using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

using System.IO;

namespace CSLG.CTalk.UI

{

public partial class FrmLogin : Form

{

public FrmLogin()

{

InitializeComponent();

}

List<string> userInfo = new List<string>();

private Model.userInfo \_user;

private int \_imageIndex;

string mainPath;

List<string> pwds = new List<string>();

List<int> stations = new List<int>();

List<int> isRememberPwds = new List<int>();

List<DateTime> dateTimes = new List<DateTime>();

public Boolean \_isHide = true;

private int \_station;

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

{

LoginToMain();

string userID = this.cmbUser.Text;

int isAutoLogin = 0;

int isRemmenberPwd = 0;

BLL.userInfo bllUserInfo = new BLL.userInfo();

Model.userInfo user = bllUserInfo.GetModel(Convert.ToInt32(userID));

string pwd = " ";

// int userState=this.

///获取登录时间

DateTime LoginTime = DateTime.Now;

if (cbPassword.Checked)

pwd = user.userPWD;

if (cbPassword.Checked)

isRemmenberPwd = 1;

if (cbLogin.Checked)

{

isAutoLogin = 1;

}

StreamReader sr = new StreamReader(mainPath + "record.txt");

Boolean flag = false;

while (sr.Peek() > -1)

{

string tmp = sr.ReadLine();

string[] str = tmp.Split('-');

if (user.userID.ToString() == str[0])

{

sr.Close();

updateRecord(user.userID.ToString(), pwd, isAutoLogin, isRemmenberPwd, \_station, LoginTime);

flag = true;

break;

}

}

if (!flag)

{

sr.Close();

recordSuccessUser(user.userID.ToString(), pwd, isAutoLogin, isRemmenberPwd, \_station, LoginTime);

}

}

private void updateRecord(string id, string pwd, int isAutoLogin, int isRememberPwd, int station, DateTime dt)

{

StreamReader sr = new StreamReader(mainPath + @"record.txt");

StreamWriter sw = new StreamWriter(mainPath + @"temp.txt", true);

while (sr.Peek() > -1)

{

string tmp = sr.ReadLine();

string[] str = tmp.Split('-');

if (id == str[0])

{

sw.WriteLine(id + "-" + pwd + "-" + isAutoLogin + "-" + isRememberPwd + "-" + station + "-" + dt);

//return;

continue;

}

sw.WriteLine(tmp);

}

sw.Close();

sr.Close();

File.Copy(mainPath + @"\temp.txt", mainPath + @"record.txt", true);

File.Delete(mainPath + @"temp.txt");

}

private void recordSuccessUser(string id, string pwd, int isAutoLogin, int isRememberPwd, int station, DateTime dt)

{

StreamWriter sw = new StreamWriter(mainPath + @"record.txt", true);

sw.WriteLine(id + "-" + pwd + "-" + isAutoLogin + "-" + isRememberPwd + "-" + station + "-" + dt);

sw.Close();

}

private int temp = 1;

private void LoginToMain()

{

string userID = this.cmbUser.Text;

string userPWD = this.textBox1.Text;

if (userID == "" || userPWD == "")

{

return;

}

BLL.userInfo bllUserInfo = new BLL.userInfo();

Model.userInfo user = bllUserInfo.GetModel(Convert.ToInt32(userID));

if (userPWD != user.userPWD)

{

return;

}

user.userID = Convert.ToInt32(userID);

user.userState = temp;

bllUserInfo.Update(user);

FrmMain frm = new FrmMain(user, this);

frm.Show();

this.Hide();

}

private void CreateFrmMain(int id)

{

BLL.userInfo bllUserInfo = new BLL.userInfo();

Model.userInfo user = bllUserInfo.GetModel(id);

FrmMain frm = new FrmMain(user, this);

user.userState = 1;

bllUserInfo.Update(user);

frm.Show();

this.\_isHide = false;

this.Hide();

}

private int getLastLoginUser(List<DateTime> dts)

{

int max = 0;

for (int i = 1; i < dts.Count; i++)

