



INSTITUTO TECNOLÓGICO SUPERIOR DE CALKINÍ EN EL ESTADO DE
CAMPECHE.

Carrera:

INGENIERÍA EN SISTEMAS
COMPUTACIONALES.

Asignatura:

TOPICOS AVANZADOS DE
PROGRAMACION

Docente:

JOSE LUIS LIRA TURRIZA

Actividad:

REPORTE DE PRÁCTICA NO. 2.-
GENERACIÓN DE APLICACIONES
CON GRÁFICOS EN UN
COMPONENTE TIPO CANVAS

Alumno:

CARLOS FERNANDO CHAN CAUICH
(7010)

Fecha:

23/03/2023

Ciclo escolar:

2022-2023P

PRÁCTICA NO. 2.- GENERACIÓN DE APLICACIONES CON GRÁFICOS EN UN COMPONENTE TIPO CANVAS

INTRODUCCIÓN

La interfaz es la parte visible de las aplicaciones, siendo lo que se percibe de las mismas; por ello, cada vez se les está dando una importancia mayor y se está poniendo más cuidado en su desarrollo. La creación de interfaces de usuario es un área, dentro del desarrollo de software, que ha evolucionado mucho en los últimos años y lo sigue haciendo a día de hoy.

Se podría decir que la interfaz de usuario es lo primero que se juzga de una aplicación, y si no tiene la calidad adecuada puede producir rechazo por parte del usuario. Una interfaz puede incluso poner limitaciones en la comunicación de la máquina con el usuario. Todo aquello que no se pueda expresar a través de la interfaz se perderá, por ello, tiene gran importancia dentro del desarrollo de una aplicación el diseño de su interfaz.

Los elementos que componen la interfaz gráfica son elementos visuales, y a través de ellos el usuario puede interactuar con la aplicación. En esta interacción el usuario introduce datos que el programa necesita para llevar a cabo su funcionalidad y obtiene los resultados de procesar dichos datos. Por ejemplo, las ventanas, los botones, las imágenes, etc. Son elementos gráficos. Una diferencia clara entre una aplicación de consola y una aplicación con interfaz gráfica de usuario, es que la primera no tiene ningún elemento gráfico, mientras que en la segunda éstos si existen.

Las interfaces gráficas están formadas por ventanas de diferentes tipos que se pueden solapar, mover, cerrar, etc. Dentro de estas ventanas se encuentran otros elementos (botones, etiquetas, campos de texto, imágenes, etc.) que permiten introducir datos y mostrar el estado de la aplicación. El ratón y el teclado permiten manejar los elementos que forman parte de la interfaz.

Para crear nuevas formas existe la clase Canvas. Ésta existe para tener descendientes, no hace nada por sí misma; sólo proporciona una forma para

implementar un componente personalizado. Por ejemplo, los Lienzos (Canvas) son útiles para áreas de dibujo para imágenes y gráficos del cliente, tanto si se desea o no manejar eventos que ocurran dentro del área de pantalla.

Los Canvas también son útiles cuando se quiera un control -- un botón, por ejemplo -- que se parezca a la implementación por defecto de ese control. Como no se puede cambiar la apariencia de los controles estándares creando una subclase de la clase Component correspondiente (Button, por ejemplo), en su lugar se puede crear una subclase de Canvas que tenga el comportamiento que se quiera y el mismo comportamiento que la implementación por defecto del control.

OBJETIVO

El estudiante será capaz de implementar interfaces gráficas utilizando las clases Canvas para dibujar en su superficie.

LUGAR

AULA

-SEMANA DE EJECUCIÓN

SEMANA CUATRO (Parcial 1)

MATERIAL Y EQUIPO

- Sistema Operativo
- Procesador de Textos
- Software para el desarrollo de aplicaciones “eclipse”.
- Cañón
- Plumones
- Pizarrón.
- Equipos de cómputo para todos los estudiantes de la asignatura.

DESARROLLO DE LA PRÁCTICA

Una empresa de desarrollo de video juegos pretende crear sus propios componentes visuales para sus juegos, los componentes que desea crear deben

ser dependientes del teclado y del mouse debido a que son juegos de PC. Se desean crear 3 componentes diferentes, todos ellos asociados a los botones comunes de las aplicaciones de escritorio, por lo que se desea que reaccionen al clic del mouse.

