using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Data.Entity;

using System.Drawing;

using System.Linq;

using System.Reflection.Emit;

using System.Security.Policy;

using System.Text;

using System.Threading;

using System.Threading.Tasks;

using System.Timers;

using System.Windows.Forms;

using 抽卡.Model;

using static System.Windows.Forms.VisualStyles.VisualStyleElement;

using static System.Windows.Forms.VisualStyles.VisualStyleElement.ProgressBar;

using static System.Windows.Forms.VisualStyles.VisualStyleElement.Rebar;

namespace 抽卡

{

public partial class 抽卡 : Form

{

private bool isToggleOn = false;

private bool isBtnOn = true;

private bool isSingleToggleOn = true;

private bool isSingleBtnOn = false;

private bool isSwitchCatchaMOn = false;

貓抓抓Entities dbContext = new 貓抓抓Entities();

private System.Timers.Timer Timer\_CloseGif;

private System.Timers.Timer Timer\_CloseCatcha;

string Pn;

mixRandomDrow RD = new mixRandomDrow();

int CatPoint = 45000000;

int Gash = 900000;

bool hasWonPrize;

int x;

int y;

int couponCount;

System.Drawing.Image P;

string prizeN;

public 抽卡()

{

InitializeComponent();

//InitializeAnimation();

this.pictureBox1.Image = new Bitmap("../../CatPicture\\cat\_g\_sR.gif");

Lab\_Gash.Text = $"紅利:{Gash}";

Lab\_CatPoint.Text = $"貓幣:{CatPoint}";

label1.ForeColor = Color.FromArgb(0x9D, 0x5C, 0x63);

this.KeyDown += 抽卡\_KeyDown;

}

//禁止使用ENTER

private void 抽卡\_KeyDown(object sender, KeyEventArgs e)

{

if (e.KeyCode == Keys.Enter)

{

e.Handled = true;

e.SuppressKeyPress = true;

}

}

private void button2\_Click\_1(object sender, EventArgs e)

{

RD.ItemP(dataGridView1);

}

//紅利單抽

private void SingleGashDraw\_Click(object sender, EventArgs e)

{

//GashaponAnimation frmGA = new GashaponAnimation();

if (Gash >= 100)

{

if (MessageBox.Show("消耗100紅利抽卡", "", MessageBoxButtons.YesNo) == DialogResult.Yes)

{

SingleGashDraw();

Lab\_Gash.Text = $"紅利:{(Gash -= 100)}";

}

}

else

{

MessageBox.Show("紅利不足");

}

Btn\_back.Focus();

}

//紅利十抽

private void TenGashDraw\_Click(object sender, EventArgs e)

{

if (Gash >= 900)

{

if (MessageBox.Show("消耗900紅利抽卡", "", MessageBoxButtons.YesNo) == DialogResult.Yes)

{

TenGashDraw();

Lab\_Gash.Text = $"紅利:{(Gash -= 900)}";

}

}

else

{

MessageBox.Show("紅利不足");

}

Btn\_back.Focus();

}

//貓幣單抽

private void SingleCatPointDraw\_Click(object sender, EventArgs e)

{

if (CatPoint >= 500)

{

if (MessageBox.Show("消耗500貓幣抽卡", "", MessageBoxButtons.YesNo) == DialogResult.Yes)

{

SingleCatPointDraw();

Lab\_CatPoint.Text = $"貓幣:{(CatPoint -= 500)}";

}

}

else

{

MessageBox.Show("貓幣不足");

}

Btn\_back.Focus();

}

//貓幣十抽

private void TenCatPointDraw\_Click(object sender, EventArgs e)

{

if (CatPoint >= 4500)

{

if (MessageBox.Show("消耗4500貓幣抽卡", "", MessageBoxButtons.YesNo) == DialogResult.Yes)

{

TenCatPointDraw();

Lab\_CatPoint.Text = $"貓幣:{(CatPoint -= 4500)}";

}

}

else

{

MessageBox.Show("貓幣不足");

}

Btn\_back.Focus();

}

//將轉蛋動畫移置最上層

//機率測試

//int prize()

int prize()

{

//實體折價券 2 %,貓咪 5 %,背景 5 %,飯盆 13 %,家具5 %,飼料25 %,水 % 25,貓幣20 %

//判斷抽獎獎勵方法

int y = 0;

RD.Drow();

P = RD.getP();

int x = RD.getx();

Pn = RD.prizeName();

if (x == 1)

{

prizelist.AppendText($"{Pn}\*1\r\n");

y = 1;

Single\_LabPirzeName.Text = $"{Pn} \* 1";

}

else if (x == 2)

{

prizelist.AppendText($"{Pn}\*1\r\n");

y = 1;

Single\_LabPirzeName.Text = $"{Pn} \* 1";

}

else if (x == 3)

