**HỌ TÊN:** Hồ Chí Nguyên

**STT:** 36

**MSSV:** 2251050048

**Mục lục**

[**1.** **Form đăng kí - đăng nhập** 3](#_Toc164590895)

[Yêu cầu 1: Nhấn picEye thì đổi ảnh mắt đóng thành mắt mở và ngược lại 5](#_Toc164590896)

[Yêu cầu 2: Nhấn lbQuenMK thì hiện thông báo trợ giúp 6](#_Toc164590897)

[Yêu cầu 3: 6](#_Toc164590898)

[Nhấn btnLogin để đăng nhập, nếu thành công thì hiện thông báo thành công. 6](#_Toc164590899)

[Nếu 1 trong 2 ô txtUser và txtPassWord chưa nhập hoặc nhập sai thì báo lỗi. 6](#_Toc164590900)

[Nội dung tài khoản mật khẩu được đọc từ file .txt. 6](#_Toc164590901)

[Yêu cầu 4: 8](#_Toc164590902)

[Nhấn btnSignUp để đăng kí, nếu thành công thì đóng Form formLogin và mở Form isRobot. Dữ liệu txtUser và txtPassWord được ghi lại vào file .txt 8](#_Toc164590903)

[Nếu 1 trong 2 ô txtUser và txtPassWord chưa nhập thì báo lỗi. 8](#_Toc164590904)

[Đọc file .txt, nếu txtUser và txtPassWord đã tồn tại thì báo lỗi 8](#_Toc164590905)

[Yêu cầu 5: Css màu, hiệu ứng 9](#_Toc164590906)

[**2.** **Form Xác thực** 10](#_Toc164590907)

[Yêu cầu 1: Nhấn giữ hình ảnh và kéo thả vào panel, nếu đúng thì đóng form isRobot, thông báo xác thực thành công và mở form formLogin. 10](#_Toc164590908)

[**3.** **Form Chính (Trang chủ)** 11](#_Toc164590909)

[Yêu cầu 1: Nhấn vào hình ảnh picAvatar thì mở thư mục chọn hình từ máy, sau đó thay hình vừa chọn vào picAvatar 16](#_Toc164590910)

[Yêu cầu 2: Nhấn vào button btnTaiChinh và btnTaiKhoanNganHang thì mở form tương ứng (dạng MDI) 16](#_Toc164590911)

[Yêu cầu 3: Sau 3 giây thay đổi hình ảnh của picQuangCao\_36\_Nguyen một lần 17](#_Toc164590912)

[Yêu cầu 4: Thực hiện hiển thị thời gian ngày tháng năm giờ hiện tại cho label lbDateTime\_36\_Nguyen 18](#_Toc164590913)

[Yêu cầu 5: Mở form lên thêm sẵn nội dung số thứ tự cho các dòng của gridBangThoiKhoaBieu 19](#_Toc164590914)

[Yêu cầu 6: Thực hiện các chức năng trên thanh menustrip, *gồm:* 19](#_Toc164590915)

[Làm mới, lưu, mở bảng gridBangThoiKhoaBieu 19](#_Toc164590916)

[Đóng ứng dụng (có hỏi lại bạn có chắc) 19](#_Toc164590917)

[cài đặt font chữ, màu chữ, màu control, màu nền, mở tắt nhạc 19](#_Toc164590918)

[Mở thông báo trợ giúp 19](#_Toc164590919)

[Thoát tài khoản (có hỏi lại bạn có chắc) 19](#_Toc164590920)

[**4.** **Form quản lí tài chính chi tiêu** 25](#_Toc164590921)

[Yêu cầu 1: nhập số tiền vào các TextBox: txtNguonTien1, txtNguonTien2, txtNguonTien3, txtNguonTien4, txtSoTien\_36\_Nguyen. Nội dung phải là số không được nhập chữ. 29](#_Toc164590922)

[Yêu cầu 2: Đặt sẵn nội dung mặc định cho các comboBox: cbNguonTien1, cbNguonTien2, cbNguonTien3, cbNguonTien4, cbMucDich\_36\_Nguyen 29](#_Toc164590923)

[Yêu cầu 3: Khi nhập số tiền ở các textbox: txtNguonTien1, txtNguonTien2, txtNguonTien3, txtNguonTien4 thì số tiền ở txtTong tự động cập nhật (cộng 4 nguồn tiền lại) 30](#_Toc164590924)

[Yêu cầu 4: Nhấn Button btnXuatDanhSach\_36\_Nguyen thì xuất tất cả nội dung của listBoxChiTieu\_36\_Nguyen thành file 31](#_Toc164590925)

[Yêu cầu 5: Nhấn vào Button btnThem\_36\_Nguyen thì thêm nội dung nguồn tiền, tổng tiền, số tiền chi tiêu, mục đích mô tả và tiền còn lại. Cập nhật số tiền ở TextBox txtTong 32](#_Toc164590926)

[**5.** **Form quản lí tài khoản ngân hàng** 34](#_Toc164590927)

[Yêu cầu 1: Khi mở form lên mặc định giá trị cho comboBox cbChonTaiKhoanNganHang và ListTaiKhoanNganHang 37](#_Toc164590928)

[Yêu cầu 2: Nhấn vào Button btnThemNganHang thì thêm nội dung từ txtTenTaiKhoanNganHang và txtMatKhauNganHang vào ListTaiKhoanNganHang. Nếu 1 trong 2 ô txt không có nội dung thì báo lỗi 38](#_Toc164590929)

[Yêu cầu 3: Nhấn vào Button btnTimKiem thì tìm kiếm nội dung trong ListTaiKhoanNganHang. Nếu không tìm thấy báo lỗi, nếu tìm thấy thì đánh dấu 40](#_Toc164590930)

[Yêu cầu 4: Nhấn vào button btnXuatFile xuất danh sách nội dung trong ListTaiKhoanNganHang thành file txt 41](#_Toc164590931)

[Yêu cầu 5: Nhấn vào Button btnMoFile mở file txt danh sách tài khoản có sẵn 42](#_Toc164590932)

[Yêu cầu 6: Nhấn vào button btnXoaAll Xoá tất cả nội dung trong ListTaiKhoanNganHang 43](#_Toc164590933)

[Yêu cầu 7: Nhấn vào button btnXoa thì xoá một nội dung đang chọn trong ListTaiKhoanNganHang. Nếu chưa chọn thì báo lỗi 43](#_Toc164590934)

**Time Schedule and Expense Management**

***Ý tưởng:*** *App giúp quản lí thiết lập thời gian biểu cá nhân, quản lí tài khoản ngân và chi tiêu cá nhân, cập nhật tin tức tài chính*