1. El primer componente es el botón, en éste se debe poner el texto asociado a la acción a utilizar y la posibilidad de marcar una letra para poder ejecutar la acción rápida con el teclado.

2. El segundo elemento es un contenedor, que permitirá “arrastrar” los objetos de un lado a otro dentro de su área dibujable con los elementos del mouse.

3. Y el tercer componente será un contenedor temporal, que contendrá un conjunto



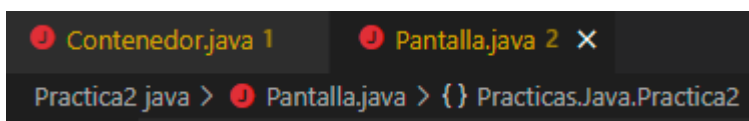
de componentes “arrastrables” que permitan llevar

los objetos que se usarán dentro del componente contenedor y del cual se recuperarán dichos componentes.

4. Escribe una aplicación que

utilice los diseños creados y demuestren la funcionalidad de cada uno.

5. Documente los resultados.



Como se puede observar, se crearon 2 clases

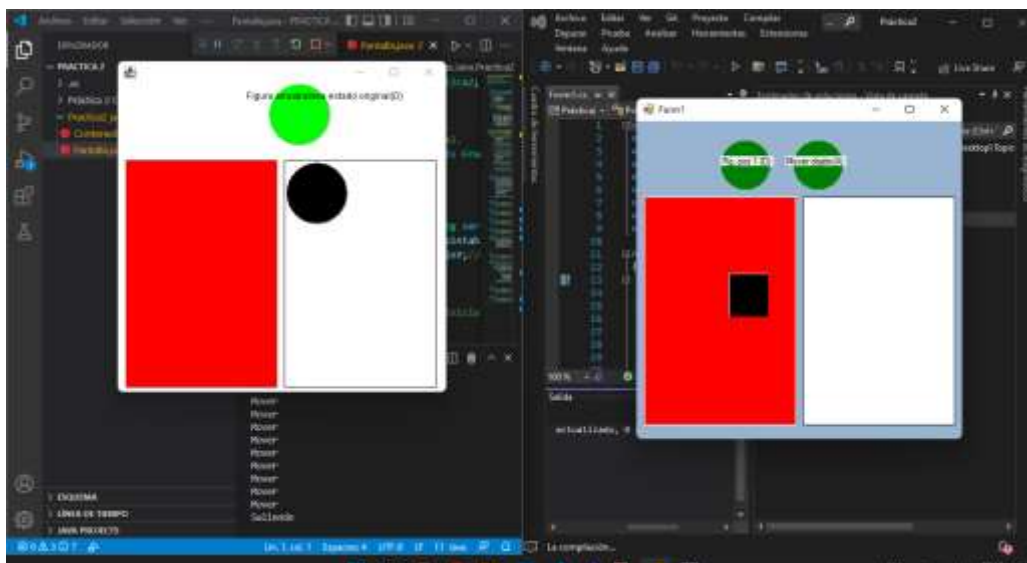
En la clase contenedor es en ella en la que se crean las figuras, heredando de la clase Cava, la que nos permite el trazo de estas, en el que se definieron tanto las coordenadas en las que deben de ser dibujadas, hasta el tipo de figuras que fueron

figura, luego de esto se definieron las acciones que tendrá el mouse y una tecla, de manera general se arrastrará la figura con el mouse y con una tecla se reiniciará la figura arrastrable a su posición original.

En el caso del C#, como bien se sabe este ya tiene una estructura definida, en una clase se definen todos los componentes, en otra se les da función, en otra se ejecutan y hay una ventana que nos permite ver de manera gráfica lo que se está haciendo. En este caso se usó un picturebox para poder dibujar las figuras, así como tres paneles, dos para botones y uno para la figura arrastrable, en los cuales se dibujaron figuras correspondientes a lo solicitado.

