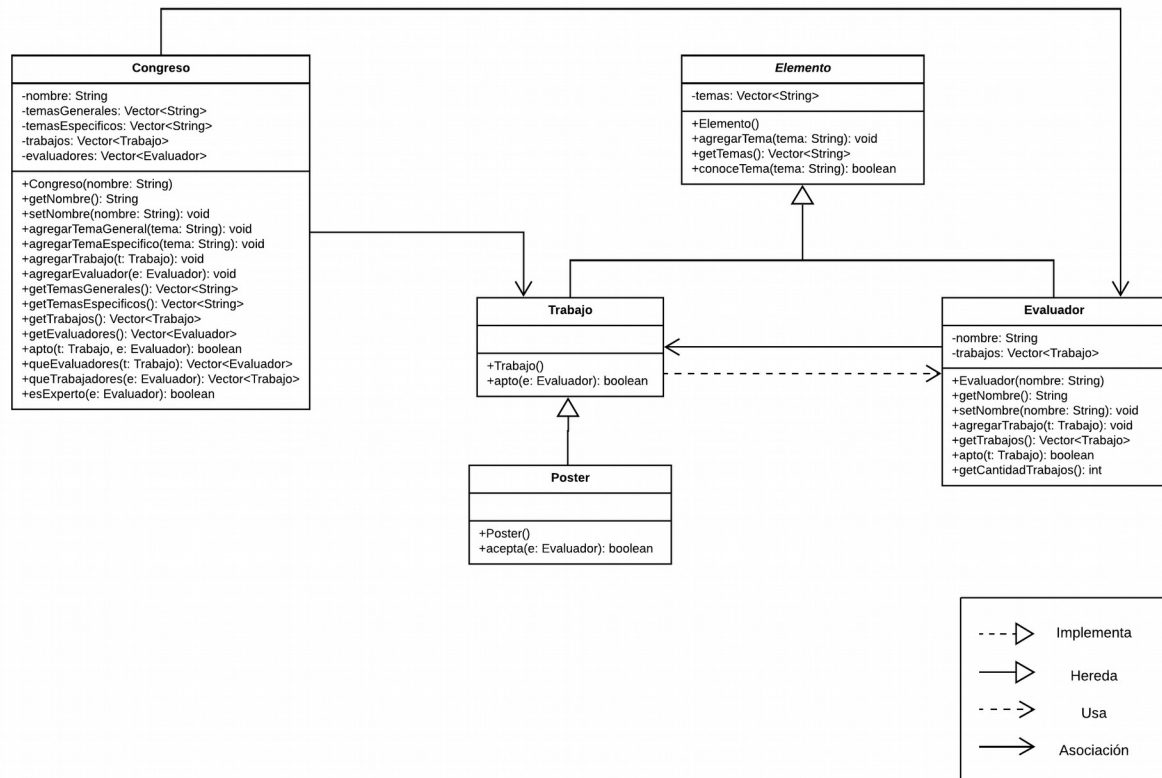


# Resolución Congreso Ciencias de la Computación

Hechos 16.31: Cree en el Señor Jesucristo, y serás salvo, tú y tu casa.

Diagrama de clases Congreso de Ciencias de la Computación



- Palabras, tipos y métodos reservados de Java.
- Clases creadas para la resolución.

```
public abstract class Elemento {
    private Vector<String> temas;
    public Elemento(){
        this.temas = new Vector<String>();
    }
    public void agregarTema(String tema){
        if(!this.temas.contains(tema)){
            this.temas.add(tema);
        }
    }
    public Vector<String> getTemas(){
        Vector<String> retorno = new Vector<String>();
        for(int i = 0; i < this.temas.size(); i++){
            retorno.add(this.temas.elementAt(i));
        }
        return retorno;
    }
    public boolean conoceTema(String tema){
        return this.temas.contains(tema);
    }
}
```

```
}  
}
```

```
public class Trabajo extends Elemento {  
    public Trabajo(){  
        super();  
    }  
    public boolean apto(Evaluador e){  
        Vector<String> temas = this.getTemas();  
        for(int i = 0; i < temas.size(); i++){  
            if(!e.conoceTema(temas.elementAt(i))){  
                return false;  
            }  
        }  
        return true;  
    }  
}
```

```
public class Poster extends Trabajo {  
    public Poster(){  
        super();  
    }  
    public boolean apto(Evaluador e){  
        Vector<String> temas = this.getTemas();  
        for(int i = 0; i < temas.size(); i++){  
            if(e.conoceTema(temas.elementAt(i))){  
                return true;  
            }  
        }  
        return false;  
    }  
}
```

```
public class Evaluador extends Elemento {  
    private String nombre;  
    private Vector<Trabajo> trabajos;  
    public Evaluador(String nombre){  
        super();  
        this.nombre = nombre;  
    }  
    public String getNombre() {  
        return this.nombre;  
    }  
    public void setNombre(String nombre) {  
        this.nombre = nombre;  
    }  
    public void agregarTrabajo(Trabajo t){  
        if(this.aptos(t)){  
            this.trabajos.add(t);  
        }  
    }  
    public Vector<Trabajo> getTrabajos() {
```

```

        Vector<Trabajo> retorno = new Vector<Trabajo>();
        for(int i = 0; i < this.trabajos.size(); i++){
            retorno.add(this.trabajos.elementAt(i));
        }
        return retorno;
    }
    public boolean apto(Trabajo t){
        return t.apto(this);
    }
    public int getCantidadTrabajos(){
        return this.trabajos.size();
    }
}

