

SOLUCIÓN EJERCICIO 3

Diseño de clases y constructores

DOCENCIA VIRTUAL

Finalidad:

Prestación del servicio Público de educación superior (art. 1 LOU)

Responsable:

Universitat Politècnica de València.

Derechos de acceso, rectificación, supresión, portabilidad, limitación u oposición al tratamiento conforme a políticas de privacidad:

<http://www.upv.es/contenidos/DPD/>

Propiedad intelectual:

Uso exclusivo en el entorno de aula virtual.

Queda prohibida la difusión, distribución o divulgación de la grabación de las clases y particularmente su compartición en redes sociales o servicios dedicados a compartir apuntes.

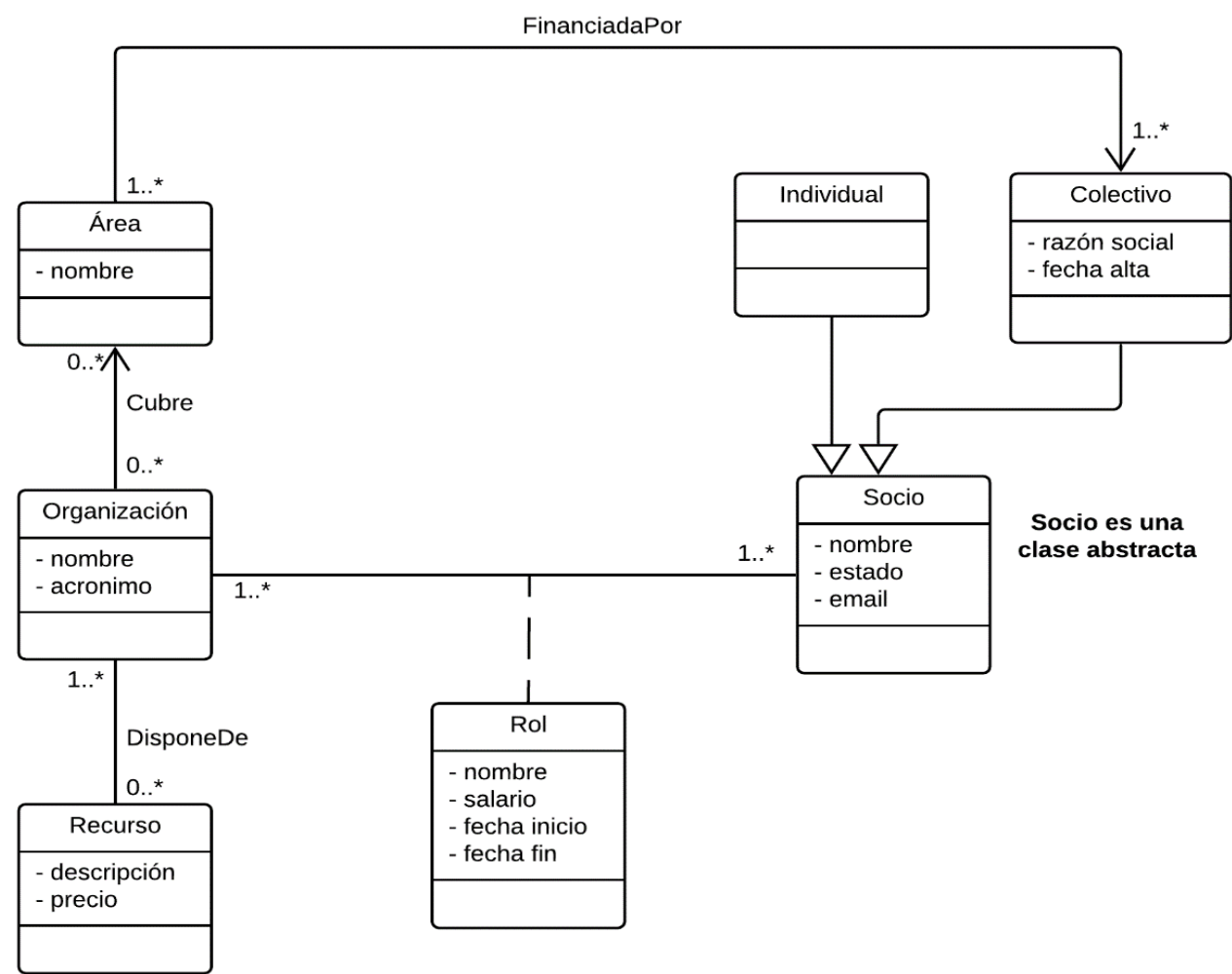
La infracción de esta prohibición puede generar responsabilidad disciplinaria, administrativa o civil