Lo que se hizo de manera general es en el picturebox se dibujaron los dos cuadrados para el contenedor y el contenedor temporal; en el caso de los botones y el objeto arrastrable estos ya se dibujaron en paneles correspondientes, luego de esto a los botones se les dio función con el clic, el cual es el de mover la figura y de devolver la figura arrastrable a su posición inicial, debido a que como se sabe, al momento de ingresar en el cuadrado rojo, este ya no podrá salir de su área. Por lo tanto, también se usó un método de colisiones y varios booleanos para poder trabajar de una mejor manera.

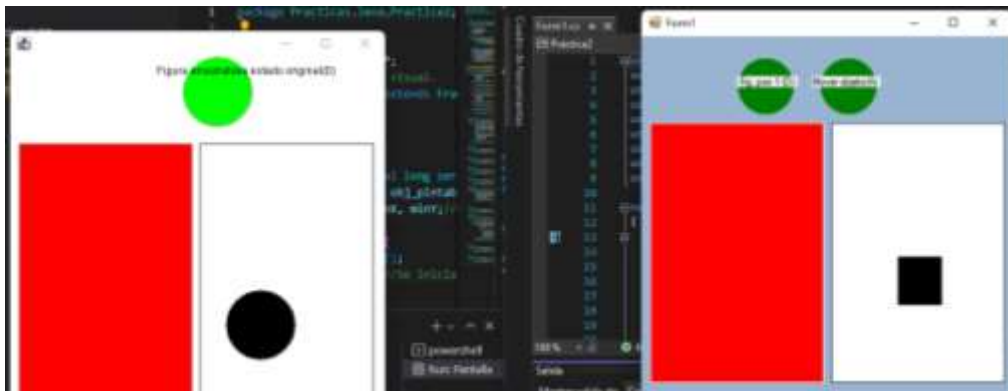
EJECUCION DEL PROGRAMA



C#

Como se observa, se presentan los 3 elementos solicitados, presentando en la parte de arriba un botón de color verde (dos botones en el caso de C#), que al ser presionados realizan una acción, un cuadro rojo que al meter elementos en el no deja que estos salgan y un contenedor temporal que contiene un objeto arrastrable en este caso.

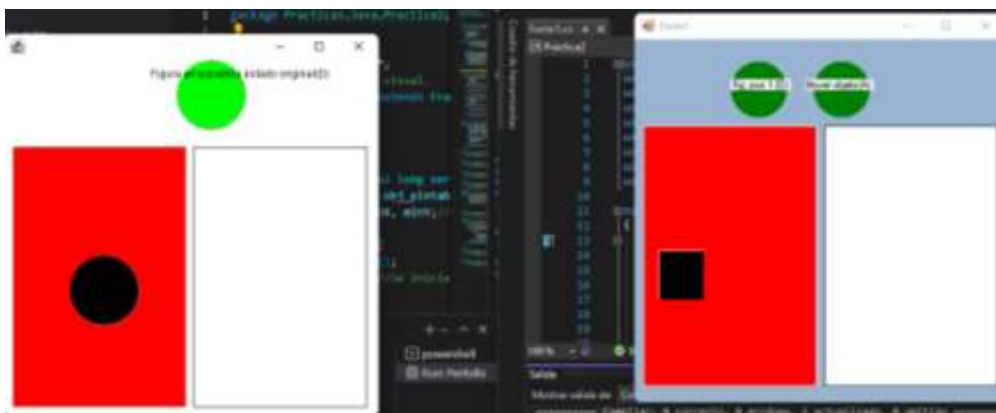
CASO DEL CONTENEDOR TEMPORAL.



