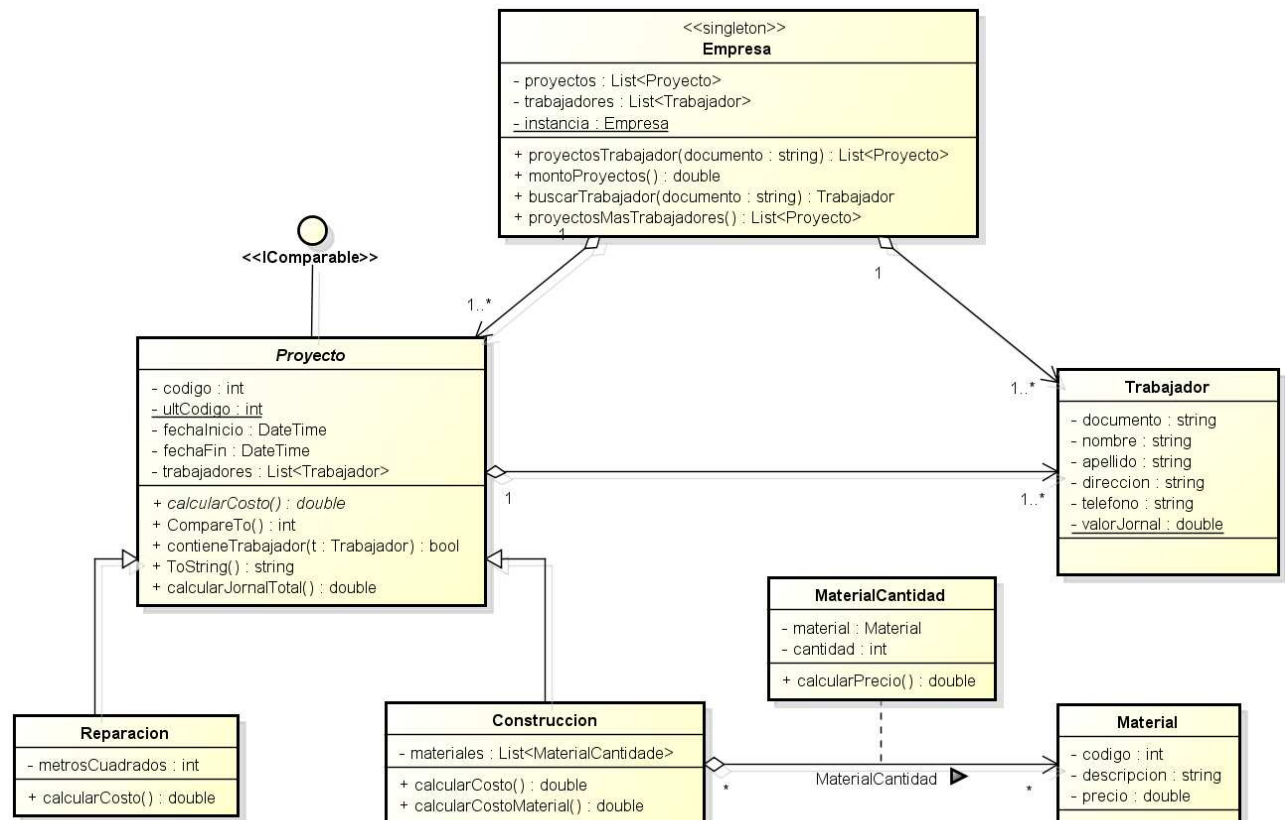


EVALUACION	SOLUCION EXAMEN	GRUPO	TODOS	FECHA	23/10/2015
MATERIA	PROGRAMACIÓN 2				
CARRERA	AP/ATI/APW				
CONDICIONES	<p>- Puntos: 100</p> <p>- Duración: 3 Horas</p> <p>- Sin material</p> <p>- Otros : No escriba la hoja de la letra</p> <p>Consultas solamente sobre interpretación de letra y sintaxis específica del lenguaje.</p> <p>Numerar las hojas entregadas.</p> <p>Indicar nombre del docente del curso en primera hoja del examen</p>				