1. **Form đăng kí - đăng nhập**

A screenshot of a login screen

Description automatically generated

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Control** | **Loại** | **Thuộc tính** | **Tên hàm xử lí** |
| **1** | FormLogin | Form | Autosize: true  AcceptButton: btnLogin  Icon: nhập từ máy  BackgroundImageL nhập từ máy  StartPosition: Centergreen  BorderStyle: Fix3D  Text: Login |  |
| **2** | picbanner | PictureBox | BackColor: Limegreen  SizeMode: Normal  BoderStyle: FixSingle |  |
| **3** | lbCauNoi | Label | Text: Cần cù thì bù siêng năng, có tiết kiệm thì mới có ăn  BackColor: PeachPuff  AutoSize: true |  |
| **4** | panelLogin | Panel | BackColor: AntiqueWhite  AutoSize: true  BoderStyle: FixSingle |  |
| **5** | lbLogin | Label | BackColor: Lime  AutoSize: true  BoderStyle: FixSingle  Text: LOGIN |  |
| **6** | lbUser | Label | Text: Tên đăng nhập  AutoSize: true |  |
| **7** | txtUser | TextBox | BorderStyle: Fix3D  TextAlign: Center  AcceptsTab: true  AcceptReturn: true | txtUser\_MouseDown  txtUser\_MouseLeave |
| **8** | lbPassWord | Label | Text: Mật khẩu  AutoSize: true |  |
| **9** | txtPassWord | TextBox | BorderStyle: Fix3D  TextAlign: Center  AcceptsTab: true  AcceptReturn: true | txtPassWork\_MouseDown  txtPassWork\_MouseLeave |
| **10** | picEye | PictureBox | BackColor: AntiqueWhite  Cursor: hand  SizeMode: zoom | picEye\_Click |
| **12** | lbQuenMK | Label | Text: Quên mật khẩu?  Italic: true | lbQuenMK\_Click  lbQuenMK\_MouseHover  lbQuenMK\_MouseLeave |
| **13** | btnSignUp | Button | Text: Đăng kí  AutoSize: true  Bold: true  TextAlign: MiddleCenter  BackColor: : AntiqueWhite | btnSignUp\_Click |
| **14** | btnLogin | Button | Text: Đăng nhập  AutoSize: true  Bold: true  TextAlign: MiddleCenter  BackColor: : AntiqueWhite | btnLogin\_Click |

## Yêu cầu 1: Nhấn picEye thì đổi ảnh mắt đóng thành mắt mở và ngược lại

private bool isEyeOpen = false;

//Phương thức thay đổi ảnh mật khẩu hiện và ẩn

private void ChangeEyeImage()

{

if (isEyeOpen)

{

picEye.Image = Properties.Resources.EyeOpen;

}

else

{

picEye.Image = Properties.Resources.Eye;

}

}

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

{

// Khi PictureBox picEye được nhấn, chuyển đổi hiển thị mật khẩu

if (txtPassWord.PasswordChar == '\*')

txtPassWord.PasswordChar = '\0'; // Hiển thị mật khẩu

else

txtPassWord.PasswordChar = '\*'; // Ẩn mật khẩu

// nếu isEyeOpen là false sẽ thành true và ngược lại

isEyeOpen = !isEyeOpen;

ChangeEyeImage();

}

## Yêu cầu 2: Nhấn lbQuenMK thì hiện thông báo trợ giúp

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

{

MessageBox.Show("Vui lòng liên hệ hỗ trợ qua email: " +

"nguyen.hochi2004@gmail.com\n" +

"Hoặc liên hệ qua hot line: 0332636829", "Help", MessageBoxButtons.OK, MessageBoxIcon.Information);

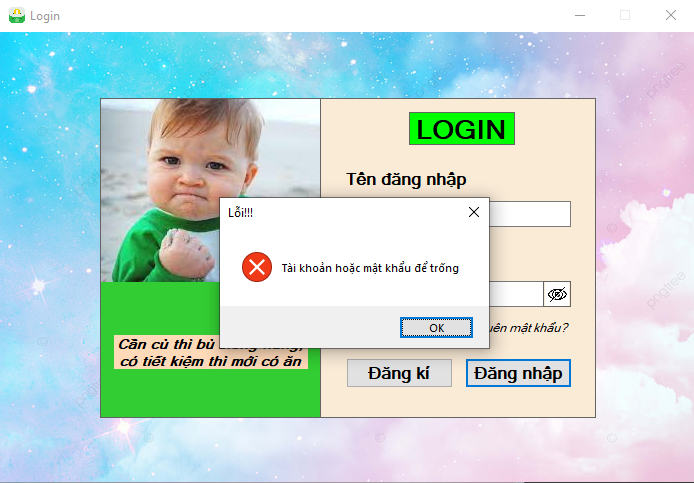
}

## Yêu cầu 3:

## Nhấn btnLogin để đăng nhập, nếu thành công thì hiện thông báo thành công.

## Nếu 1 trong 2 ô txtUser và txtPassWord chưa nhập hoặc nhập sai thì báo lỗi.

## Nội dung tài khoản mật khẩu được đọc từ file .txt.

A screenshot of a computer screen

Description automatically generated

//Phương thức đọc file tài khoản (accounts) có tồn tại hay không

private bool CheckCredentials(string username, string password)

{

string filePath = "C:\\Users\\NGUYEN\\Time\_Schedule\_and\_Expense\_Management\\Quản Lý Thời Gian Biểu và Chi Tiêu\\resources\\accounts.txt";

try

{

if (File.Exists(filePath))

{

string[] lines = File.ReadAllLines(filePath);

foreach (string line in lines)

{

string[] parts = line.Split(',');

if (parts.Length == 2 && parts[0].Trim() == username && parts[1].Trim() == password)

{

return true; // Tìm thấy tài khoản và mật khẩu trùng khớp

}

}

}

else

{

//MessageBox.Show("Không tìm thấy tệp lưu trữ tài khoản!", "Lỗi");

return false; // Không tìm thấy tài khoản hoặc mật khẩu

}

}

catch (Exception ex)

{

Console.WriteLine("Lỗi: " + ex.Message);

}

return false; // Không tìm thấy tài khoản hoặc mật khẩu

}

//Login

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

{

string username = txtUser.Text;

string password = txtPassWord.Text;

bool isAuthenticated = CheckCredentials(username, password);

if (isAuthenticated)

{

MessageBox.Show("Đăng nhập thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

this.Hide();

FormMain frmMain = new FormMain();

frmMain.ShowDialog();

this.Close();

}

else if (txtUser.Text == "Nguyen" && txtPassWord.Text == "36")

{

MessageBox.Show("Đăng nhập thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

this.Hide();

FormMain frmMain = new FormMain();

frmMain.ShowDialog();

this.Close();

}

else if (txtUser.Text == "" || txtPassWord.Text == "")

MessageBox.Show("Tài khoản hoặc mật khẩu để trống", "Lỗi!!!", MessageBoxButtons.OK, MessageBoxIcon.Error);

else

{

MessageBox.Show("Không đúng tên người dùng / mật khẩu!!!", "Lỗi!!!", MessageBoxButtons.OK, MessageBoxIcon.Error);

txtUser.Focus();

}

}

## Yêu cầu 4:

## Nhấn btnSignUp để đăng kí, nếu thành công thì đóng Form formLogin và mở Form isRobot. Dữ liệu txtUser và txtPassWord được ghi lại vào file .txt

## Nếu 1 trong 2 ô txtUser và txtPassWord chưa nhập thì báo lỗi.

## Đọc file .txt, nếu txtUser và txtPassWord đã tồn tại thì báo lỗi

A screenshot of a computer

Description automatically generated

//Ghi file tài khoản và mật khẩu khi đăng kí (Sign Up)

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

{

string username = txtUser.Text;

string password = txtPassWord.Text;

bool isAuthenticated = CheckCredentials(username, password);

if (isAuthenticated)

{

MessageBox.Show("Tài khoản đã tồn tại", "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

else if (txtUser.Text == "" || txtPassWord.Text == "")

MessageBox.Show("Tài khoản hoặc mật khẩu để trống", "Lỗi!!!", MessageBoxButtons.OK, MessageBoxIcon.Error);

else

{

try

{

// Đường dẫn đến tệp tin, khi sử dụng ký tự @, không cần escape các ký tự \ trong chuỗi đường dẫn

string filePath = @"C:\Users\NGUYEN\Time\_Schedule\_and\_Expense\_Management\Quản Lý Thời Gian Biểu và Chi Tiêu\resources\accounts.txt";

