using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Threading;

using System.Windows.Forms;

using FireSharp.Config;

using FireSharp.Response;

using FireSharp.Interfaces;

namespace GatewayExpress

{

public partial class Drivers : Form

{

Thread th; //declaration of thread as th

int panelWidth, taken=0;

bool Hidden;

int Counter = 0;

string idNumber = "", timeNow = "", dateNow = "", dateRegistered = "", accountStatus = "", idHolder = "", purpose = "";

DateTime time = DateTime.Now;

IFirebaseConfig ifc = new FirebaseConfig()

{

AuthSecret = "P2k1udVSBnRvxFKNk1Ve5OVZo8A4x90jWAEFXqYH",

BasePath = "https://lancaster-new-city-default-rtdb.firebaseio.com/"

};

IFirebaseClient client;

void refresh()

{

btnAdd.Enabled = true;

btnUpdate.Enabled = false;

btnDelete.Enabled = false;

try

{

client = new FireSharp.FirebaseClient(ifc);

DataTable dt = new DataTable();

dt.Columns.Add("ID NUMBER");

dt.Columns.Add("USERNAME");

dt.Columns.Add("PASSWORD");

dt.Columns.Add("GIVEN NAME");

dt.Columns.Add("MIDDLE NAME");

dt.Columns.Add("LAST NAME");

dt.Columns.Add("PHONE NUMBER");

dt.Columns.Add("TELEPHONE NUMBER");

dt.Columns.Add("BIRTHDATE");

dt.Columns.Add("GENDER");

dt.Columns.Add("NATIONALITY");

dt.Columns.Add("HOME ADDRESS");

dt.Columns.Add("CITY ADDRESS");

dt.Columns.Add("DATE REGISTERED");

dt.Columns.Add("USER TYPE");

dt.Columns.Add("PURPOSE");

dt.Columns.Add("BALANCE");

dt.Columns.Add("ACCOUNT\_STATUS");

dataGridView.Rows.Clear();

FirebaseResponse res = client.Get(@"USER\_COUNTER/USER\_COUNTER");

int Counter = int.Parse(res.ResultAs<String>());

for (int i = Counter; i >= 0; i--)

{

FirebaseResponse res2 = client.Get(@"USER\_ID/" + i + "/ID\_NUMBER");

string RollNo = res2.ResultAs<string>();

var res3 = client.Get(@"USER INFO/" + RollNo);

MyUser std = res3.ResultAs<MyUser>();

if (std.GIVEN\_NAME != "")//(std!=null)

{

if (std.ACCOUNT\_STATUS == "ACTIVE" || std.ACCOUNT\_STATUS == "PENDING")

{

dt.Rows.Add(RollNo, std.USERNAME, std.PASSWORD, std.GIVEN\_NAME, std.MIDDLE\_NAME, std.LAST\_NAME, std.PHONE\_NUMBER, std.TELEPHONE\_NUMBER, std.BIRTHDATE, std.GENDER, std.NATIONALITY, std.HOME\_ADDRESS, std.CITY\_ADDRESS, std.DATE\_AND\_TIME, std.STATUS, std.PURPOSE, std.BALANCE, std.ACCOUNT\_STATUS);

}

}

}

foreach (DataRow item in dt.Rows)

{

dataGridView.Rows.Add(item.ItemArray);

}

}

catch

{

//MessageBox.Show("No Internet or Connection Problem");

}

dataGridView.BorderStyle = BorderStyle.None;

dataGridView.AlternatingRowsDefaultCellStyle.BackColor = Color.FromArgb(238, 239, 249);

dataGridView.CellBorderStyle = DataGridViewCellBorderStyle.SingleHorizontal;

dataGridView.DefaultCellStyle.SelectionBackColor = Color.DarkTurquoise;

dataGridView.DefaultCellStyle.SelectionForeColor = Color.WhiteSmoke;

dataGridView.BackgroundColor = Color.White;

dataGridView.EnableHeadersVisualStyles = false;

dataGridView.ColumnHeadersBorderStyle = DataGridViewHeaderBorderStyle.None;

dataGridView.ColumnHeadersDefaultCellStyle.BackColor = Color.FromArgb(36, 37, 45);

dataGridView.ColumnHeadersDefaultCellStyle.ForeColor = Color.White;