```

```

public class Congreso {
    private String nombre;
    private Vector<String> temasGenerales;
    private Vector<String> temasEspecificos;
    private Vector<Trabajo> trabajos;
    private Vector<Evaluador> evaluadores;
    public Congreso(String nombre){
        this.nombre = nombre;
        this.temasGenerales = new Vector<String>();
        this.temasEspecificos = new Vector<String>();
        this.trabajos = new Vector<Trabajo>();
        this.evaluadores = new Vector<Evaluador>();
    }
    public String getNombre() {
        return this.nombre;
    }
    public void setNombre(String nombre) {
        this.nombre = nombre;
    }
    public void agregarTemaGeneral(String tema){
        if(!this.temasGenerales.contains(tema)){
            this.temasGenerales.add(tema);
        }
    }
    public void agregarTemaEspecifico(String tema){
        if(!this.temasEspecificos.contains(tema)){
            this.temasEspecificos.add(tema);
        }
    }
    public void agregarTrabajo(Trabajo t){
        this.trabajos.add(t);
    }
    public void agregarEvaluador(Evaluador e){
        this.evaluadores.add(e);
    }
    public Vector<String> getTemasGenerales(){
        Vector<String> retorno = new Vector<String>();
        for(int i = 0; i < this.temasGenerales.size(); i++){

```

```

        retorno.add(this.temasGenerales.elementAt(i));
    }
    return retorno;
}
public Vector<String> getTemasEspecificos(){
    Vector<String> retorno = new Vector<String>();
    for(int i = 0; i < this.temasEspecificos.size(); i++){
        retorno.add(this.temasEspecificos.elementAt(i));
    }
    return retorno;
}
public Vector<Trabajo> getTrabajos() {
    Vector<Trabajo> retorno = new Vector<Trabajo>();
    for(int i = 0; i < this.trabajos.size(); i++){
        retorno.add(this.trabajos.elementAt(i));
    }
    return retorno;
}
public Vector<Evaluador> getEvaluadores() {
    Vector<Evaluador> retorno = new Vector<Evaluador>();
    for(int i = 0; i < this.evaluadores.size(); i++){
        retorno.add(this.evaluadores.elementAt(i));
    }
    return retorno;
}
public boolean apto(Trabajo t, Evaluador e){
    return t.apto(e);
}
public Vector<Evaluador> queEvaluadores(Trabajo t){
    Vector<Evaluador> retorno = new Vector<Evaluador>();
    for(int i = 0; i < this.evaluadores.size(); i++){
        if(t.apto(this.evaluadores.elementAt(i))){
            retorno.add(this.evaluadores.elementAt(i));
        }
    }
    return retorno;
}
public Vector<Trabajo> queTrabajadores(Evaluador e){
    Vector<Trabajo> retorno = new Vector<Trabajo>();
    for(int i = 0; i < this.trabajos.size(); i++){
        if(e.apto(this.trabajos.elementAt(i))){
            retorno.add(this.trabajos.elementAt(i));
        }
    }
    return retorno;
}
public boolean esExperto(Evaluador e){
    for(int i = 0; i < this.temasEspecificos.size(); i++){
        if(e.conoceTema(this.temasEspecificos.elementAt(i))){
            return true;
        }
    }
}

```

```
    return false;  
  }  
}
```