{

if (dts[max] < dts[i])

{

max = i;

}

}

return max;

}

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

{

ShowList();

StreamReader sr = null;

mainPath = AppDomain.CurrentDomain.BaseDirectory + @"app\";

DirectoryInfo dir = Directory.CreateDirectory(mainPath);

if (!dir.Exists)

{

dir.Create();

}

string filePath = mainPath + @"record.txt";

if (!File.Exists(filePath))

{

File.Create(filePath);

}

try

{

this.cmbUser.Items.Clear();

sr = new StreamReader(filePath);

while (sr.Peek() > -1)

{

string tmp = sr.ReadLine();

string[] str = tmp.Split('-');

int id = Convert.ToInt32(str[0]);

string pwd = str[1];

int isAutoLogin = Convert.ToInt32(str[2]);

int isRememberPwd = Convert.ToInt32(str[3]);

int station = Convert.ToInt32(str[4]);

DateTime time = Convert.ToDateTime(str[5]);

this.cmbUser.Items.Add(str[0]);

pwds.Add(pwd);

stations.Add(station);

isRememberPwds.Add(isRememberPwd);

dateTimes.Add(time);

if (isAutoLogin == 1)

{

CreateFrmMain(id);

break;

}

}

int lastIndex = getLastLoginUser(dateTimes);

this.cmbUser.SelectedIndex = lastIndex;

}

catch (Exception)

{

throw;

}

finally

{

sr.Close();

}

}

public void ShowList()

{

BLL.userInfo bllUserInfo = new BLL.userInfo();

List<Model.userInfo> users = bllUserInfo.GetModelList("1=1");

this.cmbUser.Items.Clear();

foreach (Model.userInfo u in users)

{

this.cmbUser.Items.Add(u.userID);

}

}

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

{

BLL.userInfo blluser = new BLL.userInfo();

int id = Convert.ToInt32(this.cmbUser.Text);

Model.userInfo bll = blluser.GetModel(id);

FrmMain frm = new FrmMain();

this.pictureBox1.Image = frm.imageList1.Images[Convert.ToInt32( bll.userHeadImage)];

this.picLogin.Image = frm.imageList2.Images[0];

foreach (string s in userInfo)

{

string[] res = s.Split(' ');

if (res[0] == this.cmbUser.Text)

{

this.textBox1 .Text = res[1];

}

}

textBox1.Text = pwds[cmbUser.SelectedIndex].ToString();

int station = stations[cmbUser.SelectedIndex];

\_station = station;

int isRememberPwd = isRememberPwds[cmbUser.SelectedIndex];

if (isRememberPwd > 0)

{

this.cbPassword.Checked = true;

}

else

this.cbPassword.Checked = false;

}

private void linkLabel1\_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)

{

FrmAddUser frm = new FrmAddUser(this );

frm.ShowDialog();

}

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

{

this.contextMenuStrip1.Show(Control.MousePosition);

}

private void toolStripMenuItem2\_Click(object sender, EventArgs e)

{

FrmMain frm=new FrmMain();

temp = 1;

this.picLogin.Image= frm.imageList2.Images[0];

}

private void toolStripMenuItem3\_Click(object sender, EventArgs e)

{

FrmMain frm = new FrmMain();

temp = 3;

this.picLogin.Image = frm.imageList2.Images[2];

}

private void toolStripMenuItem4\_Click(object sender, EventArgs e)

{

FrmMain frm = new FrmMain();

temp = 5;

this.picLogin.Image = frm.imageList2.Images[4];

}

private void toolStripMenuItem5\_Click(object sender, EventArgs e)

{

FrmMain frm = new FrmMain();

temp = 4;

this.picLogin.Image = frm.imageList2.Images[3];

}

private void toolStripMenuItem6\_Click(object sender, EventArgs e)

{

FrmMain frm = new FrmMain();

temp = 2;

this.picLogin.Image = frm.imageList2.Images[1];

}

private void linkLabel2\_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)

{

}

}

}