}

void clear()

{

idNumber = "";

txtUsername.Text = "";

txtPassword.Text = "";

txtGivenName.Text = "";

txtMiddleName.Text = "";

txtLastName.Text = "";

txtPhoneNumber.Text = "(+63)";

txtTelephoneNumber.Text = "";

dtpBirthDate.Text = dateNow;

cboGender.Text = "";

txtNationality.Text = "";

txtHomeAddress.Text = "";

txtCityAddress.Text = "";

dateRegistered = "";

cboStatus.Text = "";

txtBalance.Text = "";

cboAccountStatus.Text = "";

txtPurpose.Text = "";

}

public Drivers()

{

InitializeComponent();

panelWidth = PanelSlide.Width;

Hidden = false;

}

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

{

this.Close(); //command for closing the application

th = new Thread(openSplashScreen); //creating thread

th.SetApartmentState(ApartmentState.STA); //setting thread as STA

th.Start(); //start the thread

}

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

{

this.Close(); //command for closing the application

th = new Thread(openNewEToll); //creating thread

th.SetApartmentState(ApartmentState.STA); //setting thread as STA

th.Start(); //start the thread

}

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

{

this.Close(); //command for closing the application

th = new Thread(openNewLoad); //creating thread

th.SetApartmentState(ApartmentState.STA); //setting thread as STA

th.Start(); //start the thread

}

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

{

this.Refresh();

}

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

{

this.Close(); //command for closing the application

th = new Thread(openNewAdmins); //creating thread

th.SetApartmentState(ApartmentState.STA); //setting thread as STA

th.Start(); //start the thread

}

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

{

this.Close(); //command for closing the application

th = new Thread(openNewForm1); //creating thread

th.SetApartmentState(ApartmentState.STA); //setting thread as STA

th.Start(); //start the thread

}

private void openNewDashBoad(object obj)

{

Application.Run(new Dashboard()); //Run Dashboard Tab

}

private void openNewEToll(object obj)

{

Application.Run(new EToll()); //Run EToll Tab

}

private void openNewLoad(object obj)

{

Application.Run(new Load()); //Run Load Tab

}

private void openNewAdmins(object obj)

{

Application.Run(new Admins()); //Run Admins' Info Tab

}

private void openSplashScreen(object obj)

{

Application.Run(new SplashScreen()); //Run SplashScreen

}

private void openNewForm1(object obj)

{

Application.Run(new Form1()); //Run Form1 or Login Tab

}

private void Drivers\_Load(object sender, EventArgs e)

{

timer1.Start();

refresh();

}

private void textBox1\_TextChanged(object sender, EventArgs e)

{

}

private void label6\_Click(object sender, EventArgs e)

{

}

private void dataGridView\_MouseClick(object sender, MouseEventArgs e)

{

try

{

idNumber = dataGridView.SelectedRows[0].Cells[0].Value.ToString().Trim();

txtUsername.Text = dataGridView.SelectedRows[0].Cells[1].Value.ToString();

txtPassword.Text = dataGridView.SelectedRows[0].Cells[2].Value.ToString();

txtGivenName.Text = dataGridView.SelectedRows[0].Cells[3].Value.ToString();

txtMiddleName.Text = dataGridView.SelectedRows[0].Cells[4].Value.ToString();

txtLastName.Text = dataGridView.SelectedRows[0].Cells[5].Value.ToString();

txtPhoneNumber.Text = dataGridView.SelectedRows[0].Cells[6].Value.ToString();

txtTelephoneNumber.Text = dataGridView.SelectedRows[0].Cells[7].Value.ToString();

dtpBirthDate.Text = dataGridView.SelectedRows[0].Cells[8].Value.ToString();

cboGender.Text = dataGridView.SelectedRows[0].Cells[9].Value.ToString();

txtNationality.Text = dataGridView.SelectedRows[0].Cells[10].Value.ToString();

txtHomeAddress.Text = dataGridView.SelectedRows[0].Cells[11].Value.ToString();

