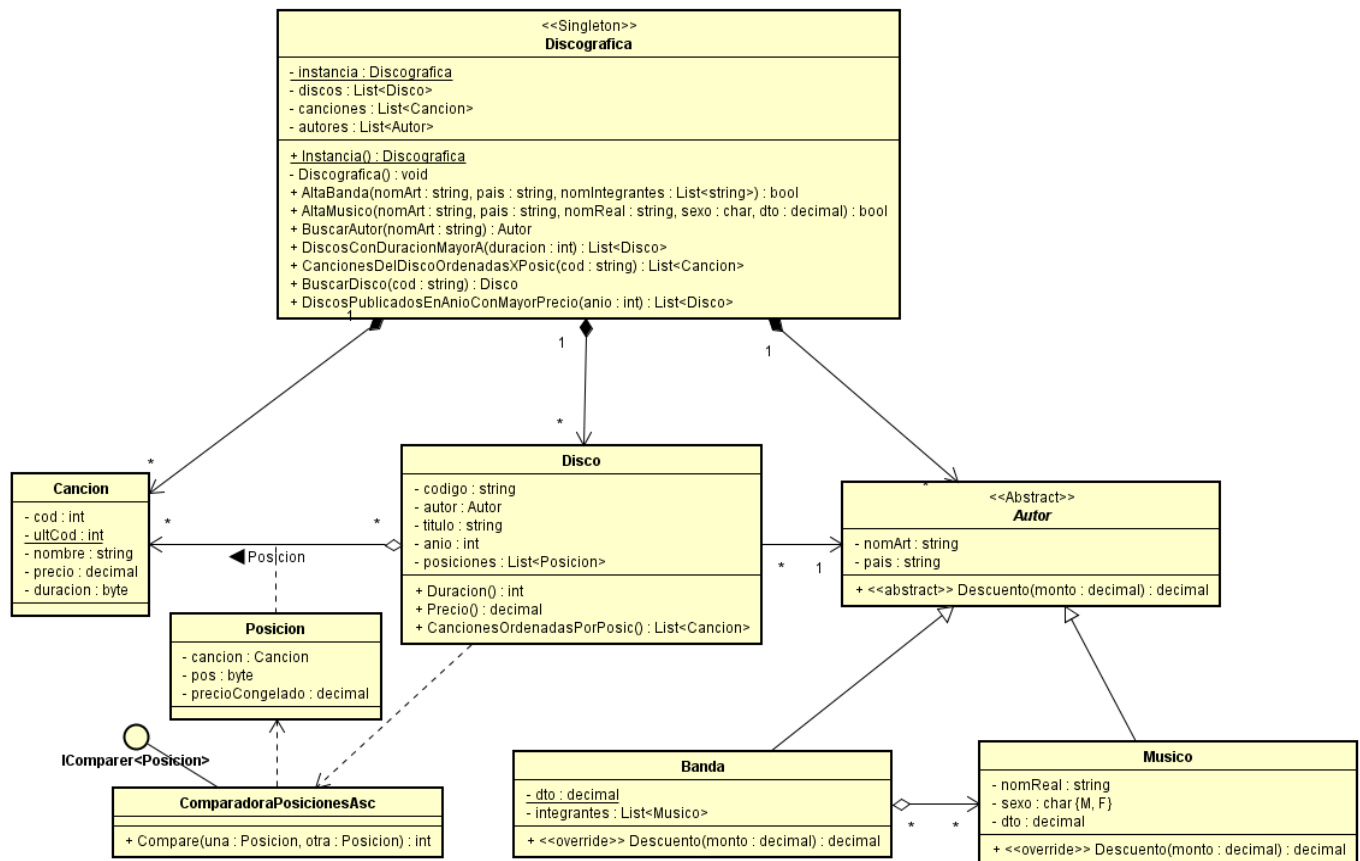


EVALUACION	SOLUCION EXAMEN	GRUPO	TODOS	FECHA	23/02/2016
MATERIA	PROGRAMACIÓN 2				
CARRERA	AP/ATI/APW				
CONDICIONES	<p>- Puntos: 100</p> <p>- Duración: 3 Horas</p> <p>- Sin material</p> <p>- Otros : <b>No escriba la hoja de la letra</b> Consultas solamente sobre interpretación de letra y sintaxis específica del lenguaje. Numerar las hojas entregadas. <b>Indicar nombre del docente del curso en primera hoja del examen</b></p>				

1. UML:



## 2. IMPLEMENTACIÓN:

b.

En Discográfica:

```
public List<Disco> DiscosConDuracionMayorA(int duracion)
{
    List<Disco> discos = new List<Disco>();

    foreach (Disco d in this.discos)
    {
        if (d.Duracion() > duracion)
        {
            discos.Add(d);
        }
    }

    return discos;
}
```

En Disco:

```
public int Duracion()
{
    int duracion = 0;

    foreach (Posicion p in this.posiciones)
    {
        duracion += p.Cancion.Duracion;
    }

    return duracion;
}
```

c.

En Discográfica:

```
public List<Cancion> CancionesDelDiscoOrdenadasXPosic(string cod)
{
    List<Cancion> lista = new List<Cancion>();
    Disco disco = this.BuscarDisco(cod);

    if (disco != null)
    {
        lista = disco.CancionesOrdenadasPorPosic();
    }

    return lista;
}
```

---

```
public Disco BuscarDisco(string cod)
{
    Disco aux = null;
    int i = 0;

    while (i < this.discos.Count && aux == null)
    {
        if (this.discos[i].Codigo == cod)
        {
            aux = this.discos[i];
        }

        i++;
    }

    return aux;
}

En Disco:
public List<Cancion> CancionesOrdenadasPorPosic()
{
    List<Cancion> canciones = new List<Cancion>();

    this.posiciones.Sort(new ComparadoraPosicionesAsc());

    foreach (Posicion p in this.posiciones)
    {
        canciones.Add(p.Cancion);
    }

    return canciones;
}

En ComparadoraPosicionesAsc:
public class ComparadoraPosicionesAsc : IComparer<Posicion>
{...
    public int Compare(Posicion una, Posicion otra)
    {
        return una.Pos - otra.Pos;
    }
}
```

d.

En Discográfica:

```
public List<Disco> DiscosPublicadosEnAnioConMayorPrecio(int anio)
{
    List<Disco> lista = new List<Disco>();
    decimal mayorPrecio = 0;

    foreach (Disco d in this.discos)
    {
        if (d.Anio == anio)
        {
            decimal precio = d.Precio();

            if (precio >= mayorPrecio)
            {
                if (precio > mayorPrecio)
                {
                    mayorPrecio = precio;
                    lista.Clear();
                }

                lista.Add(d);
            }
        }
    }

    return lista;
}
```

En Disco:

```
public decimal Precio()
{
    decimal precio = 0;

    foreach (Posicion p in this.posiciones)
    {
        precio += p.Cancion.Precio;
    }

    precio -= this.autor.descuento(precio);

    return precio;
}
```

En Autor:

```
public abstract class Autor
{ ...
    public abstract decimal descuento(decimal monto);
}
```

En Banda:

```
public class Banda : Autor
{...
    public override decimal descuento(decimal monto)
    {
        return (this.integrantes.Count > 4 ? dto : 0);
    }
}
```

En Musico:

```
public class Musico : Autor
{ ...
    public override decimal descuento(decimal monto)
    {
        return (this.sexo == 'M' ? dto : 0);
    }
}
```