Solución Examen Octubre 2015

Parte 1



powered by Astah

Parte 2

Requerimiento 1. Dado un documento de trabajador devolver todos los proyectos en los cuales ha participado ordenado por fecha de inicio en forma descendente en el siguiente formato.
Fecha de inicio – Fecha de fin –Costo del proyecto

Clase Empresa

```
public List<Proyecto> proyectosTrabajador(string documento)
{
    List<Proyecto> proyectosTrabajador = new List<Proyecto>();

    Trabajador t = buscarTrabajador(documento);
    if (t != null)
    {
        foreach (Proyecto p in this.proyectos)
        {
            if (p.contieneTrabajador(t))
            {
                proyectosTrabajador.Add(p);
            }
        }
        proyectosTrabajador.Sort();
    }
    return proyectosTrabajador;
}

public Trabajador buscarTrabajador(string documento)
{
    Trabajador t = null;
    int i = 0;
    bool bandera = false;
    while (i < this.trabajadores.Count && !bandera)
    {
        if (this.trabajadores[i].Documento == documento)
        {
            bandera = true;
            t = this.trabajadores[i];
        }
        i++;
    }
    return t;
}
```

Clase Proyecto

```
public bool contieneTrabajador(Trabajador t)
{
    return this.trabajadores.Contains(t);
}
```

```
public int CompareTo(Proyecto p)
{
    return this.fechaInicio.CompareTo(p.fechaInicio);
}

public override string ToString()
{
    return this.fechaInicio + "-" + this.fechaFin + "-" + this.calcularCosto();
}

public double calcularJornalTotal()
{
    double costo = 0;
    int cantidadJornal = this.trabajadores.Count;
    int dias = this.fechaFin.Subtract(this.fechaInicio).Days;
    costo = dias * cantidadJornal * Trabajador.ValorJornal;
    return costo;
}

public abstract double calcularCosto();
```

Clase Reparacion

```
public override double calcularCosto()
{
    double precio = 0;
    precio = calcularJornalTotal();
    precio = precio * 1.10;
    return precio;
}
```

Clase Construccion

```
public override double calcularCosto()
{
    double costo = 0;
    costo = calcularJornalTotal();
    if (this.materiales.Count > 0)
    {
        costo = costo + calcularCostoMaterial();
    }
    return costo;
}

public double calcularCostoMaterial()
{
    double costo = 0;
    foreach (MaterialCantidad materialCantidad in this.materiales)
    {
        costo = costo + materialCantidad.calcularPrecio();
    }
    return costo;
}
```

Clase MaterialCantidad

```
public double calcularPrecio()
{
    double precio = this.material.Precio * this.cantidad;
    return precio;
}
```

Requerimiento 2. Calcular el monto recaudado por la empresa por todos los proyectos que comenzaron en el presente año

Clase Empresa

```
public double montoProyectos()
{
    double monto = 0;
    int anioActual=DateTime.Now.Year;
    foreach (Proyecto p in this.proyectos)
    {
        if (p.FechaInicio.Year == anioActual)
        {
            monto = monto + p.calcularCosto();
        }
    }
    return monto;
}
```

Requerimiento 3. Retornar todos los proyectos en los cuales han participado la mayor cantidad de trabajadores

Clase Empresa

```
public List<Proyecto> proyectosMasTrabajadores()
{
    List<Proyecto> proyectosMasTrabajadores = new List<Proyecto>();
    int cantidadMayor = 0;
    foreach (Proyecto p in this.proyectos)
    {
        int cantidadTrabajadores=p.Trabajadores.Count;
        if (cantidadTrabajadores > cantidadMayor)
        {
            cantidadMayor = cantidadTrabajadores;
            proyectosMasTrabajadores.Clear();
            proyectosMasTrabajadores.Add(p);
        }
        else if (cantidadTrabajadores == cantidadMayor)
        {
            proyectosMasTrabajadores.Add(p);
        }
    }
    return proyectosMasTrabajadores;
}
```