// Mở tệp tin để ghi

using (StreamWriter sw = File.AppendText(filePath))

{

// Ghi tên người dùng và mật khẩu vào tệp tin

sw.WriteLine(/\*"\n" + \*/username + "," + password);

}

this.Hide();

IsRobot robot = new IsRobot();

robot.ShowDialog();

this.Close();

}

catch (Exception ex)

{

MessageBox.Show("Xảy ra lỗi khi đăng kí tài khoản: " + ex.Message, "Lỗi!!!", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

}

## Yêu cầu 5: Css màu, hiệu ứng

// in nghiêng ô quên mk khi hover

private void lbQuenMK\_MouseHover(object sender, EventArgs e)

{

lbQuenMK.Font = new Font(lbQuenMK.Font, FontStyle.Underline);

}

private void lbQuenMK\_MouseLeave(object sender, EventArgs e)

{

lbQuenMK.Font = new Font(lbQuenMK.Font, FontStyle.Italic);

}

//Hover vào có màu ở ô Đăng nhập

private void txtUser\_MouseDown(object sender, MouseEventArgs e)

{

txtUser.BackColor = Color.LightGreen;

}

private void txtUser\_MouseLeave(object sender, EventArgs e)

{

txtUser.BackColor = Color.White;

}

//Hover vào có màu ở ô Mật khẩu

private void txtPassWork\_MouseDown(object sender, MouseEventArgs e)

{

txtPassWord.BackColor = Color.LightGreen;

}

private void txtPassWork\_MouseLeave(object sender, EventArgs e)

{

txtPassWord.BackColor = Color.White;

}

1. **Form Xác thực**

A person standing in front of a tree

Description automatically generatedA screenshot of a computer

Description automatically generated

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên control** | **Loại** | **Thuộc tính** | **Hàm xử lí** |
| **1** | isRobot | Form | Autosize: true  Icon: nhập từ máy  BackgroundImageL nhập từ máy |  |
| **2** | picIsRobot | PictureBox | BorderStyle: FixedSingle  Cursor: Hand  SizeMode: Zoom | picIsRobot\_MouseDown  picIsRobot\_MouseMove  picIsRobot\_MouseUp |
| **3** | panIsRobot | Panel | BackColor: Pink  AutoSize: false |  |
| **4** | lbIsRobot | Label | Text: Kéo hình ảnh vào ô trống để xác minh bạn không phải là Robot  ForeColor: OrangeRed  Italic: true  AutoSize: true |  |

## Yêu cầu 1: Nhấn giữ hình ảnh và kéo thả vào panel, nếu đúng thì đóng form isRobot, thông báo xác thực thành công và mở form formLogin.

Point p0ldPoint; //Biến lưu giữ vị trí chuột

private void picIsRobot\_MouseDown(object sender, MouseEventArgs e)

{

p0ldPoint = e.Location;

}

private void picIsRobot\_MouseMove(object sender, MouseEventArgs e)

{

if (e.Button == MouseButtons.Left)

{

int dx = e.X - p0ldPoint.X;

int dy = e.Y - p0ldPoint.Y;

picIsRobot.Left += dx;

picIsRobot.Top += dy;

}

}

private void picIsRobot\_MouseUp(object sender, MouseEventArgs e)

{

if(panIsRobot.Bounds.Contains(picIsRobot.Bounds))

{

MessageBox.Show("Đăng kí tài khoản thành công, mời bạn đăng nhập ứng dụng", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

this.Hide();

FormLogin login = new FormLogin();

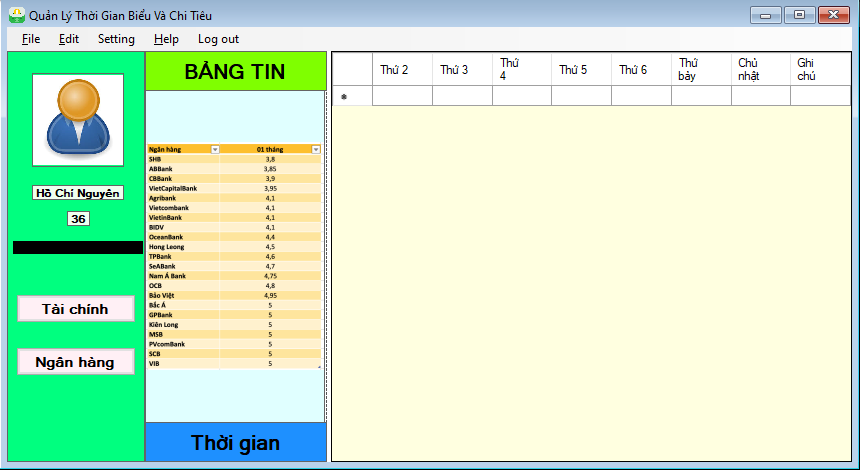
login.ShowDialog();

this.Close();

}

}

1. **Form Chính (Trang chủ)**

****

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Control** | **Loại** | **Thuộc tính** | **Hàm xử lí** |
| **1** | FormMain | Form | WindowState: Maximized  Text: Quản lí thời gian biểu và chi tiêu  AutoSize: true  Icon: nhập từ máy  StartPosition: Centergreen | FormMain\_Load |
| **2** | TaskBar | Panel | BackColor: SpringGreen  AutoSize: False  BorderStyle: FixSingle  Dock: Left |  |
| **3** | picAvatar | PictureBox | BorderStyle: FixSingle  Cursor: Hand  Image: nhập từ máy  SizeMode: Zoom  Archor: Top, Bottom, Left, Right | picAvatar\_Click |
| **4** | lbHoTen | Label | BackColor: Honeydew  BorderStyle: FixSingle  AutoSize: true  Text: Hồ Chí Nguyên |  |
| **5** | lbId | Label | BackColor: Honeydew  BorderStyle: FixSingle  AutoSize: true  Text: 36 |  |
| **6** | lbDuongKe | Label | BackColor: InactiveCaptionText |  |
| **7** | btnTaiChinh | Button | BackColor: LavenderBlush  Text: Tài chính  TextAlign: MiddleCenter  Cursor: Hand | btnTaiChinh\_Click |
| **8** | btnTaiKhoan  Ngan Hang | Button | BackColor: LavenderBlush  Text: Ngân hàng  TextAlign: MiddleCenter  Cursor: Hand | btnTaiKhoanNganHang\_Click |
| **9** | SpitContainer | Chia form thành 2 phần một bên là picQuangCao\_36\_Nguyen, bên còn lại là gridBangThoiKhoaBieu | | |
| **10** | picQuangCao  \_36\_Nguyen | PictureBox | Image: Nhập từ máy  BackColor: LightCyan  BorderStyle: FixSingle  SizeMode: Zoom |  |
| **11** | lbBangTin  \_36\_Nguyen | Label | BackColor: Chartreuse  Text: BẢNG TIN  BorderStyle: FixSingle  TextAlign: MiddleCenter  Dock: Top |  |
| **11** | lbDateTime  \_36\_Nguyen | Label | BackColor: DodgerBlue  Text: BẢNG TIN  BorderStyle: FixSingle  TextAlign: MiddleCenter  Dock: Bottom |  |
| **12** | gridBang  ThoiKhoaBieu | DataGridView | BackColor: LightYellow  BorderStyle: FixSingle  GridColor: ControlDark  Dock: Fill |  |
| **13** | menuStrip1 | MenuStrip | AutoSize: true  Dock: top |  |
| **14** | fileToolStrip  MenuItem1 | MenuStripItem | Text: New  TextAlign: MiddleCenter  BackColor: Control |  |
| **15** | newToolStrip  MenuItem2 | newToolStripMenuItem2\_Click |
| **16** | openToolStrip  MenuItem2 | openToolStripMenuItem2\_Click |
| **17** | saveToolStrip  MenuItem2 | saveToolStripMenuItem2\_Click |
| **18** | exitToolStrip  MenuItem2 | exitToolStripMenuItem2\_Click |
| **19** | editToolStrip  MenuItem1 |  |
| **20** | settingToolStrip  MenuItem |  |
| **21** | fontToolStrip  MenuItem |  |
| **22** | fontEditToolStrip  MenuItem1 | fontEditToolStripMenuItem1\_Click |
| **23** | fontColor  ToolStrip  MenuItem1 | fontColorToolStripMenuItem1\_Click |
| **24** | colorToolStrip  MenuItem1 |  |
| **25** | formColor  ToolStrip  MenuItem1 | formColorToolStripMenuItem1\_Click |
| **26** | controlColor  ToolStrip  MenuItem1 | controlColorToolStripMenuItem1\_Click |
| **27** | soundToolStrip  MenuItem1 |  |
| **28** | playToolStrip  MenuItem1 | playToolStripMenuItem1\_Click |
| **29** | stopToolStrip  MenuItem1 | stopToolStripMenuItem1\_Click |
| **30** | helpToolStrip  MenuItem2 | helpToolStripMenuItem2\_Click |
| **31** | logOutTool  StripMenuItem1 | logOutToolStripMenuItem1\_Click |

