

Bachillerato en Ingeniería en Sistemas

Entregable escrito

Prof. Eduardo Gonzalez Paniagua

Visión General de los Patrones de Arquitectura

Elaborado por:

Santiago Betancur

II Cuatrimestre 2019

INTRODUCCION

En este programa se implementa los patrones creación, estructurales y comportamiento, en el desarrollo de una aplicación de vista de objetos ilustrados en la pantalla, es una emulación de un programa tipo Paint, en el cual se puede escoger colores y tipo de forma, en la cuales se pueden implementar tanto figuras de cuadrados, círculos rectángulos y línea.

Los patrones de diseño vienen a solucionar o facilitar la creación de soluciones, también es la manera de estandarizar las soluciones para hacerlas mas entendibles para cualquier otro desarrollador.

DESARROLLO

Área de dibujo

En este objeto se crea los eventos para la implementación dibujo de las figuras, además de recoger tanto el tamaño como la ubicación de los objetos al crearlos.

El elemento encargado de la selección del tipo de objetos.

public void setTipoFigura(String dato) {}

Por parámetro se introduce que tipo de elemento es el cual se va implementar, para dibujar se usa el método

public void paint(Graphics g) {}

y se crea el método que repinta el canvas que se utiliza para mostrar las imágenes

public void update(Graphics g)

{paint(g);}

Y después viene la implementación de recoger los movimientos que se realizan con el mause

Tipo de figura

Se crean por cada una de las figuras un objeto con sus propios métodos que se definen, por medio de herencia se implementan las características que definen que tipo es, ya sea circulo, cuadro, rectángulo y circulo.

El objeto figura que es de tipo abstracto define la construcción básica de los tipos circulo, cuadro…

abstract class Figura extends javax.swing.JFrame{

private String Color;

private String Etiqueta;

private int x1, y1;

private int x2, y2;

public Figura(int x1, int y1, int x2, int y2)

{

Color = new String("Negro");

Etiqueta = new String("Sin nombre");

this.x1 = x1;

this.y1 = y1;

this.x2 = x2;

this.y2 = y2;

}

public Figura(int x1, int y1, int x2, int y2, String Color, String Etiqueta)

{

this.Color = new String(Color);

this.Etiqueta = new String(Etiqueta);

this.x1 = x1;

this.y1 = y1;

this.x2 = x2;

this.y2 = y2;

}

public String getColor()

{ //obtiene el color

return new String(Color);

}

public String getEtiqueta()

{ //obtiene el nombre

return new String(Etiqueta);

}

public int getX1()

{ //obtiene el primer eje x

return x1;

}

public int getY1()

{ //obtiene el primer eje y

return y1;

}

public int getX2()

{

return x2;

}

public int getY2()

{

return y2;

}

public abstract void dibujar( Graphics g);

public abstract boolean Esta\_en\_las\_Coordenadas(int x, int y);

}

Además se implementa la clase principal llamada paint0.3 que tiene el constructor del frame y como el método que define si el programa esta en función o no para terminarlo.

Se creo también la clase que define todos los elementos que se van a utilizar tanto como botones o checks como la creación del canvas que se define dentro de in frame.

El canvas es el cual se encarga de sobrescribir los objetos en el para la visualización.