txtCityAddress.Text = dataGridView.SelectedRows[0].Cells[12].Value.ToString();

dateRegistered = dataGridView.SelectedRows[0].Cells[13].Value.ToString();

cboStatus.Text = dataGridView.SelectedRows[0].Cells[14].Value.ToString();

txtPurpose.Text = dataGridView.SelectedRows[0].Cells[15].Value.ToString();

txtBalance.Text = dataGridView.SelectedRows[0].Cells[16].Value.ToString();

cboAccountStatus.Text = dataGridView.SelectedRows[0].Cells[17].Value.ToString();

btnAdd.Enabled = false;

btnUpdate.Enabled = true;

btnDelete.Enabled = true;

}

catch (Exception ex1)

{

MessageBox.Show(ex1.Message);

clear();

}

}

private void timer1\_Tick(object sender, EventArgs e)

{

this.dateNow = time.ToString("MM-dd-yyyy");

this.timeNow = time.ToString("H:mm:s");

}

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

{

try

{

FirebaseResponse res1 = client.Get(@"USER\_COUNTER/USER\_COUNTER");

Counter = int.Parse(res1.ResultAs<String>());

for (int i = Counter; i >= 0; i--)

{

if (Counter < 10) { idHolder = "LNC01-01-000" + i.ToString(); }

else if (Counter < 100) { idHolder = "LNC01-01-00" + i.ToString(); }

else if (Counter < 1000) { idHolder = "LNC01-01-0" + i.ToString(); }

else { idHolder = "LNC01-01-" + Counter.ToString(); }

var res3 = client.Get(@"USER INFO/" + idHolder);

MyUser std = res3.ResultAs<MyUser>();

MessageBox.Show(std.USERNAME.ToString());

}

if (taken == 0)

{

if (txtUsername.Text != "" && txtPassword.Text != "" && txtGivenName.Text != "" && txtLastName.Text != "" && txtPhoneNumber.Text.Length == 17 && cboGender.Text != "" && txtNationality.Text != "" && txtHomeAddress.Text != "" && txtCityAddress.Text != "" && cboStatus.Text != "" && txtBalance.Text != "")

{

if (txtPhoneNumber.Text.Length == 17)

{

Counter++;

MyUser std1 = new MyUser()

{

USER\_COUNTER = Counter.ToString()

};

var setter1 = client.Set("USER\_COUNTER", std1);

if (Counter < 10) { idNumber = "LNC01-01-000" + Counter.ToString(); }

else if (Counter < 100) { idNumber = "LNC01-01-00" + Counter.ToString(); }

else if (Counter < 1000) { idNumber = "LNC01-01-0" + Counter.ToString(); }

else { idNumber = "LNC01-01-" + Counter.ToString(); }

MyUser std2 = new MyUser()

{

ID\_NUMBER = idNumber.ToString().Trim()

};

var setter2 = client.Set("USER\_ID/" + Counter.ToString(), std2);

if (cboStatus.Text == "Home Owner")

{

purpose = "Enter";

}

else if (cboStatus.Text == "Visitor")

{

purpose = txtPurpose.Text;

}

else

{

purpose = "Pass";

}

MyUser std3 = new MyUser()

{

USERNAME = txtUsername.Text,

PASSWORD = txtPassword.Text,

GIVEN\_NAME = txtGivenName.Text,

MIDDLE\_NAME = txtMiddleName.Text,

LAST\_NAME = txtLastName.Text,

PHONE\_NUMBER = txtPhoneNumber.Text,

TELEPHONE\_NUMBER = txtTelephoneNumber.Text,

BIRTHDATE = dtpBirthDate.Text,

GENDER = cboGender.Text,

NATIONALITY = txtNationality.Text,

HOME\_ADDRESS = txtHomeAddress.Text,

CITY\_ADDRESS = txtCityAddress.Text,

DATE\_AND\_TIME = dateNow + " :: " + timeNow,

STATUS = cboStatus.Text,

BALANCE = txtBalance.Text,

ACCOUNT\_STATUS = cboAccountStatus.Text,

PURPOSE = purpose

};

var setter3 = client.Set("USER INFO/" + idNumber, std3);