## Yêu cầu 1: Nhấn vào hình ảnh picAvatar thì mở thư mục chọn hình từ máy, sau đó thay hình vừa chọn vào picAvatar

**A computer screen shot of a computer

Description automatically generated**

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

{

OpenFileDialog openFile = new OpenFileDialog();

//set tiêu đề

openFile.Title = "Mở ảnh đại diện";

//check điều kiện lọc file

openFile.Filter = "File anh|\*.jpg;|All file|\*.\*";

//check xem người dùng đã chọn file hay chưa

if (openFile.ShowDialog() == DialogResult.OK)

{

picAvatar.Image = Image.FromFile(openFile.FileName);

}

}

## Yêu cầu 2: Nhấn vào button btnTaiChinh và btnTaiKhoanNganHang thì mở form tương ứng (dạng MDI)

**A screenshot of a computer

Description automatically generated**

//mở form các tài khoản ngân hàng

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

{

FormTaiKhoan f = new FormTaiKhoan();

f.TopLevel = false;

this.Controls.Add(f); // Thêm form con vào các điều khiển của form cha

f.Dock = DockStyle.Fill; // Đặt đầy đủ kích thước của form con

f.BringToFront(); // Đưa form con lên phía trước

f.Show(); // Hiển thị form con

}

//Mở form tài chính

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

{

FormTaiChinh f = new FormTaiChinh();

f.TopLevel = false;

this.Controls.Add(f); // Thêm form con vào các điều khiển của form cha

f.Dock = DockStyle.Fill; // Đặt đầy đủ kích thước của form con

f.BringToFront(); // Đưa form con lên phía trước

f.Show(); // Hiển thị form con

}

## Yêu cầu 3: Sau 3 giây thay đổi hình ảnh của picQuangCao\_36\_Nguyen một lần

A screenshot of a computer

Description automatically generated

private int currentImageIndex\_36\_Nguyen = 0;

private List<Image> images = new List<Image>(); // Danh sách các ảnh cần thay đổi

// Thêm các ảnh vào danh sách

images.Add(Properties.Resources.cake);

images.Add(Properties.Resources.gioiThieuCake);

images.Add(Properties.Resources.laisuatall);

// Bắt đầu đếm thời gian sau mỗi 3 giây

timerImg\_36\_Nguyen.Start();

private void timerImg\_36\_Nguyen\_Tick(object sender, EventArgs e)

{

// Kiểm tra nếu đã đến ảnh cuối cùng trong danh sách, quay lại ảnh đầu tiên

if (currentImageIndex\_36\_Nguyen == images.Count - 1)

{

currentImageIndex\_36\_Nguyen = 0;

}

else

{

currentImageIndex\_36\_Nguyen++;

}

// Hiển thị ảnh tương ứng trong Panel

picQuangCao\_36\_Nguyen.Image = images[currentImageIndex\_36\_Nguyen];

}

## Yêu cầu 4: Thực hiện hiển thị thời gian ngày tháng năm giờ hiện tại cho label lbDateTime\_36\_Nguyen

// ngày giờ

timerDateTime\_36\_Nguyen.Start();

private void timerDateTime\_36\_Nguyen\_Tick(object sender, EventArgs e)

{

// Cập nhật thời gian lên Label

lbDateTime\_36\_Nguyen.Text = DateTime.Now.ToString("dd/MM/yyyy HH:mm:ss");

}

## Yêu cầu 5: Mở form lên thêm sẵn nội dung số thứ tự cho các dòng của gridBangThoiKhoaBieu

// thêm row cho bảng thời khoá biểu khi form dc load

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

{

for (int i = 1; i <= 31; i++)

{

gridBangThoiKhoaBieu.Rows.Add(i + ". ", i + ". ", i + ". ", i + ". ", i + ". ", i + ". ", i + ". ");

}

}

## Yêu cầu 6: Thực hiện các chức năng trên thanh menustrip, *gồm:*

## Làm mới, lưu, mở bảng gridBangThoiKhoaBieu

## Đóng ứng dụng (có hỏi lại bạn có chắc)

## cài đặt font chữ, màu chữ, màu control, màu nền, mở tắt nhạc

## Mở thông báo trợ giúp

## Thoát tài khoản (có hỏi lại bạn có chắc)

**A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated**

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

//Làm mới bảng

private void newToolStripMenuItem2\_Click(object sender, EventArgs e)

{

gridBangThoiKhoaBieu.Rows.Clear();

for (int i = 1; i <= 29; i++)

{

gridBangThoiKhoaBieu.Rows.Add(i + ". ", i + ". ", i + ". ", i + ". ", i + ". ", i + ". ", i + ". ");

}

}

//Mở bảng grid tkb

private void openToolStripMenuItem2\_Click(object sender, EventArgs e)

{

OpenFileDialog openFileDialog = new OpenFileDialog();

openFileDialog.Filter = "Text files (\*.txt)|\*.txt|All files (\*.\*)|\*.\*";

openFileDialog.FilterIndex = 1; // mặc định chọn đuôi .txt

if (openFileDialog.ShowDialog() == DialogResult.OK)

{

try

{

gridBangThoiKhoaBieu.Rows.Clear();

string filePath = openFileDialog.FileName;

// Đọc dữ liệu từ tệp và hiển thị trên DataGridView

string[] lines = File.ReadAllLines(filePath);

// Bỏ qua dòng đầu tiên

lines = lines.Skip(1).ToArray();

foreach (string line in lines)

{

string[] data = line.Split('\t'); // Giả sử dữ liệu được phân tách bằng tab (\t)

gridBangThoiKhoaBieu.Rows.Add(data);

}

MessageBox.Show("Đã mở tệp thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

catch (Exception ex)

