## Código Fuente

```java
package command_update;

public enum Boton {
    BotonX,
    BotonY,
    BotonA,
    BotonB
}
```

```java
package command_update;
/**
 * @author Victor Lavalle
 */
public interface Command {
public abstract void ejecutar(Personaje per);
}
```

```java
package command_update;
/**
 * @author Victor Lavalle
 */
public class CommandBrincar implements Command {
    @Override
    public void ejecutar(Personaje per) {
            per.brincar();
    }
}
```

```java
package command_update;
/**
 * @author Victor Lavalle
 */
public class CommandCorrer implements Command{
    @Override
    public void ejecutar(Personaje per) {
        per.correr();
    }
}

```

```java
package command_update;
/**
 * @author Victor Lavalle
 */
public class CommandDisparar implements Command{
    @Override
    public void ejecutar(Personaje per) {
      per.disparar();
    }
}
```

```java
package command_update;
/**
 * @author Victor Lavalle
 */
public class CommandInclinarse implements Command {

    @Override
    public void ejecutar(Personaje per) {
      per.inclinarse();
    }

}
```

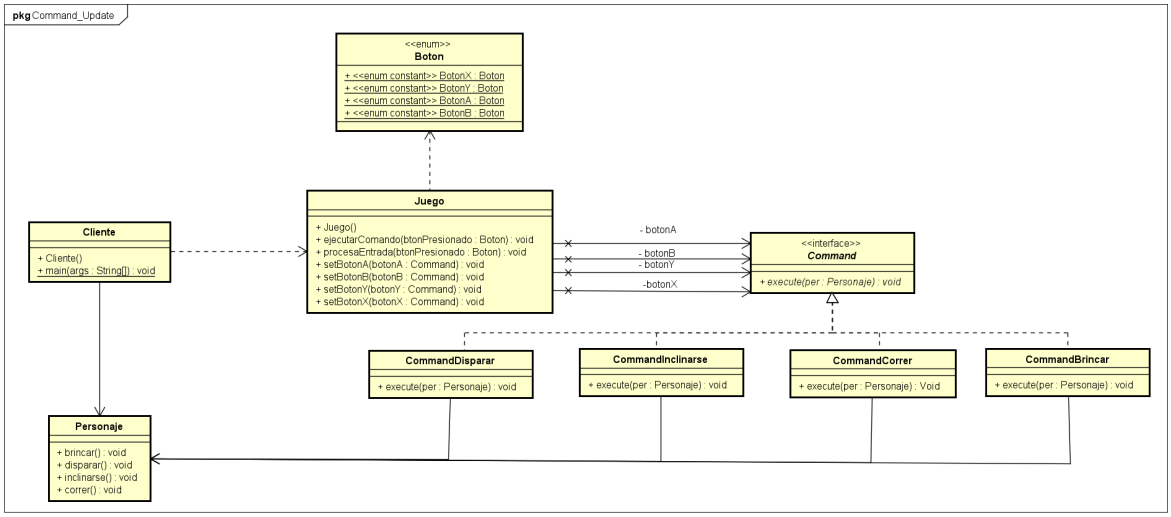
```java
package command_update;

public class Juego {

    private Personaje personaje;
    private Command botonA;
    private Command botonB;
    private Command botonY;
    private Command botonX;

    public Juego() {
      this.personaje = new Personaje();
    }

    public void ejecutarComando(Boton btonPresionado){
    procesaEntrada(btonPresionado);
    }

    public void procesaEntrada(Boton btonPresionado){
    if(btonPresionado==Boton.BotonA)botonA.ejecutar(this.personaje);
    if(btonPresionado==Boton.BotonB)botonB.ejecutar(this.personaje);
    if(btonPresionado==Boton.BotonX)botonX.ejecutar(this.personaje);
    if(btonPresionado==Boton.BotonY)botonY.ejecutar(this.personaje);
    }

    public void setBotonA(Command botonA) {
        this.botonA = botonA;
    }

    public void setBotonB(Command botonB) {
        this.botonB = botonB;
    }

    public void setBotonY(Command botonY) {
        this.botonY = botonY;
    }

    public void setBotonX(Command botonX) {
        this.botonX = botonX;
    }
}
```

```java
package command_update;
/**
 * @author Victor Lavalle
 */
public class Cliente {

    public Cliente(){

    Juego Game = new Juego();

    Game.setBotonA(new CommandBrincar());
    Game.setBotonB(new CommandCorrer());
    Game.setBotonX(new CommandDisparar());
    Game.setBotonY(new CommandInclinarse());

    Game.ejecutarComando(Boton.BotonY);
    Game.ejecutarComando(Boton.BotonA);
    Game.ejecutarComando(Boton.BotonB);
    Game.ejecutarComando(Boton.BotonX);
    }

    public static void main(String[] args) {
        System.out.println("»COMMAND GAME«\n");
        Cliente cliente = new Cliente();
    }
}
```