{

prizelist.AppendText($"{Pn}\*1\r\n");

y = 2;

Single\_LabPirzeName.Text = $"{Pn} \* 1";

}

else if (x == 4)

{

prizelist.AppendText($"{Pn}\*1\r\n");

y = 2;

Single\_LabPirzeName.Text = $"{Pn} \* 1";

}

else if (x == 5)

{

prizelist.AppendText($"{Pn}\*1\r\n");

y = 2;

Single\_LabPirzeName.Text = $"{Pn} \* 1";

}

else if (x == 6)

{

prizelist.AppendText($"{Pn}\*1\r\n");

y = 3;

Single\_LabPirzeName.Text = $"{Pn} \* 1";

}

else if (x == 7)

{

prizelist.AppendText($"{Pn}\*1\r\n");

y = 3;

Single\_LabPirzeName.Text = $"{Pn} \* 1";

}

else if (x == 8)

{

prizelist.AppendText($"{Pn}\*1\r\n");

y = 3;

Single\_LabPirzeName.Text = $"{Pn} \* 1";

}

return y;

}

//{

// //實體折價券 2 %,貓咪 5 %,背景 5 %,飯盆 13 %,家具5 %,飼料25 %,水 % 25,貓幣20 %

// //判斷抽獎獎勵方法

// int y = 0;

// RD.Drow(items, out hasWonPrize, out x, out y, out couponCount, out P, out prizeN); ;

// if (x == 1)

// {

// prizelist.AppendText($"{Pn}\*1\r\n");

// y = 1;

// Single\_LabPirzeName.Text = $"{Pn} \* 1";

// }

// else if (x == 2)

// {

// prizelist.AppendText($"{Pn}\*1\r\n");

// y = 1;

// Single\_LabPirzeName.Text = $"{Pn} \* 1";

// }

// else if (x == 3)

// {

// prizelist.AppendText($"{Pn}\*1\r\n");

// y = 2;

// Single\_LabPirzeName.Text = $"{Pn} \* 1";

// }

// else if (x == 4)

// {

// prizelist.AppendText($"{Pn}\*1\r\n");

// y = 2;

// Single\_LabPirzeName.Text = $"{Pn} \* 1";

// }

// else if (x == 5)

// {

// prizelist.AppendText($"{Pn}\*1\r\n");

// y = 2;

// Single\_LabPirzeName.Text = $"{Pn} \* 1";

// }

// else if (x == 6)

// {

// prizelist.AppendText($"{Pn}\*1\r\n");

// y = 3;

// Single\_LabPirzeName.Text = $"{Pn} \* 1";

// }

// else if (x == 7)

// {

// prizelist.AppendText($"{Pn}\*1\r\n");

// y = 3;

// Single\_LabPirzeName.Text = $"{Pn} \* 1";

// }

// else if (x == 8)

// {

// prizelist.AppendText($"{Pn}\*1\r\n");

// y = 3;

// Single\_LabPirzeName.Text = $"{Pn} \* 1";

// }

// return y;

//}

public void SingleCatPointDraw()

{

int highestPriority = 3; // 紀錄最高優先級

int y = 0;

y = prize();

if (y < highestPriority)

{

highestPriority = y;

}

if (highestPriority == 1)

{

CatChaGachaSS();

}

else if (highestPriority == 2)

{

CatChaGachaS();

}

else if (highestPriority == 3)

{

CatChaGachaA();

}

AllCount();

SinglePictureBoxPrize();

SwitchShowSinglePirze();

SwitchTenCatchaM();

}

public void TenCatPointDraw()

{

int highestPriority = 3; // 紀錄最高優先級

int y = 0;

Image[] pValues = new Image[10];

string[] prizeName = new string[10];

for (int i = 0; i < 10; i++)

{

y = prize();

// 判斷優先級並更新最高優先級

if (y < highestPriority)

{

highestPriority = y;

}

pValues[i] = P;

prizeName[i] = Pn;

}

// 執行最高優先級的動畫

if (highestPriority == 1)

{

CatChaGachaSS();

}

else if (highestPriority == 2)

{

CatChaGachaS();

}

else if (highestPriority == 3)

{

CatChaGachaA();

}

AllCount();

for (int i = 0; i < pValues.Length; i++)

{

switch (i)

{

case 0:

PicBox1.Image = pValues[i];

P\_Lab1.Text = prizeName[i];

break;

case 1:

PicBox2.Image = pValues[i];

P\_Lab2.Text = prizeName[i];

break;

case 2:

PicBox3.Image = pValues[i];

P\_Lab3.Text = prizeName[i];

break;

case 3:

PicBox4.Image = pValues[i];

P\_Lab4.Text = prizeName[i];

break;

case 4:

PicBox5.Image = pValues[i];

P\_Lab5.Text = prizeName[i];

break;

case 5:

PicBox6.Image = pValues[i];

