

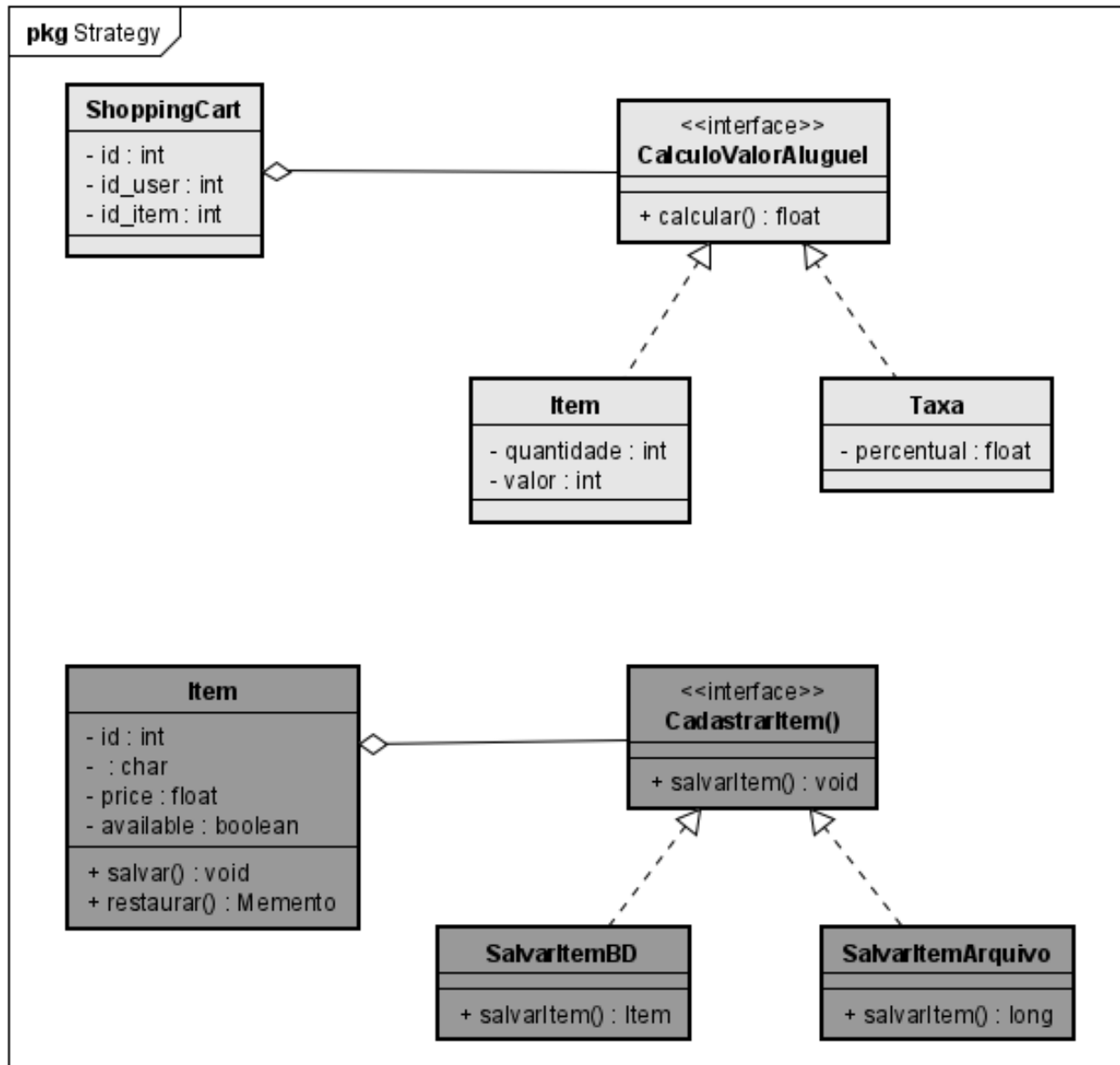


Design Patterns - Individual

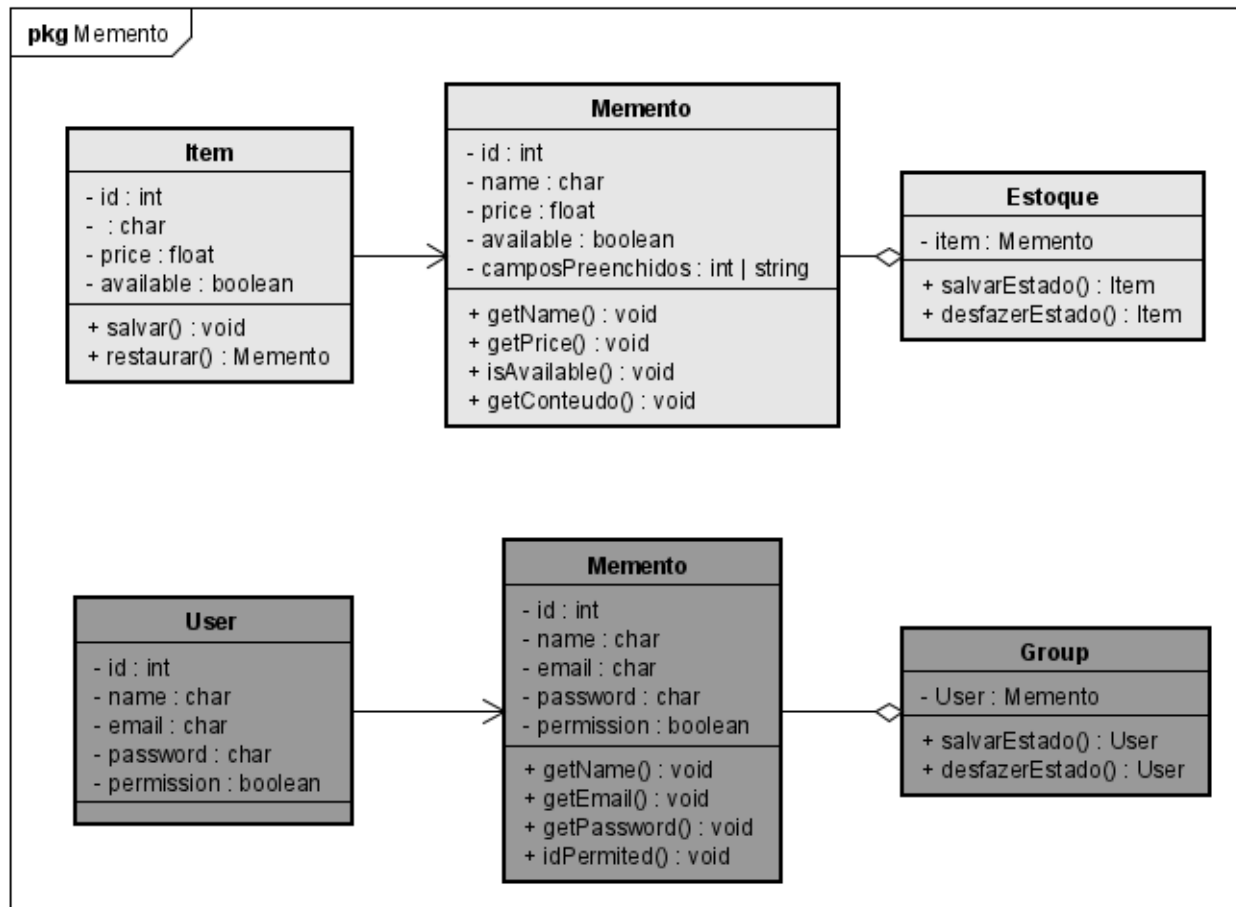
Ary Farah

Design Patterns

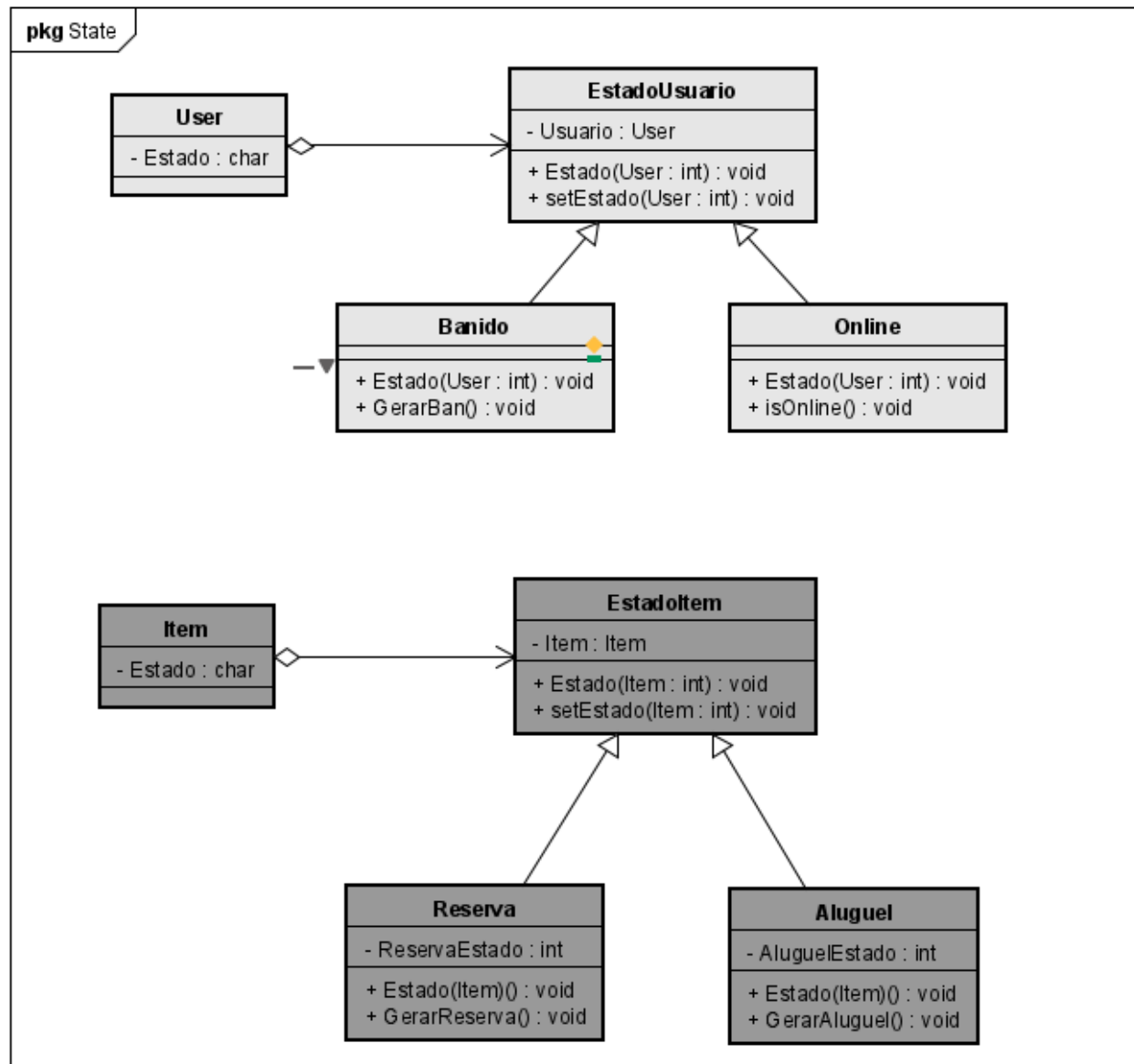
1. Strategy



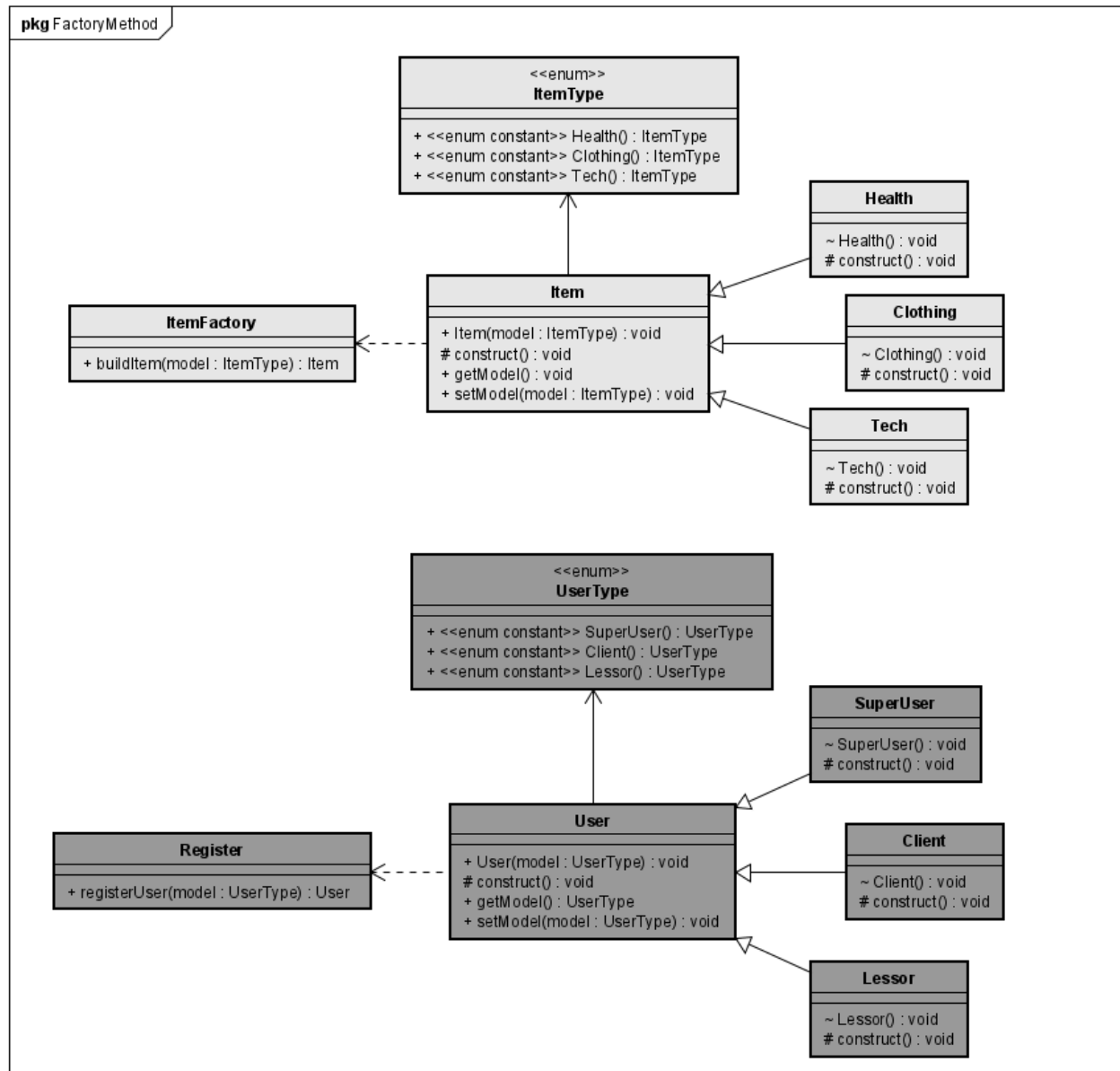
2. Memento



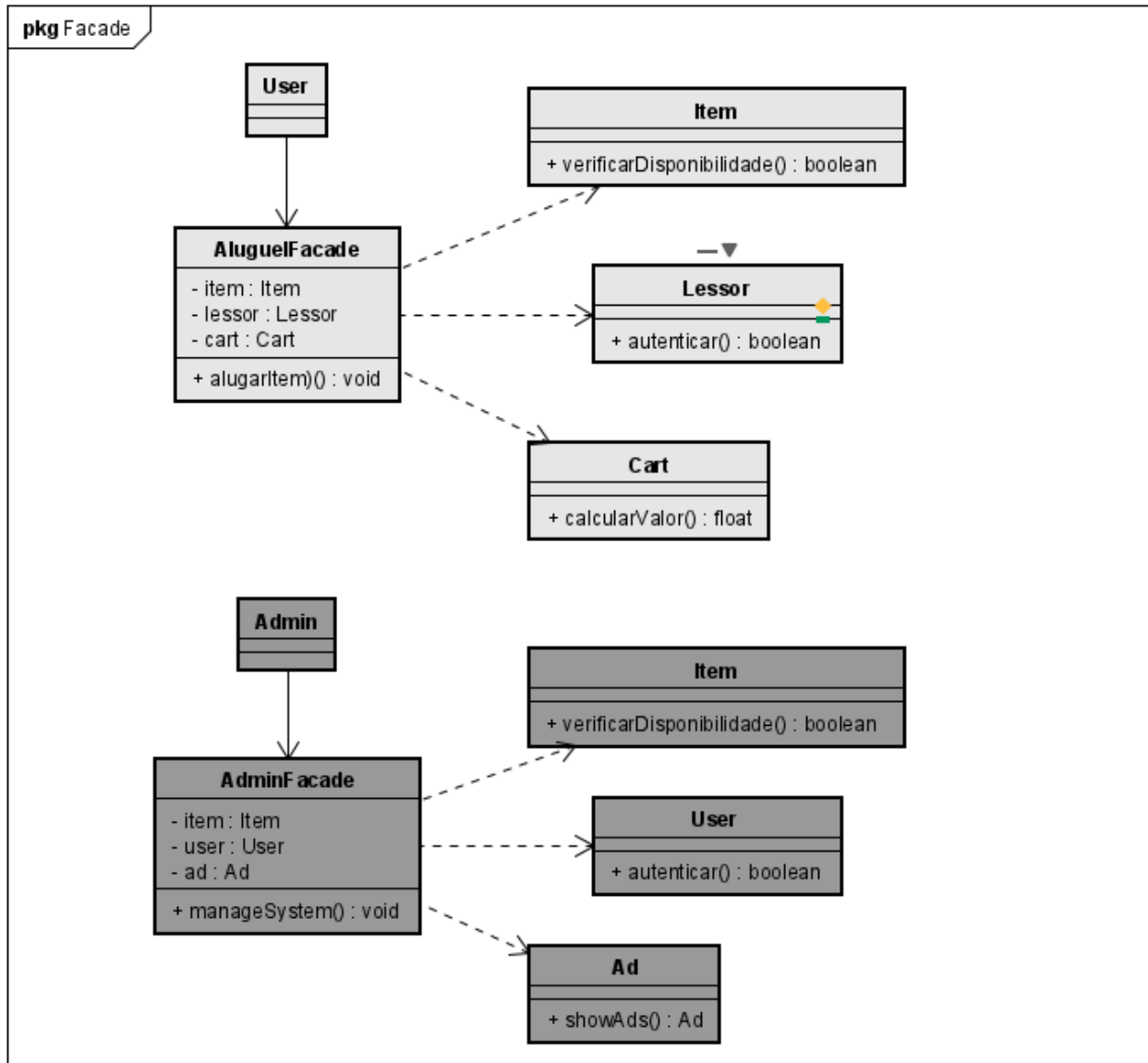
3. State



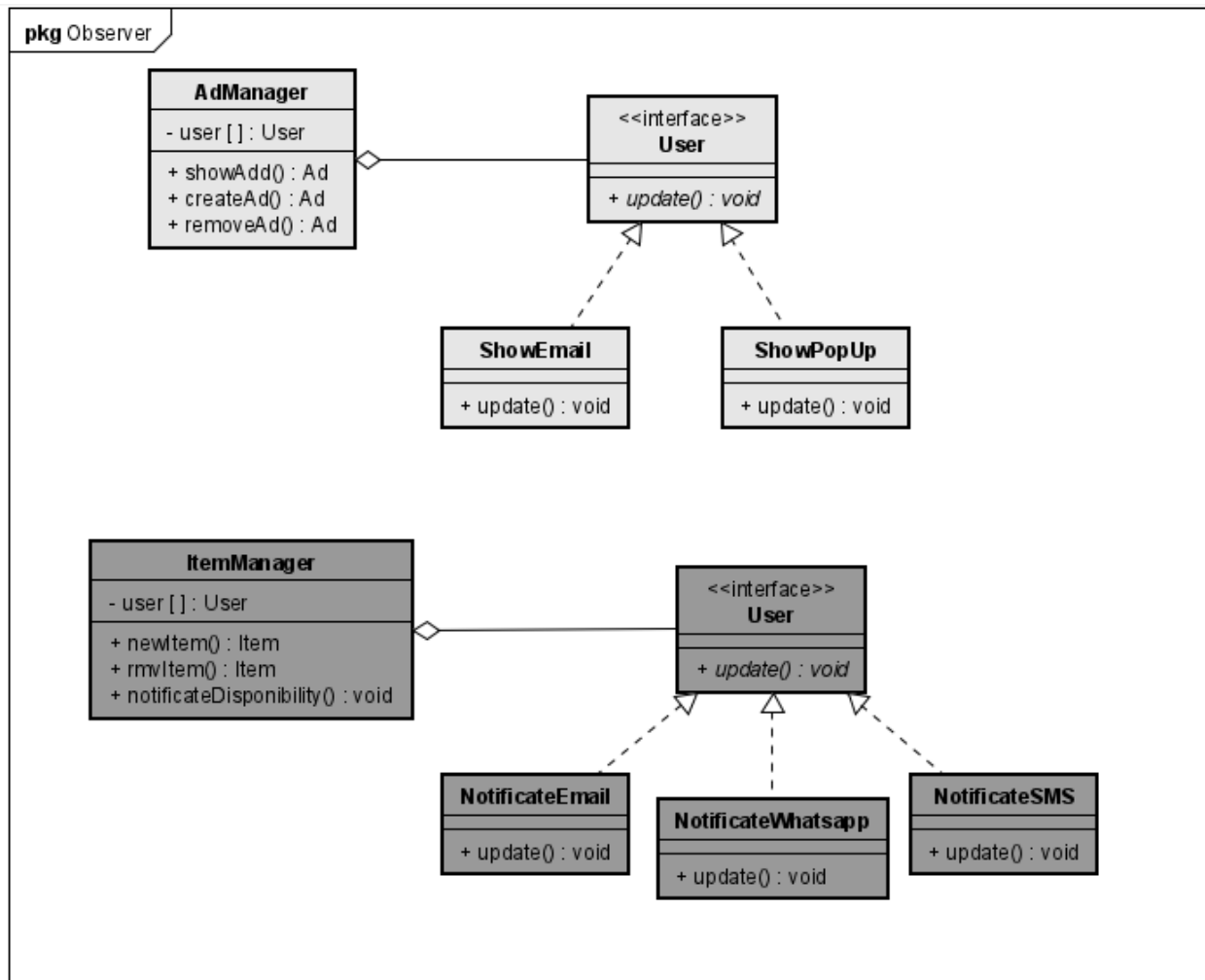
4. Factory Method