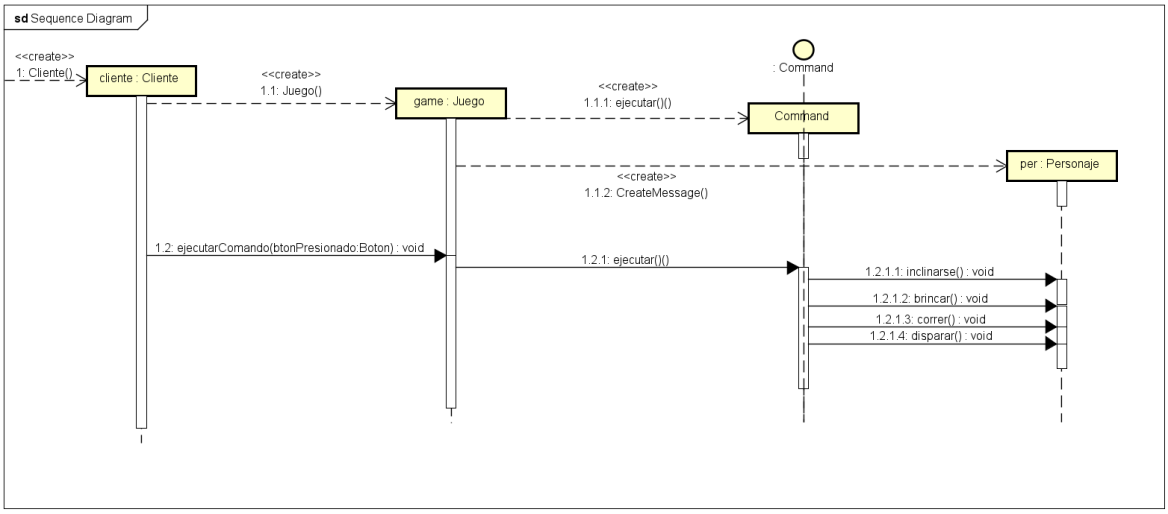
```java
package command_update;
/**
 * @author Victor Lavalle
 */
public class Personaje {
    public void brincar() {
        System.out.println("\n¤Brincando...\n"
                + "\n" +
"             (_)  /\n" +
"            /-|--/\n" +
"            \\ |\n" +
"              /--i\n" +
"             /   L\n" +
"____________L_________");
    }

    public void disparar() {
        System.out.println("\n¤Disparando...\n"
                + " O\n"
                +"/|\\╒═ °°      \n"
                + " |\n"
                +"/ \\\n"
                +"--------------------");
    }


    public void inclinarse() {
        System.out.println("\n¤Inclinándose...\n"
                + "   _ 0\n"
                +" O` / \\\n"
                +"   |\\\n"
```

```
                +"  /   |  \n"
                + "--------------------");

    }

    public void correr() {
        System.out.println("\n¤Corriendo...\n"
                + "     __O\n"
                + "    / /\\_,\n"
                + " ___/\\\n"
                + "      /_\n"
                + "--------------------");
    }

}
```

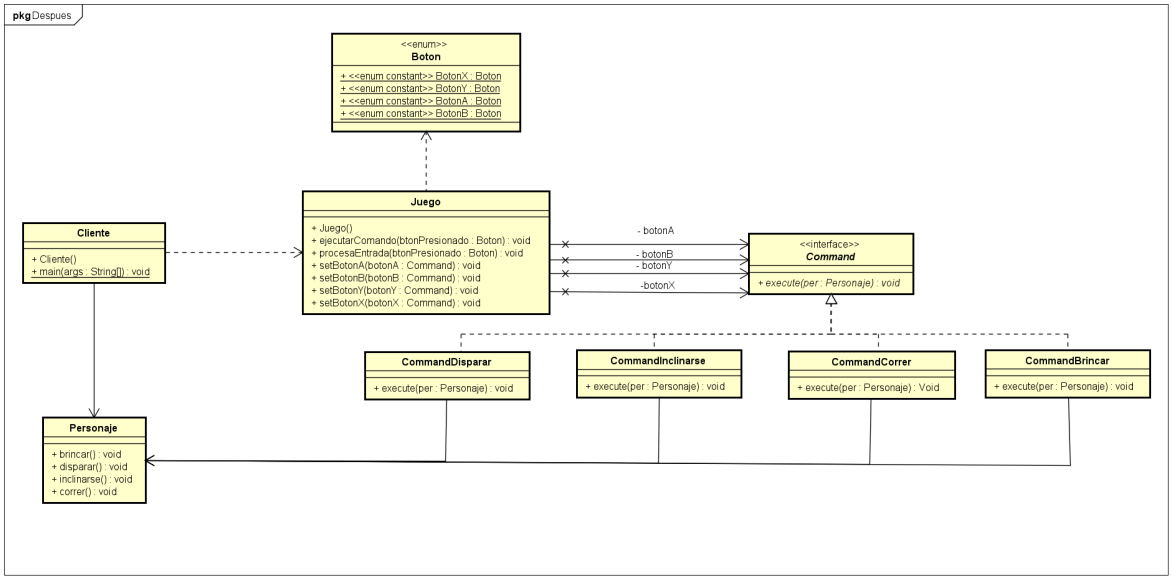
## Diagrama de Clases




## Diagrama de Secuencia



# Ejecución


pkg Despues
<<enum>>
Boton
+ <<enum constant>> BotonX : Boton
+ <<enum constant>> BotonY : Boton
+ <<enum constant>> BotonA : Boton
+ <<enum constant>> BotonB : Boton
Juego
+ Juego()
+ ejecutarComando(btonPresionado : Boton) : void
+ procesaEntrada(btonPresionado : Boton) : void
+ setBotonA(botonA : Command) : void
+ setBotonB(botonB : Command) : void
+ setBotonY(botonY : Command) : void
+ setBotonX(botonX : Command) : void
- botonA
- botonB
- botonY
-botonX
Cliente
+ Cliente()
+ main(args : String[]) : void
<<interface>>
Command
+ execute(per : Personaje) : void
CommandDisparar
+ execute(per : Personaje) : void
CommandInclinarse
+ execute(per : Personaje) : void
CommandCorrer
+ execute(per : Personaje) : Void
CommandBrincar
+ execute(per : Personaje) : void
Personaje
+ brincar() : void
+ disparar() : void
+ inclinarse() : void
+ correr() : void