C#

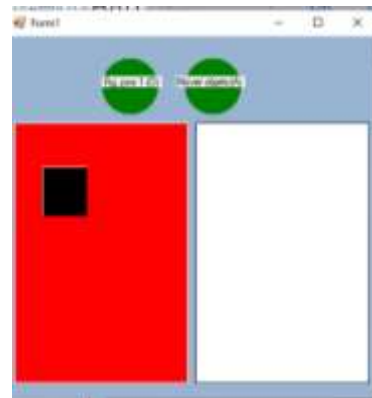
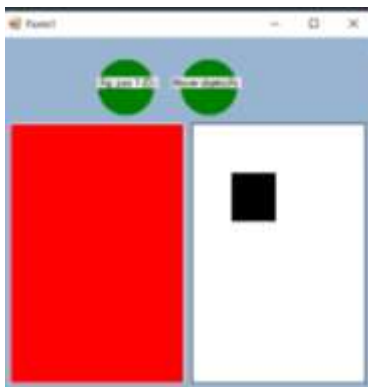
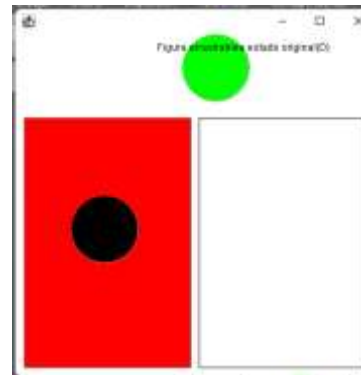
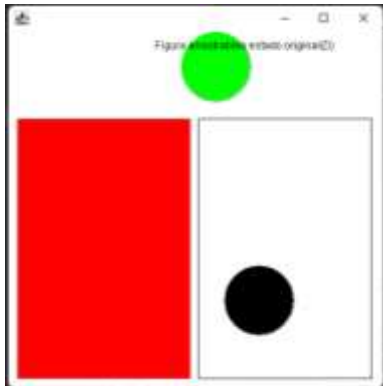
Como se puede ver, el contenedor temporal es el presentado en la parte derecha, en el que, si se notó en la imagen principal, esta tenía una figura dentro, la cual puede ser arrastrada a cualquier parte, tal y como se solicita.

CONTENEDOR.



C#

En este caso como se puede observar, se nos presenta el contenedor principal, que se encuentra en el lado izquierdo, en el que como se observa, una vez que la figura esté en su interior, esta no puede salir de aquí, ya que, al momento de tocar el borde del contenedor, esta es redibujada al interior de este.



C#

En el caso del botón, lo que se hizo fue que al momento de presionarlo o presionar la tecla que está definida en esta, en este caso la “d”, esta nos regresa al objeto arrastrable a su posición inicial dentro del contenedor temporal, ya que no hay otra manera de poder sacarlo; en el caso de C# se agrega este mismo botón y se anexa uno llamado mover, el que permite que el objeto arrastrable sea movido por la

ventana, esto se puede hacer presionando el botón u oprimiendo la letra “a” como se puede observar. Presentando así, todos y cada uno de los elementos solicitados.

Resultados: Como se pudo ver al momento de presentar los resultados de esta práctica, esta fue realizada en el lenguaje Java, así como en lenguaje de C#, aunque estos tenían sus diferencias al momento de realizarse, resultó muy similar el desarrollo de estos programas en ambos, debido a que en el caso de las colisiones y dibujo de las figuras era muy similar en ambos casos, cambiando solamente en algunos métodos. Con esto se espera que lo solicitado sea lo correcto, entonces como resultado tenemos un programa funcional que nos permite ver lo solicitado en la práctica.

CONCLUSIÓN.

Luego de haber realizado esta práctica y como bien se mencionó en la parte de resultados, no nos quedan más que algunas dudas sobre estos temas, si bien se pudo realizar lo que se consideraba correcto al momento de ir interpretando las instrucciones, luego de recibir un poco más de tiempo puedo decir que el programa presentado sí contiene los elementos solicitados, presentando un botón, un contenedor que contiene un objeto arrastrable y un contenedor en el que apenas se ingresa el elemento este queda ahí hasta que se presiona el botón o tecla que lo saca de ahí.

EVALUACIÓN Y RESULTADOS

Se describe la forma de evaluar la práctica desarrollada mediante la entrega del Informe Técnico solicitado por el Profesor, el cual puede contener tablas, planos, prototipos, gráficas, diagramas o dibujos, observaciones, conclusiones, cuestionario y referencias.

