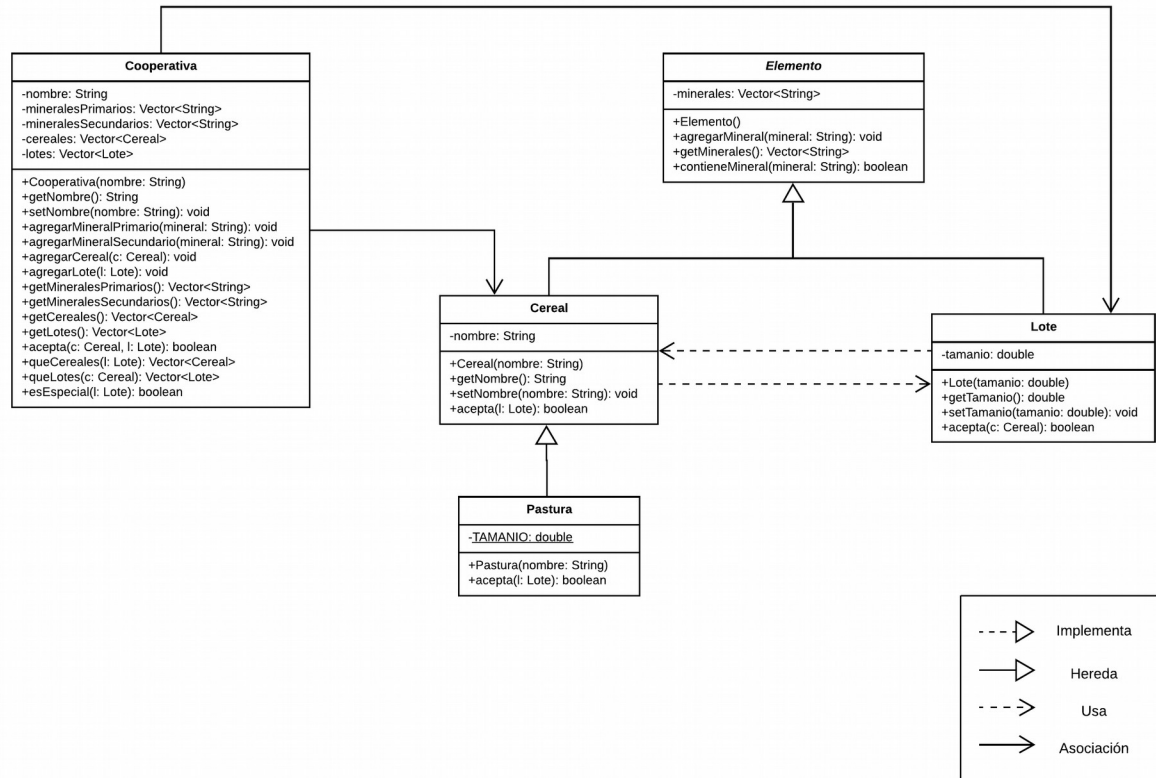


Resolución Sistema de Siembra

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

Diagrama de clases Sistema de Siembra



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

```
public abstract class Elemento {
    private Vector<String> minerales;
    public Elemento(){
        this.minerales = new Vector<String>();
    }
    public void agregarMineral(String mineral){
        if(!this.minerales.contains(mineral)){
            this.minerales.add(mineral);
        }
    }
    public Vector<String> getMinerales(){
        Vector<String> retorno = new Vector<String>();
        for(int i = 0; i < this.minerales.size(); i++){
            retorno.add(this.minerales.elementAt(i));
        }
        return retorno;
    }
    public boolean contieneMineral(String mineral){
        return this.minerales.contains(mineral);
    }
}
```

```
}  
}
```

```
public class Cereal extends Elemento {  
    private String nombre;  
    public Cereal(String nombre){  
        super();  
        this.nombre = nombre;  
    }  
    public String getNombre() {  
        return this.nombre;  
    }  
    public void setNombre(String nombre) {  
        this.nombre = nombre;  
    }  
    public boolean acepta(Lote l){  
        Vector<String> minerales = this.getMinerales();  
        for(int i = 0; i < minerales.size(); i++){  
            if(!l.contieneMineral(minerales.elementAt(i))){  
                return false;  
            }  
        }  
        return true;  
    }  
}
```

```
public class Pastura extends Cereal {  
    private static final double TAMANIO = 50.0;  
    public Pastura(String nombre){  
        super(nombre);  
    }  
    public boolean acepta(Lote l){  
        Vector<String> minerales = this.getMinerales();  
        for(int i = 0; i < minerales.size(); i++){  
            if(!l.contieneMineral(minerales.elementAt(i))){  
                return false;  
            }  
        }  
        return super.acepta(l) && !(l.getTamanio() < TAMANIO);  
    }  
}
```

```
public class Lote extends Elemento {  
    private double tamanio;  
    public Lote(double tamanio){  
        super();  
        this.tamanio = tamanio;  
    }  
    public double getTamanio() {  
        return this.tamanio;  
    }  
    public void setTamanio(double tamanio) {
```

```

        this.tamano = tamano;
    }
    public boolean acepta(Cereal c){
        return c.acepta(this);
    }
}

public class Cooperativa {
    private String nombre;
    private Vector<String> mineralesPrimarios;
    private Vector<String> mineralesSecundarios;
    private Vector<Cereal> cereales;
    private Vector<Lote> lotes;
    public Cooperativa(String nombre){
