

Seminario

SeT5_2

Ingeniería del Software
ETS Ingeniería Informática
DSIC – UPV

Curso 2017-2018

Tema 5.

Diseño de la Lógica de la Aplicación

Ejercicios

Objetivos

- Diseño de Objetos- Trabajo en equipo (2 – 4 personas)
 - A partir de diagramas de clase

Ejercicios:

- Boletín de ejercicios Tema 3. Ejercicios: 19, 20, 21, 22, 23, 26, 28

Obtener el diseño en C# del diagrama de clases propuesto en los ejercicios anteriores, de acuerdo con las pautas de diseño vistas en clase.

Cuestión

UML dispone de dos medios para la representación de propiedades de las asociaciones: los atributos de enlace y las clases asociación. Explica cuál es la diferencia entre ambos, y cómo se podrían diseñar/implementar los atributos de enlace en asociaciones 1:1, 1:N y N:N.

Problema reingeniería (1/2).

Dado el siguiente diseño en C# se pide realizar un proceso de reingeniería y obtener el diagrama de clases UML que se corresponda con el diseño. Documentar las clases con los atributos y las relaciones con su nombre y roles (Nota: Los nombres de las relaciones aparecen en comentarios, los nombres de rol se corresponden con el nombre de los atributos que describen la relación, y las multiplicidades serán de 0..1 o 1..N).

```
class Persona {
    private String dni;
    private String Nombre;
    private String Apellidos;
    private String Direccion;
}

class Trabajador : Persona {
    private Empleo Empleado;
    //Trabaja_Para
}

class Cliente : Persona {
    private List<Pedido> Realiza; //Solicitar
    private List<Companyia> PideServicios; //Suministrar
}

class Companyia {
    private String CIF;
    private String Nombre;
    private List<Empleo> Emplea; //Trabaja_Para
    private List<Producto> crea; //Fabricar
    private List<Cliente> provee; //Suministrar
}

class Empleo {
    private double Salario;
    private double Horas_Extra;
    private Companyia Lacompañia; //Trabaja_Para
    private Trabajador ElTrabajador; //Trabaja_Para
    private Empleo Jefe; //Mandar
    private List<Trabajador> Trabajadores; //Mandar
}
```

Problema reingeniería (2/2).

```
class Pedido {
    private int Numero;
    private Date Fecha_Pedido;
    private double Precio;
    private Cliente Es;          //Solicitar
    private List<LineaPedido> SeCompone; //Componer
}

class LineaPedido {
    private int Numero;
    private int Cantidad;
    private double Precio_Linea;
    private Producto Pide;      //Pedir
}

class Producto {
    private intCodigo;
    private String Descripcion;
    private double Precio_Unidad;
    private Companyia Fabricado; //Fabricar
}
```

Ejercicio

Realizar el diseño en C# del siguiente diagrama de clases