P\_Lab6.Text = prizeName[i];

break;

case 6:

PicBox7.Image = pValues[i];

P\_Lab7.Text = prizeName[i];

break;

case 7:

PicBox8.Image = pValues[i];

P\_Lab8.Text = prizeName[i];

break;

case 8:

PicBox9.Image = pValues[i];

P\_Lab9.Text = prizeName[i];

break;

case 9:

PicBox10.Image = pValues[i];

P\_Lab10.Text = prizeName[i];

break;

default:

break;

}

}

SwitchShowTenPrizes();

SwitchTenCatchaM();

}

public void SingleGashDraw()

{

int highestPriority = 3; // 紀錄最高優先級

int y = 0;

y = prize();

if (y < highestPriority)

{

highestPriority = y;

}

if (highestPriority == 1)

{

CatChaGachaSS();

}

else if (highestPriority == 2)

{

CatChaGachaS();

}

else if (highestPriority == 3)

{

CatChaGachaA();

}

AllCount();

SinglePictureBoxPrize();

SwitchShowSinglePirze();

SwitchCatchaM();

}

public void TenGashDraw()

{

int highestPriority = 3; // 紀錄最高優先級

int y = 0;

Image[] pValues = new Image[10];

string[] prizeName = new string[10];

for (int i = 0; i < 10; i++)

{

y = prize();

// 判斷優先級並更新最高優先級

if (y < highestPriority)

{

highestPriority = y;

}

pValues[i] = P;

prizeName[i] = Pn;

}

// 執行最高優先級的動畫

if (highestPriority == 1)

{

CatChaGachaSS();

}

else if (highestPriority == 2)

{

CatChaGachaS();

}

else if (highestPriority == 3)

{

CatChaGachaA();

}

AllCount();

for (int i = 0; i <pValues.Length; i++)

{

switch (i)

{

case 0:

PicBox1.Image = pValues[i];

P\_Lab1.Text = prizeName[i];

break;

case 1:

PicBox2.Image = pValues[i];

P\_Lab2.Text = prizeName[i];

break;

case 2:

PicBox3.Image = pValues[i];

P\_Lab3.Text = prizeName[i];

break;

case 3:

PicBox4.Image = pValues[i];

P\_Lab4.Text = prizeName[i];

break;

case 4:

PicBox5.Image = pValues[i];

P\_Lab5.Text = prizeName[i];

break;

case 5:

PicBox6.Image = pValues[i];

P\_Lab6.Text = prizeName[i];

break;

case 6:

PicBox7.Image = pValues[i];

P\_Lab7.Text = prizeName[i];

break;

case 7:

PicBox8.Image = pValues[i];

P\_Lab8.Text = prizeName[i];

break;

case 8:

PicBox9.Image = pValues[i];

P\_Lab9.Text = prizeName[i];

break;

case 9:

PicBox10.Image = pValues[i];

P\_Lab10.Text = prizeName[i];

break;

default:

break;

}

}

SwitchShowTenPrizes();

SwitchTenCatchaM();

}

//計算抽獎機率

public void SinglePictureBoxPrize()

{

SinglePicturebox.Image = P;

}

//todo..trycatch

//顯示的抽卡動畫A~SS

public void CatChaGachaA()

{

StartAnimation();

pictureBox2.Image = new Bitmap("../../CatPicture\\gacha\_Animate\_A\_v2.gif");

}

public void CatChaGachaS()

{

StartAnimation();

pictureBox2.Image = new Bitmap("../../CatPicture\\gacha\_Animate\_S\_v2.gif");

}

public void CatChaGachaSS()

{

StartAnimation();

pictureBox2.Image = new Bitmap("../../CatPicture\\gacha\_Animate\_SS\_v2.gif");

}

//使用事件設定秒數後關掉gif

void StartAnimation()

{

if (Timer\_CloseGif != null)

{

Timer\_CloseGif.Stop();

Timer\_CloseGif.Dispose();

}

Timer\_CloseGif = new System.Timers.Timer(5900);

Timer\_CloseGif.Elapsed += TimeElapsed;

Timer\_CloseGif.Start();

}

//pircurebox2停止及消失的方法

private void TimeElapsed(object sender, ElapsedEventArgs e)

{

Timer\_CloseGif.Stop();

this.Invoke((MethodInvoker)(() =>

{

this.pictureBox2.Image = null;

this.pictureBox2.Visible = false;

}));

Timer\_CloseGif.Dispose();

}

public void AllCount()