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Ejercicio 3



Ejercicio 3

```
public abstract class Socio
{
    private String nombre;
    private String estado;
    private String email;

    private ICollection<Rol> roles;

    //Se elimina del constructor el parámetro de tipo Rol
    //por la inconsistencia de cardinalidades mínimas
    //Se debe mantener por código para garantizar la consistencia del modelo
    2 referencias
    public Socio(string nombre, string estado, string email) //, Rol rol)
    {
        this.nombre = nombre;
        this.estado = estado;
        this.email = email;
        this.roles = new List<Rol>();
    }
}
```

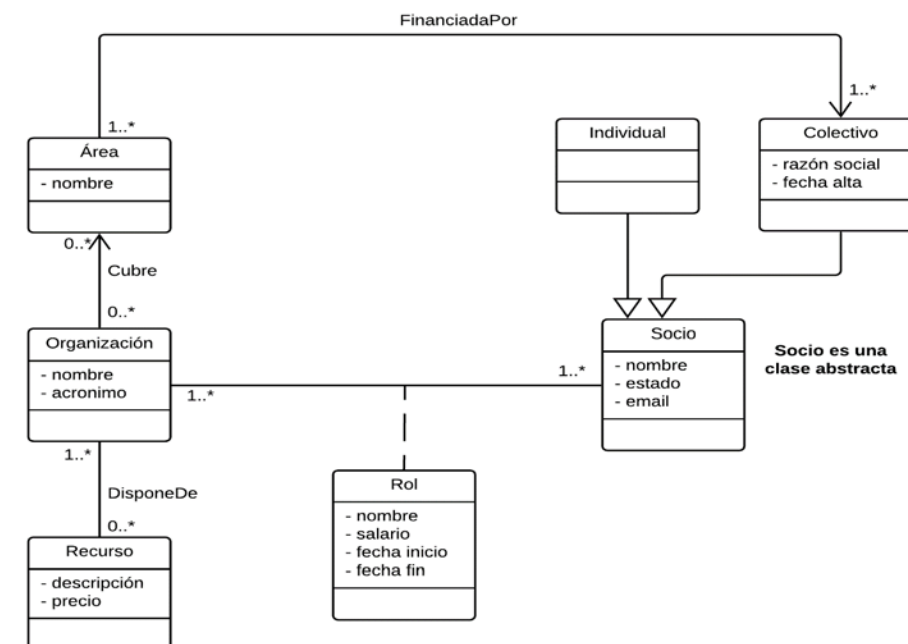
```
public class Colectivo : Socio
```

```
{
    private String razonSocial;
    private DateTime fechaAlta;

    //No se necesita por la restricción de navegación
    //private ICollection<Area> financiadaPor;
}
```

```
0 referencias
public Colectivo(string nombre, string estado, string email, string razonSocial, DateTime fechaAlta):base(nombre, estado, email)
{
    this.razonSocial = razonSocial;
    this.fechaAlta = fechaAlta;
}
```

```
public class Individual:Socio
{
    0 referencias
    public Individual(string nombre, string estado, string email)
        : base(nombre, estado, email)
    {
    }
}
```



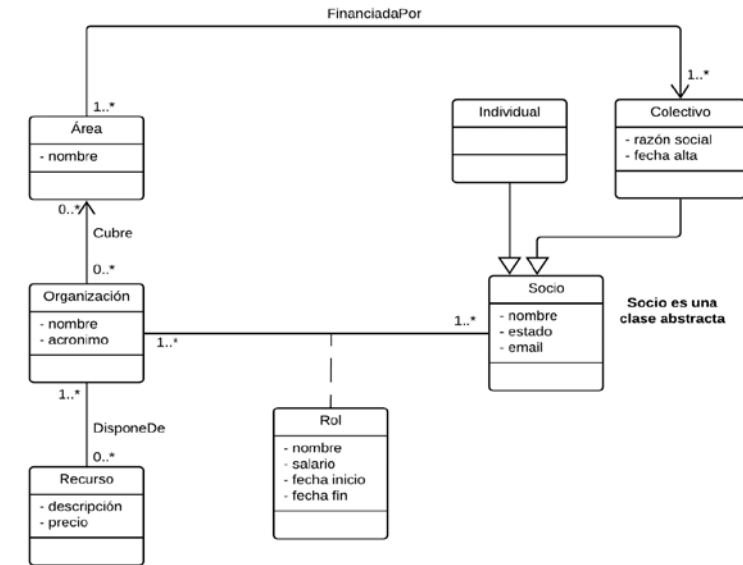
Ejercicio 3

```
public class Rol
{
    private String nombre;
    private double salario;
    private DateTime fechaInicio;
    private DateTime fechaFin;

    private Socio socio;
    private Organizacion organizacion;
```