{

MessageBox.Show("Lỗi khi mở tệp: " + ex.Message, "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

}

//hàm xử lí Lưu bảng thành file

private void SaveToTextFile(DataGridView dataGridView, string filePath)

{

try

{

using (StreamWriter writer = new StreamWriter(filePath))

{

// Ghi tiêu đề cột

for (int i = 0; i < dataGridView.Columns.Count; i++)

{

writer.Write(dataGridView.Columns[i].HeaderText);

if (i < dataGridView.Columns.Count - 1)

writer.Write("\t");

}

writer.WriteLine();

// Ghi dữ liệu từ các dòng

foreach (DataGridViewRow row in dataGridView.Rows)

{

for (int i = 0; i < dataGridView.Columns.Count; i++)

{

writer.Write(row.Cells[i].Value);

if (i < dataGridView.Columns.Count - 1)

writer.Write("\t");

}

writer.WriteLine();

}

}

MessageBox.Show("Lưu thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

catch (Exception ex)

{

MessageBox.Show("Lỗi: " + ex.Message, "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

//Lưu bảng thời khoá biểu thành txt

private void saveToolStripMenuItem2\_Click(object sender, EventArgs e)

{

SaveFileDialog saveFileDialog = new SaveFileDialog();

saveFileDialog.Filter = "Text files (\*.txt)|\*.txt|File pdf|\*.pdf |All files (\*.\*)|\*.\*";

saveFileDialog.FilterIndex = 1; //Xác định chỉ số của bộ lọc mặc định được chọn trong danh sách các bộ lọc

saveFileDialog.RestoreDirectory = true; //Khôi phục thư mục hiện tại khi hộp thoại đóng lại

if (saveFileDialog.ShowDialog() == DialogResult.OK)

{

SaveToTextFile(gridBangThoiKhoaBieu, saveFileDialog.FileName);

}

}

//Nhấn Exit để đóng app

private void exitToolStripMenuItem2\_Click(object sender, EventArgs e)

{

DialogResult traLoi;

traLoi = MessageBox.Show("Bạn có chắc muốn thoát ứng dụng không?", "Trả lời", MessageBoxButtons.OKCancel, MessageBoxIcon.Question);

if (traLoi == DialogResult.OK)

{

Application.Exit();

}

}

//FontDialog menu

//Chỉnh font chữ

private void fontEditToolStripMenuItem1\_Click(object sender, EventArgs e)

{

FontDialog dlg = new FontDialog();

dlg.ShowEffects = true;

//dlg.ShowColor = true;

dlg.ShowApply = true;

if (dlg.ShowDialog() == DialogResult.OK)

{

// Lặp qua tất cả các control trong Form

foreach (Control control in this.Controls)

{

// Thay đổi font cho mỗi control là Label, Button, TextBox….

control.Font = dlg.Font;

}

}

}

//ColorDialog menu

//Màu chữ

private void fontColorToolStripMenuItem1\_Click(object sender, EventArgs e)

{

ColorDialog dlg = new ColorDialog();

dlg.FullOpen = true; // Cho phép hiển thị tất cả các màu sắc

if (dlg.ShowDialog() == DialogResult.OK)

{

foreach (Control control in this.Controls)

{

control.ForeColor = dlg.Color;

}

}

}

//Màu form

private void formColorToolStripMenuItem1\_Click(object sender, EventArgs e)

{

ColorDialog dlg = new ColorDialog();

dlg.FullOpen = true; // Cho phép hiển thị tất cả các màu sắc

if (dlg.ShowDialog() == DialogResult.OK)

{

// Gán màu được chọn từ ColorDialog cho màu nền của Form

this.BackColor = dlg.Color;

}

}

//Màu control

private void controlColorToolStripMenuItem1\_Click(object sender, EventArgs e)

{

ColorDialog dlg = new ColorDialog();

dlg.FullOpen = true;

if (dlg.ShowDialog() == DialogResult.OK)

{

foreach (Control control in this.Controls)

{

control.BackColor = dlg.Color;

}

}

}

//Phát nhạc

private SoundPlayer music; // thuộc tính để lưu nhạc

private void playToolStripMenuItem1\_Click(object sender, EventArgs e)

{

try

{

// Kiểm tra nếu âm thanh đã được khởi tạo trước đó

if (music != null)

{

// Dừng phát âm thanh nếu đang phát

music.Stop();

// Giải phóng tài nguyên

music.Dispose();

}

// Khởi tạo SoundPlayer với đường dẫn tới tệp âm thanh

music = new SoundPlayer(@"C:\Users\NGUYEN\Time\_Schedule\_and\_Expense\_Management\Quản Lý Thời Gian Biểu và Chi Tiêu\resources\autumn-july.wav");

// Phát âm thanh

music.Play();

}

catch (Exception ex)

{

// Xử lý ngoại lệ (ví dụ: hiển thị thông báo cho người dùng)

MessageBox.Show("Lỗi khi phát âm thanh: " + ex.Message, "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

//Dừng nhạc

private void stopToolStripMenuItem1\_Click(object sender, EventArgs e)

{

try

{

// Kiểm tra nếu âm thanh đã được khởi tạo trước đó

if (music != null)

{

// Dừng phát âm thanh nếu đang phát

music.Stop();

// Giải phóng tài nguyên

music.Dispose();

}

// Khởi tạo SoundPlayer với đường dẫn tới tệp âm thanh

music = new SoundPlayer(@"C:\Users\NGUYEN\Time\_Schedule\_and\_Expense\_Management\Quản Lý Thời Gian Biểu và Chi Tiêu\resources\autumn-july.wav");

// Dừng âm thanh

music.Stop();

}

catch (Exception ex)

{

// Xử lý ngoại lệ (ví dụ: hiển thị thông báo cho người dùng)

MessageBox.Show("Lỗi khi phát âm thanh: " + ex.Message, "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

//âm thanh được dừng khi form đóng

private void MainForm\_FormClosing(object sender, FormClosingEventArgs e)

{

music.Stop();

}

//help content

private void helpToolStripMenuItem2\_Click(object sender, EventArgs e)

{

MessageBox.Show("Vui lòng liên hệ qua HotLine: 0332636829\n" +

"hoặc qua email sau: nguyen.hochi2004@gmail.com", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Information);

}

//nhấn vào Log out để ra màn hình đăng nhập

private void logOutToolStripMenuItem1\_Click(object sender, EventArgs e)

{

DialogResult traLoi;

traLoi = MessageBox.Show("Bạn có chắc muốn về màn hình Login không?", "Trả lời", MessageBoxButtons.OKCancel, MessageBoxIcon.Question);

if (traLoi == DialogResult.OK)

{

this.Hide();

FormLogin login = new FormLogin();

login.ShowDialog();

this.Close();