REFERENCIAS

Como utilizar la clase Canvas. Link:
<http://www.binarycode.com/bdescargas/Manuales%20y%20Documentos/JAVA/Int>

The screenshot shows the Visual Studio Code interface with a C++ project named 'PRACTICA 2'. The Explorer on the left shows the project structure, including 'Practical.h' and 'Practical.cpp'. The Editor displays the 'Practical.h' header file, which defines the 'Controlador' class. The class has a constructor that initializes variables x, y, w, h, and isClicked. The bottom status bar shows the file path: 'C:\Users\FERNANDO\Desktop\Software\PRACTICA 2'.

```

//Practical.h
#pragma once
#include <iostream>

using namespace std;

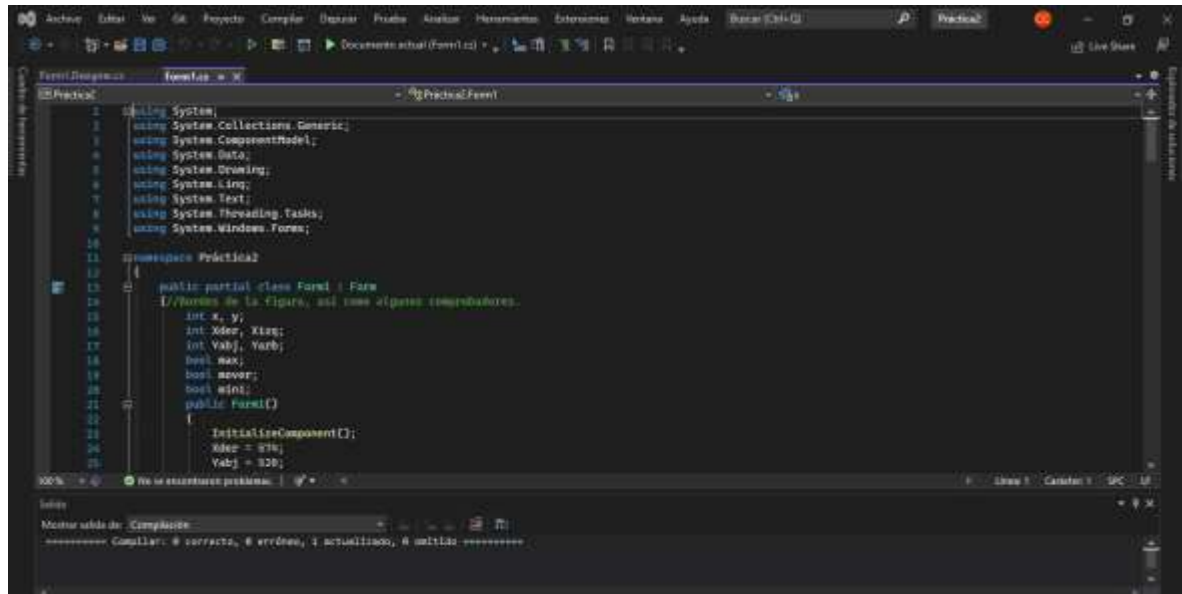
class Controlador {
public:
    Controlador();
    ~Controlador();
    void setX(int x);
    void setY(int y);
    void setW(int w);
    void setH(int h);
    void setClicked(bool isClicked);
    int getX();
    int getY();
    int getW();
    int getH();
    bool getClicked();
};

```

The bottom status bar shows the file path: 'C:\Users\FERNANDO\Desktop\Software\PRACTICA 2'.

C#

CLASE PANTALLA



```
1 using System;
2 using System.Collections.Generic;
3 using System.ComponentModel;
4 using System.Data;
5 using System.Drawing;
6 using System.Linq;
7 using System.Text;
8 using System.Threading.Tasks;
9 using System.Windows.Forms;
10
11 namespace Práctica2
12 {
13     public partial class Form1 : Form
14     {
15         //Formas de la figura, así como algunos comprobadores.
16         int x, y;
17         int Xsize, Xsize;
18         int Ysize, Ysize;
19         bool max;
20         bool menor;
21         bool mini;
22         public Form1()
23         {
24             InitializeComponent();
25             Xsize = 474;
26             Ysize = 320;
27         }
28     }
29 }
```