO

Práctica 3

Prácticas Ingeniería del Software

ETS Ingeniería Informática

DSIC – UPV

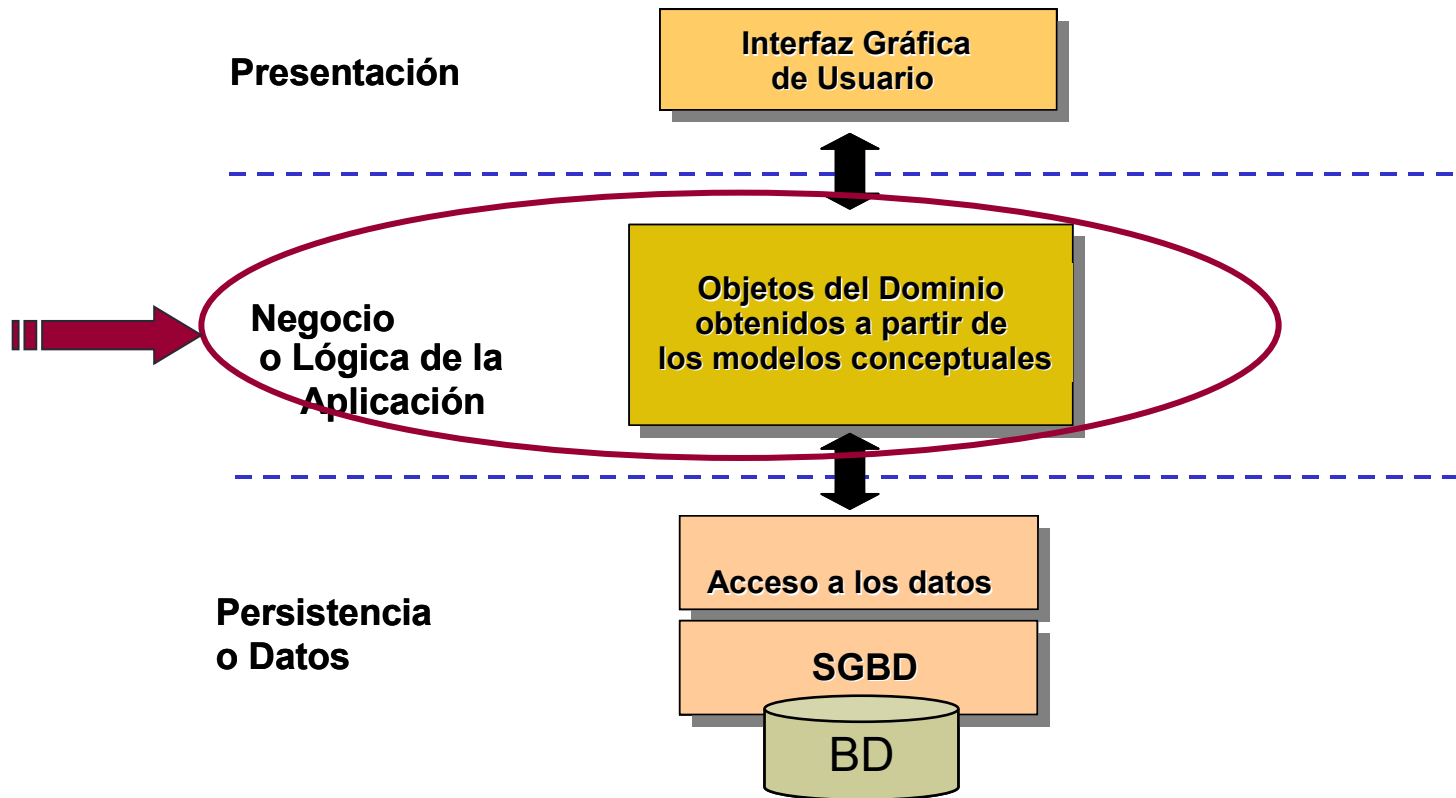
Curso 2014-2015

Índice

- Arquitectura del Sistema.
- Ejercicio. Caso de Estudio – Servicio Llamadas Emergencia
 - Modelo de Análisis.
 - Decisiones de diseño.
 - Modelo de Diseño.
- Utilidades de Implementación
 - Colecciones genéricas