}

}

1. **Form quản lí tài chính chi tiêu**A screenshot of a computer

   Description automatically generated

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Control** | **Loại** | **Thuộc tính** | **Hàm xử lí** |
| **1** | FormTaiKhoan | Form | BackColor: LightYellow  Icon: nhập từ máy  Text: Quản lí tài chính  Startposition: CenterParent | FormTaiChinh\_Load |
| **2** | lbTieuDeTaiChinh  \_36\_Nguyen | Label | BackColorL Yellow  BorderStyle: Fixsingle  Text: Tài chính của bạn  TextAlign: MiddleCenter |  |
| **3** | lbNguonTien1  \_36\_Nguyen | Label | Anchor: top  BackColor: LightYellow  AutoSize: true  Text: Nguồn tiền 1 |  |
| **3** | lbNguonTien2  \_36\_Nguyen | Label | Anchor: top  BackColor: LightYellow  AutoSize: true  Text: Nguồn tiền 2 |  |
| **4** | lbNguonTien3  \_36\_Nguyen | Label | Anchor: top  BackColor: LightYellow  AutoSize: true  Text: Nguồn tiền 3 |  |
| **5** | lbNguonTien5  \_36\_Nguyen | Label | Anchor: top  BackColor: LightYellow  AutoSize: true  Text: Nguồn tiền 4 |  |
| **6** | lbTong  \_36\_Nguyen | Label | Anchor: top  BackColor: LightYellow  AutoSize: true  Text: Tổng tiền hiện có |  |
| **7** | lbMucDich  \_36\_Nguyen | Label | Anchor: top, bottom  BackColor: LightYellow  AutoSize: true  Text: Mục đích chi tiêu |  |
| **8** | lbSoTien  \_36\_Nguyen | Label | Anchor: top, bottom  BackColor: LightYellow  AutoSize: true  Text: Số tiền |  |
| **9** | lbNgayThang  \_36\_Nguyen | Label | Anchor: top, bottom  BackColor: LightYellow  AutoSize: true  Text: Thời gian |  |
| **10** | lbChiTiet  \_36\_Nguyen | Label | Anchor: top, bottom  BackColor: LightYellow  AutoSize: true  Text: Chi tiết |  |
| **11** | cbNguonTien1 | Combobox | Archor: top  backColor: Window  DropDownStyle: DropDownList |  |
| **12** | cbNguonTien2 | Combobox | Archor: top  backColor: Window  DropDownStyle: DropDownList |  |
| **13** | cbNguonTien3 | Combobox | Archor: top  backColor: Window  DropDownStyle: DropDownList |  |
| **14** | cbNguonTien4 | Combobox | Archor: top  backColor: Window  DropDownStyle: DropDownList |  |
| **15** | txtNguonTien1 | TextBox | Archor: top  backColor: Window  BorderStyle: Fix3D  Text: 0  TextAlign: Center | txtNguonTien1\_KeyPress  txtNguonTien1\_TextChanged |
| **16** | txtNguonTien2 | TextBox | Archor: top  backColor: Window  BorderStyle: Fix3D  Text: 0  TextAlign: Center | txtNguonTien2\_KeyPress  txtNguonTien2\_TextChanged |
| **17** | txtNguonTien3 | TextBox | Archor: top  backColor: Window  BorderStyle: Fix3D  Text: 0  TextAlign: Center | txtNguonTien3\_KeyPress  txtNguonTien3\_TextChanged |
| **18** | txtNguonTien4 | TextBox | Archor: top  backColor: Window  BorderStyle: Fix3D  Text: 0  TextAlign: Center | txtNguonTien4\_KeyPress  txtNguonTien4\_TextChanged |
| **19** | txtTong | TextBox | Archor: top  backColor: Control  ReadOnly: true  BorderStyle: Fix3D  Text: 0  TextAlign: Center |  |
| **20** | cbMucDich  \_36\_Nguyen | Combobox | Archor: top, Bottom  backColor: Window  DropDownStyle: DropDownList |  |
| **21** | txtSoTien  \_36\_Nguyen | TextBox | Archor: top, bottom  backColor: Window  BorderStyle: Fix3D  TextAlign: Center | txtSoTien\_36\_Nguyen\_KeyPress |
| **22** | dateTimeChiTieu  \_36\_Nguyen | DateTimePicker | Archor: top, bottom |  |
| **23** | richTextBoxChiTiet  \_36\_Nguyen | RichTextBox | Archor: top, bottom  BackColor: Window  BorderStyle: Fixsingle |  |
| **24** | btnXuatDanhSach  \_36\_Nguyen | Button | Archor: bottom  BackColor: LightBlue  Text: Xuất File  TextAlign: MiddleCenter | btnXuatDanhSach\_36\_Nguyen\_Click |
| **25** | btnXoa  \_36\_Nguyen | Button | Archor: bottom  BackColor: GreenYellow  Text: Xoá  TextAlign: MiddleCenter | btnXoa\_36\_Nguyen\_Click |
| **26** | btnThem  \_36\_Nguyen | Button | Archor: bottom  BackColor: PaleGreen  Text: Thêm  TextAlign: MiddleCenter | btnThem\_36\_Nguyen\_Click |
| **27** | listBoxChiTieu  \_36\_Nguyen | ListBox | BackColor: PaleGreen  Archor: top, bottom |  |

## Yêu cầu 1: nhập số tiền vào các TextBox: txtNguonTien1, txtNguonTien2, txtNguonTien3, txtNguonTien4, txtSoTien\_36\_Nguyen. Nội dung phải là số không được nhập chữ.

private void txtNguonTien1\_KeyPress(object sender, KeyPressEventArgs e)

{

// Kiểm tra nếu ký tự nhập vào không phải là số hoặc không phải là ký tự điều khiển (như phím Backspace)

if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))

{

// Không cho phép ký tự được nhập vào TextBox

e.Handled = true;

}

}

**Các textbox còn lại tương tự**

## Yêu cầu 2: Đặt sẵn nội dung mặc định cho các comboBox: cbNguonTien1, cbNguonTien2, cbNguonTien3, cbNguonTien4, cbMucDich\_36\_Nguyen

**A screenshot of a computer

Description automatically generated**

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

{

cbNguonTien1.SelectedIndex = 0;

cbNguonTien2.SelectedIndex = 1;

cbNguonTien3.SelectedIndex = 2;

cbNguonTien4.SelectedIndex = 3;

cbMucDich\_36\_Nguyen.SelectedIndex = 0;

}

## Yêu cầu 3: Khi nhập số tiền ở các textbox: txtNguonTien1, txtNguonTien2, txtNguonTien3, txtNguonTien4 thì số tiền ở txtTong tự động cập nhật (cộng 4 nguồn tiền lại)

**A screenshot of a computer

Description automatically generated**

private void tinhTong()

{

decimal tong = 0;

// Kiểm tra và cộng giá trị của từng ô TextBox vào tổng

if (decimal.TryParse(txtNguonTien1.Text, out decimal source1))

{

tong += source1;

}

if (decimal.TryParse(txtNguonTien2.Text, out decimal source2))

{

tong += source2;

}

if (decimal.TryParse(txtNguonTien3.Text, out decimal source3))

{

tong += source3;

}

if (decimal.TryParse(txtNguonTien4.Text, out decimal source4))

{

tong += source4;

}

// Hiển thị tổng trong TextBox tổng

txtTong.Text = tong.ToString();

}

private void txtNguonTien1\_TextChanged(object sender, EventArgs e)

{

tinhTong();

}

**Các textbox còn lại tương tự**

## Yêu cầu 4: Nhấn Button btnXuatDanhSach\_36\_Nguyen thì xuất tất cả nội dung của listBoxChiTieu\_36\_Nguyen thành file

A screenshot of a computer

Description automatically generated

private void xuatGhiFile(string filePath, ListBox listBox)

{

try

{

// Mở tệp văn bản để ghi

using (StreamWriter writer = new StreamWriter(filePath))

{

// Lặp qua từng mục trong ListBox và ghi vào tệp

foreach (var item in listBox.Items)

{

writer.WriteLine(item.ToString());

}

}

MessageBox.Show("Xuất ListBox thành công vào tệp " + filePath, "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

catch (Exception ex)