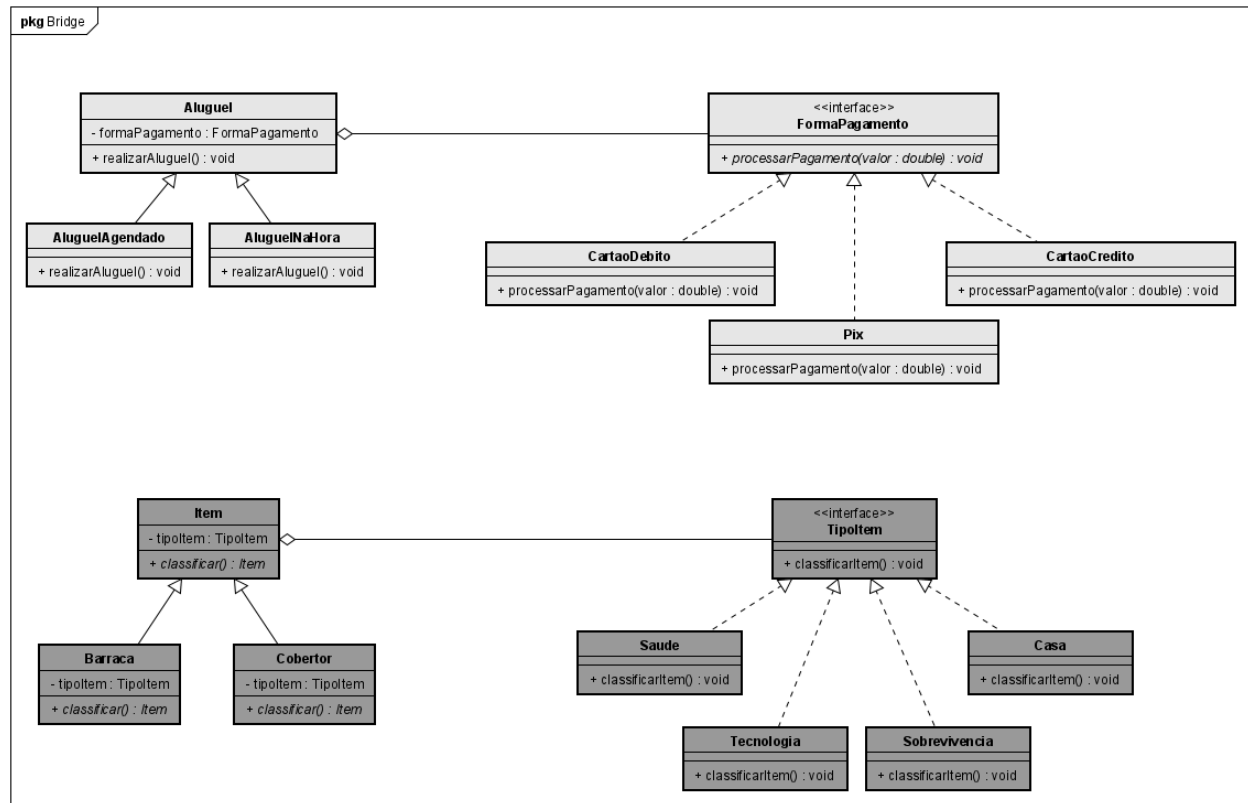
5. Facade



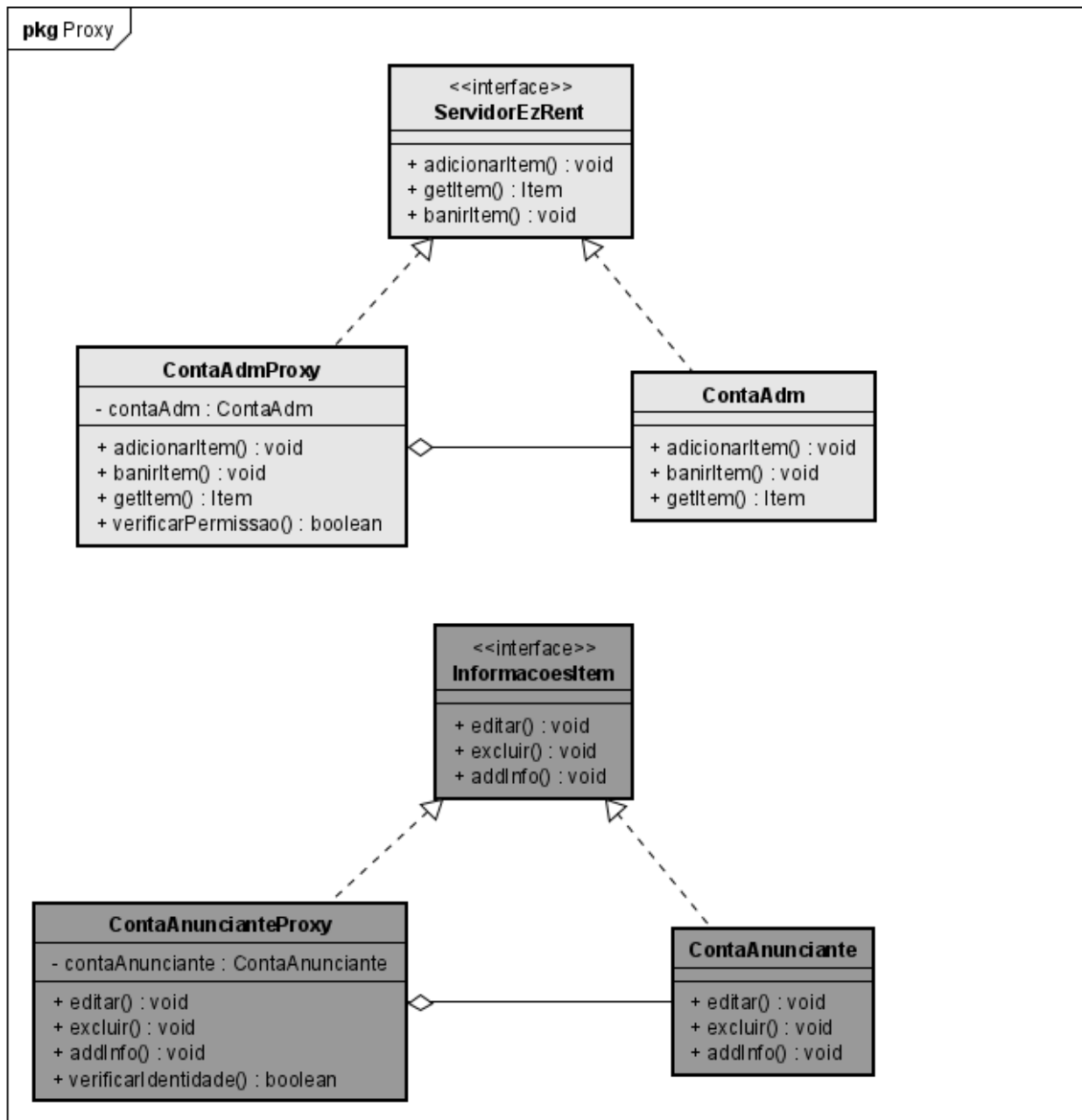
6. Observer



7. Bridge



8. Proxy



Implementação

1. Memento

```
public class Estoque {  
    private Memento memento;  
  
    // CONSTRUTOR  
    public Estoque(Memento memento) {  
        this.memento = memento;  
    }  
  
    public void salvarEstado(Item item) {  