Arquitectura del Sistema

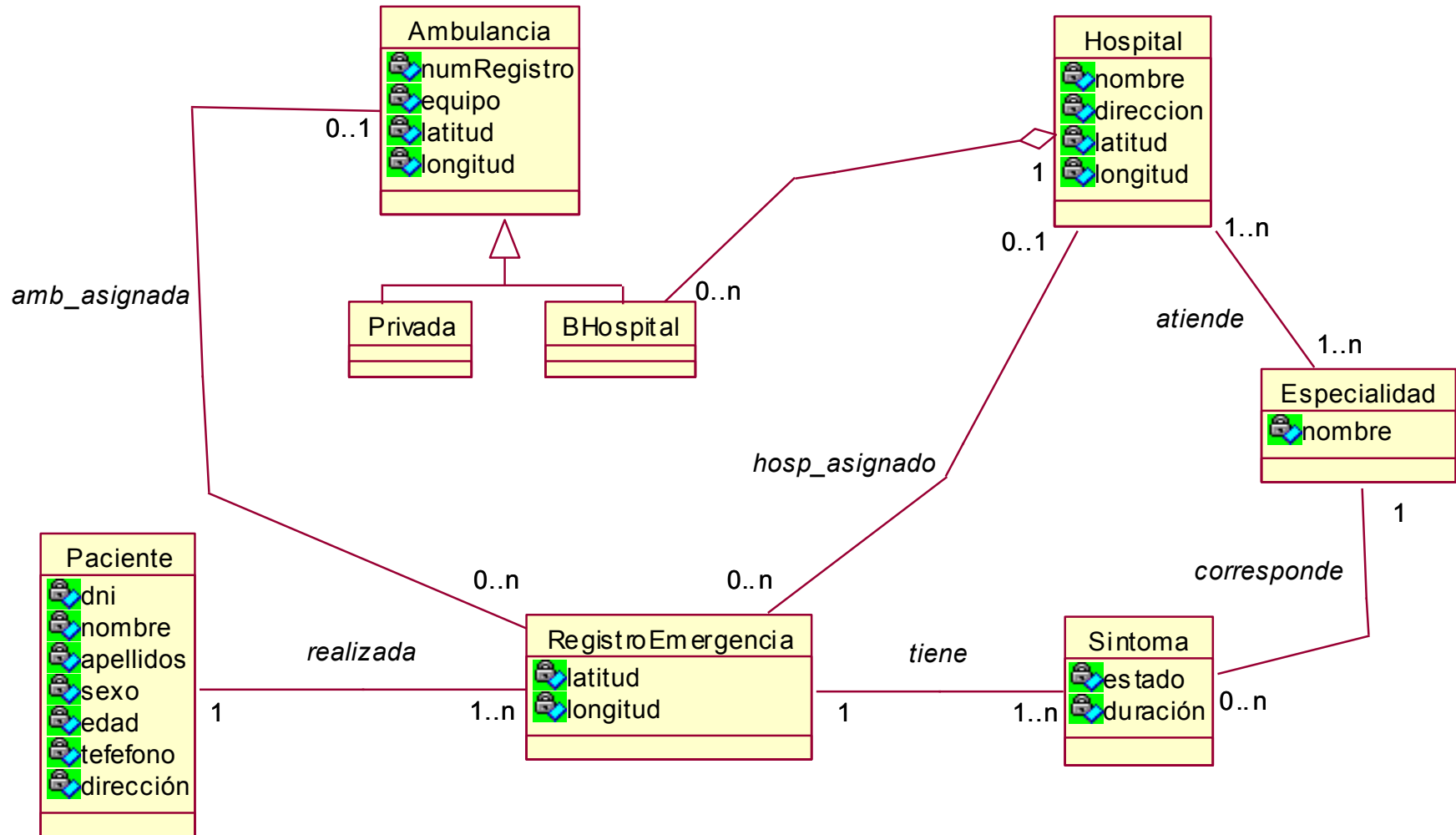
- Diseño arquitectónico en tres capas.



Ejercicio: Caso de Estudio – Servicio Llamadas Emergencia

- Generación del proyecto en Java. Concepto de aplicación.
- Realizar el diseño de objetos
 - Definición de las clases, atributos y métodos interfaz.
 - Relaciones entre clases. Diseño de asociaciones y agregaciones.
 - Selección de estructuras de datos adecuadas para manejar las colecciones.
 - Definición de métodos interfaz para las colecciones.
 - Aplicación de pautas de diseño.

