# Ayudantía 5

Alexander Inostroza – alexander.inostroza@usm.cl

La **encapsulación** es un principio fundamental de la programación orientada a objetos y consiste en ocultar el estado interno del objeto y obligar a que toda interacción se realice a través de los métodos del objeto.

### Ejemplo

Para una clase **Ladrón**, que le roba a una clase **Víctima,**

```
public class Ladron {
    int dinero;
    public void robar(Victima victima){
        this.dinero = victima.dinero;
        victima.dinero = 0;
    }
}
public class Victima {
    int dinero;
}
```

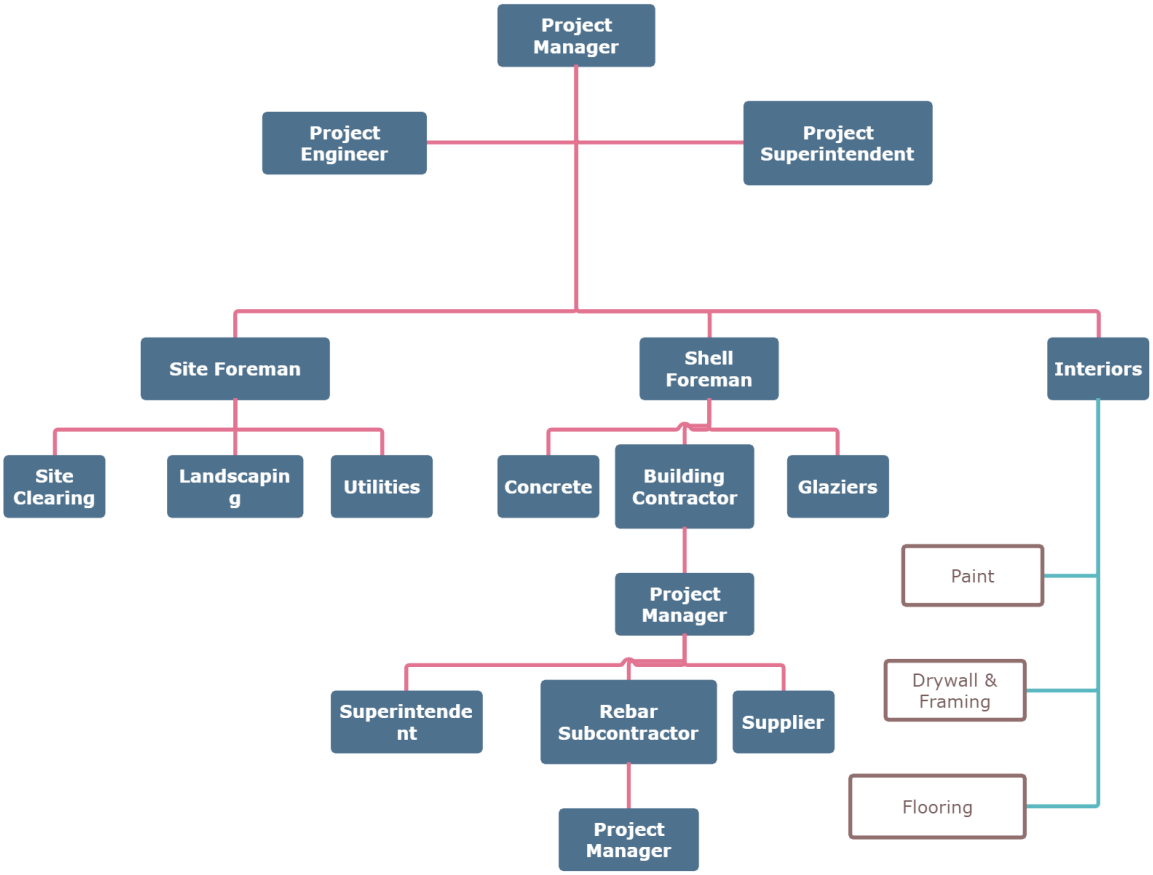
Para aplicar encapsulamiento, cambiaríamos el código a:

```
public class Ladron {
    int dinero;
    public void robar(Victima victima){
        this.dinero = victima.entregarDinero();
    }
}
public class Victima {
    int dinero;
    public int entregarDinero(){
        int r = dinero;
        this.dinero = 0;
        return r;
    }
}
```

1. **Combate**
   Modifique el siguiente código aplicando el principio de encapsulamiento.

```java
public class LuchadorA {

    int hp;
    int dmg;

    public LuchadorA(){
        hp = 500;
        dmg = 50;
    }

    public void golpear(LuchadorB luchadorB){
        luchadorB.hp = luchadorB.hp - 50;
        if (luchadorB.hp <= 0){
            System.out.println("Gana luchador A");
        }
    }
}

public class LuchadorB {
    int hp;
    int dmg;

    public LuchadorB(){
        hp = 500;
        dmg = 50;
    }

    public void golpear(LuchadorA luchadorA){
        luchadorA.hp = luchadorA.hp - 50;
        if (luchadorA.hp <= 0){
            System.out.println("Gana luchador B");
        }
    }
}
```

2. **El supermercado**
   La cajera de un supermercado atiende a los clientes, uno por uno, dispuestos en una cola. La cajera atiende al primer cliente de la cola, escanea los productos de su carrito de compras uno por uno pasándolos del carrito al área de empaquetado. Una vez escaneados todos los productos, le cobra al cliente, el cliente paga y sale de la cola, entonces la cajera atiende al siguiente cliente. Modele esta situación usando clases y aplicando encapsulamiento.


Project Manager
Project Engineer
Project Superintendent
Site Foreman
Shell Foreman
Interiors
Site Clearing
Landscaping
Utilities
Concrete
Building Contractor
Glaziers
Paint
Project Manager
Drywall & Framing
Superintendent
Rebar Subcontractor
Supplier
Flooring
Project Manager