        this.memento = item.save();  
    }  
  
    public void desfazerEstado(Item item) {  
        if (memento != null) {  
            item.restaurar(memento);  
        }  
    }  
}
```

```

public class Item {
    private String nameItem;

    // GETTERS E SETTERS
    public void setNameItem(String nameItem) {
        this.nameItem = nameItem;
        System.out.println("Nome do item alterado para: " + nameItem);
    }

    public Memento save() {
        return new Memento(nameItem);
    }

    public void restaurar(Memento memento) {
        this.nameItem = memento.getNameItem();
        System.out.println("Nome do item restaurado para: " + nameItem);
    }
}

```

```

public class Memento {
    private final String nameItem;

    // CONSTRUTOR
    public Memento(String nameItem) {
        this.nameItem = nameItem;
    }

    public String getNameItem() {
        return nameItem;
    }
}

```

2. State

```
public interface EstadoUser {  
    void estado(User user, int estado);  
}
```

```
public class Banido implements EstadoUser {  
  
    @Override  
    public void estado(User user, int estado) {  
        gerarBan();  
        System.out.println(x:"Usuário banido");  
        user.setEstado(estadoChar:"Banido");  
    }  
  
    public void gerarBan() {  
        System.out.println(x:"Você foi banido");  
    }  
  
}
```

```
public class Online implements EstadoUser {  
  
    @Override  
    public void estado(User user, int estado) {  
        System.out.println(x:"Usuário online");  
        user.setEstado(estadoChar:"Online");  
    }  
  
    public void isOnline() {  
        System.out.println(x:"Usuário online");  
    }  
  
}
```

```

public class User {
    private EstadoUser estado;
    private String estadoChar;

    public User() {
        this.estado = new Online();
    }

    public void setEstado(EstadoUser estado) {
        this.estado = estado;
    }

    public void mudarEstado(int estado) {
        this.estado.estado(this, estado);
    }

    public String getEstado() {
        return estadoChar;
    }

    public void setEstado(String estadoChar) {
        this.estadoChar = estadoChar;
    }
}

```