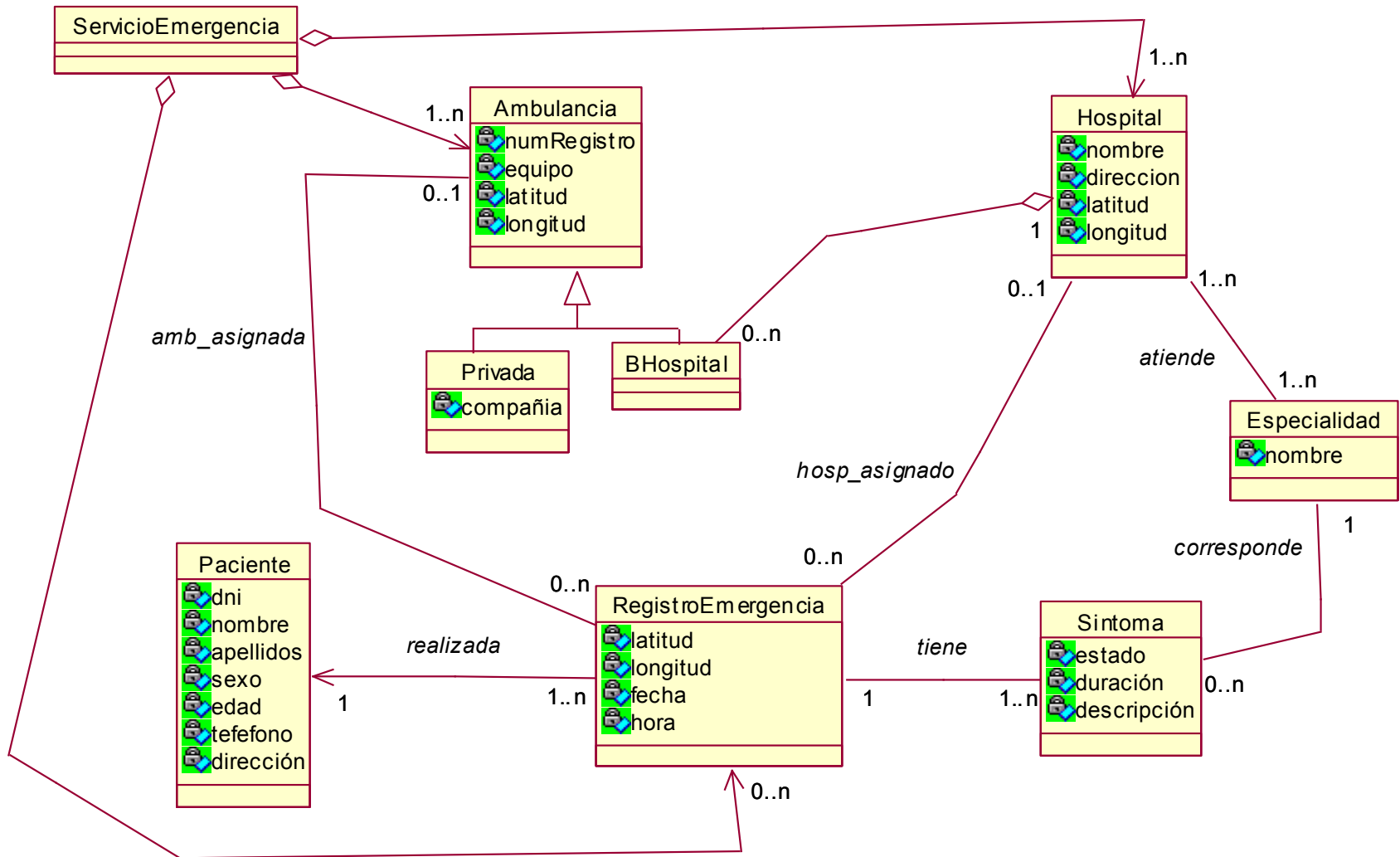
Modelo de análisis



Diseño de Objetos

- Para abordar el diseño de objetos se sugiere:
- Crear una nueva clase: **Clase ServicioEmergencia**
- Eliminar/Fusionar clases:
 - Se añade un atributo en la clase **Privada**, el atributo **compañía**, y no se elimina la clase
- Crear nuevas relaciones:
 - **ServicioEmergencia** se relaciona con **RegistroEmergencia**, **Ambulancia** y **Hospital**
- Modificar relaciones existentes:
 - Aplicar **restricciones de navegabilidad** en las relaciones de **ServicioEmergencia** con **RegistroEmergencia**, **Ambulancia** y **Hospital**
 - Aplicar restricciones de navegabilidad en la relación de **Paciente** y **RegistroEmergencia**
- Nuevos atributos:
 - Clase **RegistroEmergencia** --- atributos **fecha y hora**
 - Clase **Síntoma** ---- atributo **descripción**

Modelo de Diseño



Utilidades de Implementación. Colecciones genéricas

1. Crear un colección de cadenas

```
ArrayList<String> lista = new ArrayList<String>();
```

2. Añadir cadenas a la colección

```
lista.add("Primera");
```

```
Lista.add("Segunda");
```

3. Obtener el índice de una cadena

```
int index = lista.indexOf("Primera");
```

4. Recuperar una cadena según un índice

```
String elemento = lista.get(1);
```

5. Recorrer la lista de cadenas

```
for(String elemento: lista){  
    System.out.println("cadena: " + elemento);  
}
```

6. Borrar una cadena según un índice

```
lista.remove(1);
```

7. Reemplazar la cadena que ocupa una posición con otra cadena

```
lista.set(1, "Cuarta");
```