{

int count = RD.Count();

int couponCount = RD.CouponCount();

int catCount = RD.CatCount();

int backGorundCount = RD.BackGorundCount();

int bowlCount = RD.BowlCount();

int furnitureCount = RD.FurnitureCount();

int feedCount = RD.FeedCount();

int waterCount = RD.WaterCount();

int coinCount = RD.CoinCount();

this.Lab\_Count.Text = $"抽了{count}次";

this.Lab\_pay.Text = $"一共花了{count \* 20}元".ToString();

this.Lab\_cointCount.Text = $"貓幣\*{coinCount}機率{((float)coinCount / count \* 100):0.0}%";

this.Lab\_bowlCount.Text = $"飯盆\*{bowlCount}機率{((float)bowlCount / count \* 100):0.0}%";

this.Lab\_waterCount.Text = $"水\*{waterCount}機率{((float)waterCount / count \* 100):0.0}%";

this.Lab\_furnitureCount.Text = $"家具\*{furnitureCount}機率{((float)furnitureCount / count \* 100):0.0}%";

this.Lab\_backGorundCount.Text = $"背景\*{backGorundCount}機率{((float)backGorundCount / count \* 100):0.0}%";

this.Lab\_catCount.Text = $"活力灰貓\*{catCount}機率{((float)catCount / count \* 100):0.0}%";

this.Lab\_couponCount.Text = $"實體券\*{couponCount}機率{((float)couponCount / count \* 100):0.0}%";

this.Lab\_feedCount.Text = $"飼料\*{feedCount}機率{((float)feedCount / count \* 100):0.0}%";

}

private void panel1\_Paint(object sender, PaintEventArgs e)

{

}

private void CloseAllShowPirze\_Click(object sender, EventArgs e)

{

if (Control.ModifierKeys == Keys.Enter)

{

return; // 忽略 Enter 鍵的按鍵事件

}

SwitchShowTenPrizes();

SwitchTenCatchaM();

}

void SwitchShowTenPrizes()

{

isToggleOn = !isToggleOn;

isBtnOn = !isBtnOn;

System.Windows.Forms.Label[] labels = { P\_Lab1, P\_Lab2, P\_Lab3, P\_Lab4, P\_Lab5, P\_Lab6, P\_Lab7, P\_Lab8, P\_Lab9, P\_Lab10 };

System.Windows.Forms.PictureBox[] pictureBoxes = { PicBox1, PicBox2, PicBox3, PicBox4, PicBox5, PicBox6, PicBox7, PicBox8, PicBox9, PicBox10 };

for (int i = 0; i < labels.Length; i++)

{

labels[i].Visible = isToggleOn;

pictureBoxes[i].Visible = isToggleOn;

}

//-----------------------------------------

CloseAllShowPirze.Visible = isToggleOn;

//-----------------------------------------

Btn\_SingleCatPointDraw.Visible = isBtnOn;

Btn\_SingleGashDraw.Visible = isBtnOn;

Btn\_TenCatPointDraw.Visible = isBtnOn;

Btn\_TenGashDraw.Visible = isBtnOn;

//-----------------------------------------

Lab\_UseCatPoint.Visible = isBtnOn;

Lab\_UseCatPoint1.Visible = isBtnOn;

Lab\_UseCatPoint2.Visible = isBtnOn;

Lab\_UseGash.Visible = isBtnOn;

Lab\_UseGash1.Visible = isBtnOn;

Lab\_UseGash2.Visible = isBtnOn;

}

void SwitchShowSinglePirze()

{

isSingleToggleOn = !isSingleToggleOn;

isBtnOn = !isBtnOn;

isSingleBtnOn = !isSingleBtnOn;

SinglePicturebox.Visible = isSingleBtnOn;

Single\_LabPirzeName.Visible = isSingleBtnOn;

//-----------------------------------------

button1.Visible = isSingleBtnOn;

//-----------------------------------------

Btn\_SingleCatPointDraw.Visible = isBtnOn;

Btn\_SingleGashDraw.Visible = isBtnOn;

Btn\_TenCatPointDraw.Visible = isBtnOn;

Btn\_TenGashDraw.Visible = isBtnOn;

//-----------------------------------------

Lab\_UseCatPoint.Visible = isBtnOn;

Lab\_UseCatPoint1.Visible = isBtnOn;

Lab\_UseCatPoint2.Visible = isBtnOn;

Lab\_UseGash.Visible = isBtnOn;

Lab\_UseGash1.Visible = isBtnOn;

Lab\_UseGash2.Visible = isBtnOn;

}

void SwitchCatchaM()

{

isSwitchCatchaMOn = !isSwitchCatchaMOn;

pictureBox2.Visible = isSwitchCatchaMOn;

}

void SwitchTenCatchaM()

{

isSwitchCatchaMOn = !isSwitchCatchaMOn;

pictureBox2.Visible = isSwitchCatchaMOn;

}

private void button1\_Click(object sender, EventArgs e)

{

if (Control.ModifierKeys == Keys.Enter)

{

return; // 忽略 Enter 鍵的按鍵事件

}

SwitchShowSinglePirze();

SwitchCatchaM();

}

}

}