{

MessageBox.Show("Lỗi: " + ex.Message, "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

private void btnXuatDanhSach\_36\_Nguyen\_Click(object sender, EventArgs e)

{

SaveFileDialog saveFileDialog = new SaveFileDialog();

saveFileDialog.Filter = "Text files (\*.txt)|\*.txt|All files (\*.\*)|\*.\*";

if (saveFileDialog.ShowDialog() == DialogResult.OK)

{

string filePath = saveFileDialog.FileName;

// Gọi phương thức xuất ListBox

xuatGhiFile(filePath, listBoxChiTieu\_36\_Nguyen);

}

}

## Yêu cầu 5: Nhấn vào Button btnThem\_36\_Nguyen thì thêm nội dung nguồn tiền, tổng tiền, số tiền chi tiêu, mục đích mô tả và tiền còn lại. Cập nhật số tiền ở TextBox txtTong

**A screenshot of a computer

Description automatically generated**

private void btnThem\_36\_Nguyen\_Click(object sender, EventArgs e)

{

//kiểm tra nguồn tiền có nhập ko

if (string.IsNullOrEmpty(txtNguonTien1.Text) ||

string.IsNullOrEmpty(txtNguonTien2.Text) ||

string.IsNullOrEmpty(txtNguonTien3.Text) ||

string.IsNullOrEmpty(txtNguonTien4.Text) ||

string.IsNullOrEmpty(txtSoTien\_36\_Nguyen.Text))

{

MessageBox.Show("Vui lòng điền đầy đủ thông tin chi tiêu.", "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

return; // Dừng lại nếu có ô TextBox nào đó trống

}

// Lấy thông tin từ các điều khiển nhập liệu

decimal nguonTien1\_36\_nguyen = decimal.Parse(txtNguonTien1.Text);

decimal nguonTien2\_36\_nguyen = decimal.Parse(txtNguonTien2.Text);

decimal nguonTien3\_36\_nguyen = decimal.Parse(txtNguonTien3.Text);

decimal nguonTien4\_36\_nguyen = decimal.Parse(txtNguonTien4.Text);

decimal tongTien\_36\_Nguyen = decimal.Parse(txtTong.Text);

string mucDich = cbMucDich\_36\_Nguyen.Text;

decimal soTien = decimal.Parse(txtSoTien\_36\_Nguyen.Text); // Chuyển đổi về kiểu decimal (hoặc int)

DateTime ngayThucHien = dateTimeChiTieu\_36\_Nguyen.Value;

string moTa = richTextBoxChiTiet\_36\_Nguyen.Text;

//Tính tiền còn lại

txtTong.Text = String.Format("{0}", tongTien\_36\_Nguyen - soTien);

decimal tongTienConLai\_36\_Nguyen = tongTien\_36\_Nguyen - soTien;

// Tạo một chuỗi để hiển thị trong ListBox, .ToString("C") là để thêm đ vào sau tiền

string item = $"Tổng tiền: {tongTien\_36\_Nguyen.ToString("C")} - " +

$"Nguồn tiền: {nguonTien1\_36\_nguyen.ToString("C")}, {nguonTien2\_36\_nguyen.ToString("C")}, {nguonTien3\_36\_nguyen.ToString("C")}, {nguonTien4\_36\_nguyen.ToString("C")} - " +

$"{mucDich} - Số tiền chi: {soTien.ToString("C")} - {ngayThucHien.ToShortDateString()} - {moTa} - " +

$"Còn lại: {tongTienConLai\_36\_Nguyen.ToString("C")}";

// Thêm mục vào ListBox

listBoxChiTieu\_36\_Nguyen.Items.Add(item);

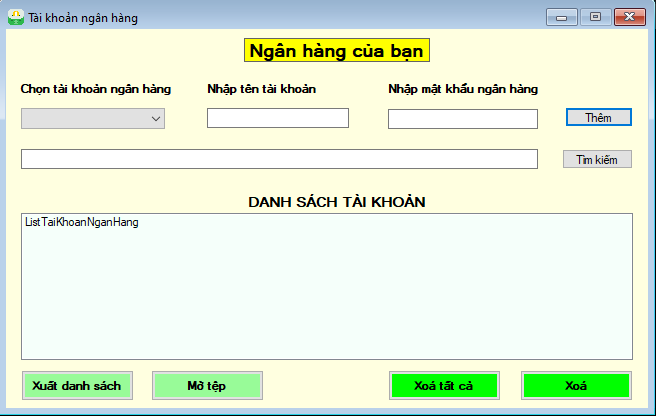
// Xóa nội dung của các TextBox sau khi thêm vào ListBox

txtSoTien\_36\_Nguyen.Clear();

richTextBoxChiTiet\_36\_Nguyen.Clear();

}

1. **Form quản lí tài khoản ngân hàng**

****

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên control** | **Loại** | **Thuộc tính** | **Hàm xử lí** |
| 1 | FormTaiKhoan | Form | AcceptButton: btnThemNganHang  BackColor: LightYellow  Icon: nhập từ máy  Text: Tài khoản ngân hàng  StartPosition: CenterParent | FormTaiKhoan\_Load |
| 2 | lbQuanLy  TaiKhoanBank | Label | BackColor: LightYellow  AutoSize: false  Bold: true  BorderStyle: fixSingle  Text: Ngân hàng của bạn  Archor: Top, Bottom |  |
| 3 | lbChonNganHang | Label | Archor: Top, Bottom  AutoSize: true  Text: Chọn tài khoản ngân hàng  BackColor: LightYellow  Bold: true |  |
| 4 | lbTen  TaiKhoan  NganHang | Label | Archor: Top, Bottom  AutoSize: true  Text: Nhập tên tài khoản  BackColor: LightYellow  Bold: true |  |
| 5 | lbMatKhau  NganHang | Label | Archor: Top, Bottom  AutoSize: true  Text: Nhập mật khẩu ngân hàng  BackColor: LightYellow  Bold: true |  |
| 6 | cbChonTaiKhoan  NganHang | ComboBox | Archor: top, Bottom  backColor: Window  DropDownStyle: DropDownList |  |
| 7 | txtTenTaiKhoan  NganHang | TextBox | Archor: top, Bottom  backColor: Window  BorderStyle: Fix3D  TextAlign: Center |  |
| 8 | txtMatKhau  NganHang | TextBox | Archor: top, Bottom  backColor: Window  BorderStyle: Fix3D  TextAlign: Center |  |
| 9 | txtTimKiem | TextBox | Archor: top, Bottom  backColor: Window  BorderStyle: Fix3D  TextAlign: Center |  |
| 10 | btnThemNganHang | Button | Archor: top  BackColor: LightYellow Text: Xuất File  TextAlign: MiddleCenter | btnThemNganHang\_Click |
| 11 | btnTimKiem | Button | Archor: top  BackColor: LightYellow Text: Xuất File  TextAlign: MiddleCenter | btnTimKiem\_Click |
| 12 | lbDsBank  \_36\_Nguyen | Label | Archor: Top, Bottom  AutoSize: false  Text: Danh sách tài khoản  BackColor: LightYellow  Bold: true |  |
| 13 | ListTaiKhoanNganHang | ListBox | BackColor: MintCream  Archor: Top, Bottom |  |
| 14 | btnXuatFile | Button | Archor: bottom  BackColor: PaleGreen  Text: Xuất danh sách  TextAlign: MiddleCenter | btnXuatFile\_Click |
| 15 | btnMoFile | Button | Archor: bottom  BackColor: PaleGreen  Text: Mở tệp  TextAlign: MiddleCenter | btnMoFile\_Click |
| 16 | btnXoaAll | Button | Archor: bottom  BackColor: Lime  Text: Xoá tất cả  TextAlign: MiddleCenter | btnXoaAll\_Click |
| 17 | btnXoa | Button | Archor: bottom  BackColor: Lime  Text: Xoá  TextAlign: MiddleCenter | btnXoa\_Click |