        this.nombre = nombre;
        this.mineralesPrimarios = new Vector<String>();
        this.mineralesSecundarios = new Vector<String>();
        this.cereales = new Vector<Cereal>();
        this.lotes = new Vector<Lote>();
    }
    public String getNombre() {
        return this.nombre;
    }
    public void setNombre(String nombre) {
        this.nombre = nombre;
    }
    public void agregarMineralPrimario(String mineral){
        if(!this.mineralesPrimarios.contains(mineral)){
            this.mineralesPrimarios.add(mineral);
        }
    }
    public void agregarMineralSecundario(String mineral){
        if(!this.mineralesSecundarios.contains(mineral)){
            this.mineralesSecundarios.add(mineral);
        }
    }
    public void agregarCereal(Cereal c){
        this.cereales.add(c);
    }
    public void agregarLote(Lote l){
        this.lotes.add(l);
    }
    public Vector<String> getMineralesPrimarios(){
        Vector<String> retorno = new Vector<String>();
        for(int i = 0; i < this.mineralesSecundarios.size(); i++){
            retorno.add(this.mineralesSecundarios.elementAt(i));
        }
        return retorno;
    }
    public Vector<String> getMineralesSecundarios(){
        Vector<String> retorno = new Vector<String>();
        for(int i = 0; i < this.mineralesSecundarios.size(); i++){

```

```

        retorno.add(this.mineralesSecundarios.elementAt(i));
    }
    return retorno;
}
public Vector<Cereal> getCereales(){
    Vector<Cereal> retorno = new Vector<Cereal>();
    for(int i = 0; i < this.cereales.size(); i++){
        retorno.add(this.cereales.elementAt(i));
    }
    return retorno;
}
public Vector<Lote> getLotes(){
    Vector<Lote> retorno = new Vector<Lote>();
    for(int i = 0; i < this.lotes.size(); i++){
        retorno.add(this.lotes.elementAt(i));
    }
    return retorno;
}
public boolean acepta(Cereal c, Lote l){
    return c.acepta(l);
}
public Vector<Cereal> queCereales(Lote l){
    Vector<Cereal> retorno = new Vector<Cereal>();
    for(int i = 0; i < this.cereales.size(); i++){
        if(l.acepta(this.cereales.elementAt(i))){
            retorno.add(this.cereales.elementAt(i));
        }
    }
    return retorno;
}
public Vector<Lote> queLotes(Cereal c){
    Vector<Lote> retorno = new Vector<Lote>();
    for(int i = 0; i < this.lotes.size(); i++){
        if(c.acepta(this.lotes.elementAt(i))){
            retorno.add(this.lotes.elementAt(i));
        }
    }
    return retorno;
}
public boolean esEspecial(Lote l){
    for(int i = 0; i < this.mineralesPrimarios.size(); i++){
        if(l.contieneMineral(this.mineralesPrimarios.elementAt(i))){
            return true;
        }
    }
    return false;
}
}

```