0 referencias

```
public Rol(string nombre, double salario, DateTime fechaInicio, DateTime fechaFin, Socio socio, Organizacion organizacion)
{
    this.nombre = nombre;
    this.salario = salario;
    this.fechaInicio = fechaInicio;
    this.fechaFin = fechaFin;
    this.socio = socio;
    this.organizacion = organizacion;
}
```



Ejercicio 3

```

public class Organizacion
{
    private String nombre;
    private String acronimo;

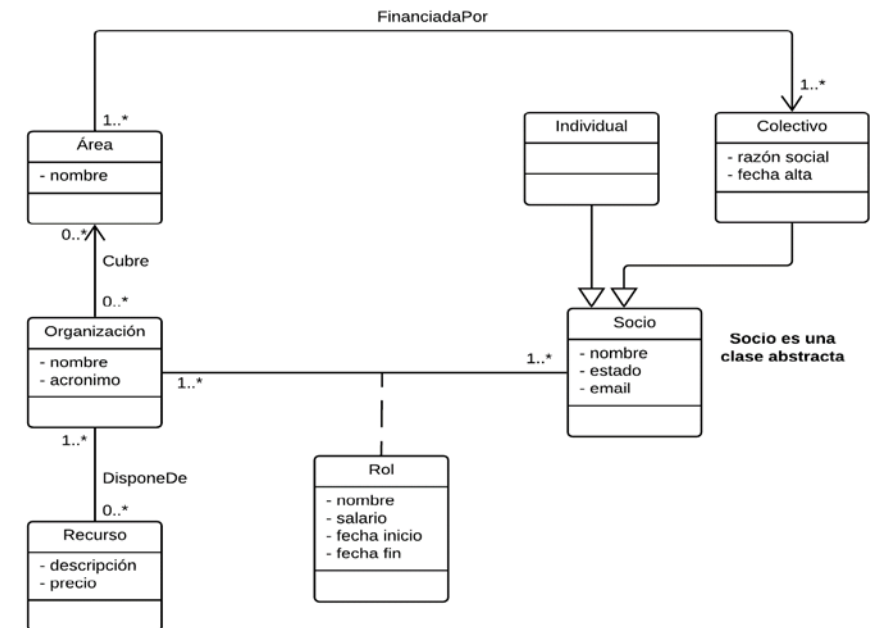
    private ICollection<Rol> roles;

    private ICollection<Recurso> disponeDe;

    private ICollection<Area> cubre;

    //Se elimina del constructor el parámetro de tipo Rol
    //por la inconsistencia de cardinalidades mínimas
    //Se debe mantener por código para garantizar la consistencia del modelo
    0 referencias
    public Organizacion(string nombre, string acronimo)//, Rol rol)
    {
        this.nombre = nombre;
        this.acronimo = acronimo;
        this.roles = new List<Rol>();
        this.disponeDe = new List<Recurso>();
        this.cubre = new List<Area>();
    }
}

```



Ejercicio 3

```
public class Area
{
    private String nombre;

    private ICollection<Colectivo> financiadasPor;
    //No se necesita por la restricción de navegación
    //private ICollection<Organizacion> cubre;

    0 referencias
    public Area(string nombre, Colectivo finaciada)
    {
        this.nombre = nombre;

        this.financiadasPor = new List<Colectivo>();
        this.financiadasPor.Add(finaciada);
    }
}
```

```
public class Recurso
{
    private String descripcion;
    private decimal precio;

    private ICollection<Organizacion> organizaciones;

    0 referencias
    public Recurso(string descripcion, decimal precio, Organizacion organizacion)
    {
        this.descripcion = descripcion;
        this.precio = precio;

        this.organizaciones = new List<Organizacion>();
        this.organizaciones.Add(organizacion);
    }
}
```