## Yêu cầu 1: Khi mở form lên mặc định giá trị cho comboBox cbChonTaiKhoanNganHang và ListTaiKhoanNganHang

A screenshot of a computer

Description automatically generated

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

{

ListTaiKhoanNganHang.Items.Add("----------DANH SÁCH TÀI KHOẢN NGÂN HÀNG----------");

cbChonTaiKhoanNganHang.SelectedIndex = 0;

}

## Yêu cầu 2: Nhấn vào Button btnThemNganHang thì thêm nội dung từ txtTenTaiKhoanNganHang và txtMatKhauNganHang vào ListTaiKhoanNganHang. Nếu 1 trong 2 ô txt không có nội dung thì báo lỗi

**A screenshot of a computer

Description automatically generated**

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

{

String tenNganHang = cbChonTaiKhoanNganHang.SelectedItem.ToString();

String taiKhoan = txtTenTaiKhoanNganHang.Text;

String matKhau = txtMatKhauNganHang.Text;

// Kiểm tra xem các TextBox có trống hay ko

if (!string.IsNullOrEmpty(tenNganHang) && !string.IsNullOrEmpty(taiKhoan) && !string.IsNullOrEmpty(matKhau))

{

// Chuỗi chứa thông tin tài khoản ngân hàng

// $ giúp chèn các biến hoặc biểu thức trực tiếp vào chuỗi mà không cần phải sử dụng các toán tử nối chuỗi

string thongTinTaiKhoan = $"{tenNganHang} - Tài khoản: {taiKhoan}, Mật khẩu: {matKhau}";

// Thêm dữ liệu từ ComboBox và TextBox vào ListBox

ListTaiKhoanNganHang.Items.Add(thongTinTaiKhoan);

// Xóa nội dung của các TextBox sau khi thêm vào ListBox

txtTenTaiKhoanNganHang.Clear();

txtMatKhauNganHang.Clear();

}

else

{

MessageBox.Show("Vui lòng nhập dữ liệu trước khi thêm vào danh sách", "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

## Yêu cầu 3: Nhấn vào Button btnTimKiem thì tìm kiếm nội dung trong ListTaiKhoanNganHang. Nếu không tìm thấy báo lỗi, nếu tìm thấy thì đánh dấu

**A screenshot of a computer

Description automatically generated**

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

{

string searchText = txtTimKiem.Text.ToLower(); // Lấy dữ liệu cần tìm kiếm từ TextBox và chuyển thành chữ thường

// Duyệt qua từng mục trong ListBox

for (int i = 0; i < ListTaiKhoanNganHang.Items.Count; i++)

{

string itemText = ListTaiKhoanNganHang.Items[i].ToString().ToLower(); // Lấy dữ liệu từng mục trong ListBox và chuyển thành chữ thường

// Nếu dữ liệu trong mục khớp với dữ liệu cần tìm kiếm

if (itemText.Contains(searchText))

{

ListTaiKhoanNganHang.SetSelected(i, true); // Đặt màu sắc cho mục tương ứng

ListTaiKhoanNganHang.TopIndex = i; // Cuộn ListBox đến mục được chọn

ListTaiKhoanNganHang.Focus(); // Đưa focus vào ListBox để dễ dàng nhìn thấy mục được chọn

return; // Dừng vòng lặp khi tìm thấy kết quả

}

}

MessageBox.Show("Không tìm thấy dữ liệu.", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

## Yêu cầu 4: Nhấn vào button btnXuatFile xuất danh sách nội dung trong ListTaiKhoanNganHang thành file txt

**A screenshot of a computer

Description automatically generated**

private void xuatGhiFile(string filePath, ListBox listBox)

{

try

{

// Mở tệp văn bản để ghi

using (StreamWriter writer = new StreamWriter(filePath))

{

// Lặp qua từng mục trong ListBox và ghi vào tệp

foreach (var item in listBox.Items)

{

writer.WriteLine(item.ToString());

}

}

MessageBox.Show("Xuất ListBox thành công vào tệp " + filePath, "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

catch (Exception ex)

{

MessageBox.Show("Lỗi: " + ex.Message, "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

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

{

SaveFileDialog saveFileDialog = new SaveFileDialog();

saveFileDialog.Filter = "Text files (\*.txt)|\*.txt|All files (\*.\*)|\*.\*";

if (saveFileDialog.ShowDialog() == DialogResult.OK)

{

string filePath = saveFileDialog.FileName;

// Gọi phương thức xuất ListBox

xuatGhiFile(filePath, ListTaiKhoanNganHang);

}

}

## Yêu cầu 5: Nhấn vào Button btnMoFile mở file txt danh sách tài khoản có sẵn

private void xuLiMoFile(string filePath, ListBox listBox)

{

try

{

// Kiểm tra xem tệp tồn tại trước khi đọc

if (File.Exists(filePath))

{

// Mở tệp văn bản để đọc

using (StreamReader reader = new StreamReader(filePath))

{

// Đọc từng dòng của tệp và thêm vào ListBox

string line;

while ((line = reader.ReadLine()) != null)

{

listBox.Items.Add(line);

}

}

}

else

{

MessageBox.Show("Tệp không tồn tại.", "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

catch (Exception ex)

{

MessageBox.Show("Lỗi: " + ex.Message, "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

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

{

OpenFileDialog openFileDialog = new OpenFileDialog();

openFileDialog.Filter = "Text files (\*.txt)|\*.txt|All files (\*.\*)|\*.\*";

if (openFileDialog.ShowDialog() == DialogResult.OK)

{

string filePath = openFileDialog.FileName;

// Xóa các mục hiện có trong ListBox trước khi tải dữ liệu mới

ListTaiKhoanNganHang.Items.Clear();

// Gọi phương thức để tải dữ liệu từ tệp vào ListBox

xuLiMoFile(filePath, ListTaiKhoanNganHang);

}

}

## Yêu cầu 6: Nhấn vào button btnXoaAll Xoá tất cả nội dung trong ListTaiKhoanNganHang

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

{

DialogResult traLoiXoaAll;

traLoiXoaAll = MessageBox.Show("Bạn có chắc muốn xoá tất cả không?", "Trả lời", MessageBoxButtons.OKCancel, MessageBoxIcon.Question);

if (traLoiXoaAll == DialogResult.OK)

{

ListTaiKhoanNganHang.Items.Clear();

}

}

## Yêu cầu 7: Nhấn vào button btnXoa thì xoá một nội dung đang chọn trong ListTaiKhoanNganHang. Nếu chưa chọn thì báo lỗi

**A screenshot of a computer

Description automatically generated**

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

{

// Kiểm tra xem có mục nào được chọn không

if (ListTaiKhoanNganHang.SelectedIndex != -1)

{

// Lấy chỉ số của mục được chọn

int mucChon = ListTaiKhoanNganHang.SelectedIndex;

// Xóa mục được chọn từ ListBox

ListTaiKhoanNganHang.Items.RemoveAt(mucChon);

}

else

{

MessageBox.Show("Vui lòng chọn một mục để xóa", "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}