MessageBox.Show("Data Inserted Successfully!");

clear();

refresh();

}

else { MessageBox.Show("Phone Number is Incomplete!"); }

}

else

{

MessageBox.Show("Please fill out all the field!");

}

}

else

{

MessageBox.Show("Username Already Taken!");

clear();

}

}

catch (Exception ex1)

{

MessageBox.Show(ex1.Message);

clear();

}

}

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

{

try{

if (txtUsername.Text != "" && txtPassword.Text != "" && txtGivenName.Text != "" && txtLastName.Text != "" && cboGender.Text != "" && txtNationality.Text != "" && txtHomeAddress.Text != "" && txtCityAddress.Text != "" && cboStatus.Text != "" && txtBalance.Text != "")

{

if (txtPhoneNumber.Text.Length == 17)

{

if (cboStatus.Text == "Home Owner")

{

purpose = "Enter";

}

else if (cboStatus.Text == "Visitor")

{

purpose = txtPurpose.Text;

}

else

{

purpose = "Pass";

}

MyUser std3 = new MyUser()

{

USERNAME = txtUsername.Text,

PASSWORD = txtPassword.Text,

GIVEN\_NAME = txtGivenName.Text,

MIDDLE\_NAME = txtMiddleName.Text,

LAST\_NAME = txtLastName.Text,

PHONE\_NUMBER = txtPhoneNumber.Text,

TELEPHONE\_NUMBER = txtTelephoneNumber.Text,

BIRTHDATE = dtpBirthDate.Text,

GENDER = cboGender.Text,

NATIONALITY = txtNationality.Text,

HOME\_ADDRESS = txtHomeAddress.Text,

CITY\_ADDRESS = txtCityAddress.Text,

DATE\_AND\_TIME = dateRegistered,

STATUS = cboStatus.Text,

BALANCE = txtBalance.Text,

ACCOUNT\_STATUS = cboAccountStatus.Text,

PURPOSE = purpose

};

var setter3 = client.Set("USER INFO/" + idNumber, std3);

MessageBox.Show("Data Updated Successfully!");

clear();

refresh();

}

else { MessageBox.Show("Phone Number is Incomplete!"); }

}

else

{

MessageBox.Show("Please fill out all the field!");

}

}

catch (Exception ex1)

{

MessageBox.Show(ex1.Message);

clear();

}

}

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

{

try

{

if (cboStatus.Text == "Home Owner")

{

purpose = "Enter";

}

else if (cboStatus.Text == "Visitor")

{

purpose = txtPurpose.Text;

}

else

{

purpose = "Pass";

}

MyUser std3 = new MyUser()

{

USERNAME = txtUsername.Text,

PASSWORD = txtPassword.Text,

GIVEN\_NAME = txtGivenName.Text,

MIDDLE\_NAME = txtMiddleName.Text,

LAST\_NAME = txtLastName.Text,

PHONE\_NUMBER = txtPhoneNumber.Text,

TELEPHONE\_NUMBER = txtTelephoneNumber.Text,

BIRTHDATE = dtpBirthDate.Text,

GENDER = cboGender.Text,

NATIONALITY = txtNationality.Text,

HOME\_ADDRESS = txtHomeAddress.Text,

CITY\_ADDRESS = txtCityAddress.Text,

DATE\_AND\_TIME = dateRegistered,

STATUS = cboStatus.Text,

BALANCE = txtBalance.Text,

ACCOUNT\_STATUS = "DEACTIVATED",

PURPOSE = purpose

};

var setter3 = client.Set("USER INFO/" + idNumber, std3);

MessageBox.Show("Data Deleted Successfully!");

clear();

refresh();

}

catch (Exception ex1)

{

MessageBox.Show(ex1.Message);

clear();

}

}

private void label15\_Click(object sender, EventArgs e)

{

}

private void comboBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

}

private void panel2\_Paint(object sender, PaintEventArgs e)

{

}

private void cboStatus\_TextChanged(object sender, EventArgs e)

{

if (cboStatus.Text == "Visitor")

{

txtPurpose.Enabled = true;

}

else

{

txtPurpose.Enabled = false;

}

}

}

}