**HỌ TÊN:** Hồ Chí Nguyên

**STT:** 36

**MSSV:** 2251050048

**Mục lục**

[**1.** **Form đăng kí - đăng nhập** 4](#_Toc164901969)

[Yêu cầu 1: Nhấn picEye thì đổi ảnh mắt đóng thành mắt mở và ngược lại 7](#_Toc164901970)

[Yêu cầu 2: Nhấn lbQuenMK thì hiện thông báo trợ giúp 8](#_Toc164901971)

[Yêu cầu 3: 8](#_Toc164901972)

[Nhấn btnLogin để đăng nhập, nếu thành công thì hiện thông báo thành công. 8](#_Toc164901973)

[Nếu 1 trong 2 ô txtUser và txtPassWord chưa nhập hoặc nhập sai thì báo lỗi. 9](#_Toc164901974)

[Nội dung tài khoản mật khẩu được đọc từ file .txt. 9](#_Toc164901975)

[Yêu cầu 4: 11](#_Toc164901976)

[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 11](#_Toc164901977)

[Nếu 1 trong 2 ô txtUser và txtPassWord chưa nhập thì báo lỗi. 11](#_Toc164901978)

[Đọc file .txt, nếu txtUser và txtPassWord đã tồn tại thì báo lỗi 11](#_Toc164901979)

[Yêu cầu 5: Css màu, hiệu ứng 14](#_Toc164901980)

[**2.** **Form Xác thực** 15](#_Toc164901981)

[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. 16](#_Toc164901982)

[Yêu cầu 1: Nhấn vào hình ảnh picAvatar thì mở thư mục chọn hình từ máy, sau đó thay hình vừa chọn vào picAvatar 23](#_Toc164901983)

[Yêu cầu 2: Nhấn vào button btnTaiChinh và btnTaiKhoanNganHang thì mở form tương ứng (dạng MDI) 25](#_Toc164901984)

[Yêu cầu 3: Sau 3 giây thay đổi hình ảnh của picQuangCao\_36\_Nguyen một lần 27](#_Toc164901985)

[Yêu cầu 4: Thực hiện hiển thị thời gian ngày tháng năm giờ hiện tại cho label lbDateTime\_36\_Nguyen 29](#_Toc164901986)

[Yêu cầu 5: Mở form lên thêm sẵn nội dung số thứ tự cho các dòng của gridBangThoiKhoaBieu 30](#_Toc164901987)

[Yêu cầu 6: Thực hiện các chức năng trên thanh menustrip, *gồm:* 30](#_Toc164901988)

[Làm mới, lưu, mở bảng gridBangThoiKhoaBieu 30](#_Toc164901989)

[Đóng ứng dụng (có hỏi lại bạn có chắc) 30](#_Toc164901990)

[cài đặt font chữ, màu chữ, màu control, màu nền, mở tắt nhạc 30](#_Toc164901991)

[Mở thông báo trợ giúp 30](#_Toc164901992)

[Thoát tài khoản (có hỏi lại bạn có chắc) 30](#_Toc164901993)

[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. 45](#_Toc164901994)

[Yêu cầu 2: Đặt sẵn nội dung mặc định cho các comboBox: cbNguonTien1, cbNguonTien2, cbNguonTien3, cbNguonTien4, cbMucDich\_36\_Nguyen 46](#_Toc164901995)

[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) 47](#_Toc164901996)

[Yêu cầu 4: Nhấn Button btnXuatDanhSach\_36\_Nguyen thì xuất tất cả nội dung của listBoxChiTieu\_36\_Nguyen thành file 48](#_Toc164901997)

[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 50](#_Toc164901998)

[Yêu cầu 6: Nhấn vào Button btnXoa\_36\_Nguyen thì xoá nội dung đang được chọn trong listBox listBoxChiTieu\_36\_Nguyen, nếu chưa chọn báo lỗi 53](#_Toc164901999)

[**5.** **Form quản lí tài khoản ngân hàng** 53](#_Toc164902000)

[Yêu cầu 1: Khi mở form lên mặc định giá trị cho comboBox cbChonTaiKhoanNganHang 59](#_Toc164902001)

[Yêu cầu 2: Nhấn vào Button btnThemNganHang thì thêm nội dung từ cbChonTaiKhoanNganHang\_36\_Nguyen, txtTenTaiKhoanNganHang\_36\_Nguyen và txtMatKhauNganHang\_36\_Nguyen vào ListView lvDanhSachNganHang\_36\_Nguyen. Nếu 1 trong 3 ô không có nội dung thì báo lỗi 61](#_Toc164902002)

[Yêu cầu 3: Nhấn vào Button btnTimKiem thì tìm kiếm nội dung trong ListView lvDanhSachNganHang\_36\_Nguyen. Nếu không tìm thấy báo lỗi, nếu tìm thấy thì đánh dấu (Có thể tìm nhiều nội dung giống nhau) 62](#_Toc164902003)

[Yêu cầu 4: Nhấn vào button btnXuatFile\_36\_Nguyen thì xuất danh sách nội dung trong ListTaiKhoanNganHang thành file .txt 64](#_Toc164902004)

[Yêu cầu 5: Nhấn vào Button btnMoFile\_36\_Nguyen thì mở file txt danh sách tài khoản có sẵn 67](#_Toc164902005)

[Yêu cầu 6: Nhấn vào button btnXoaAll\_36\_Nguyen thì Xoá tất cả nội dung lvDanhSachNganHang\_36\_Nguyen (có hỏi lại chắc chắn không) 68](#_Toc164902006)

[Yêu cầu 7: Nhấn vào button btnXoa\_36\_Nguyen thì xoá một nội dung đang chọn trong lvDanhSachNganHang\_36\_Nguyen. Nếu chưa chọn nội dung thì báo lỗi 70](#_Toc164902007)

[Yêu cầu 8: Nhấn vào Button btnSua\_36\_Nguyen thì lấy nội dung từ 3 ô cbChonTaiKhoanNganHang\_36\_Nguyen, txtTenTaiKhoanNganHang\_36\_Nguyen và txtMatKhauNganHang\_36\_Nguyen để cập nhật lại 1 nội dung được chọn trong lvDanhSachNganHang\_36\_Nguyen 71](#_Toc164902008)

[\*Lưu ý phải viết phương thức xử lí khi chọn 1 nội dung trong lvDanhSachNganHang\_36\_Nguyen thì lấy nội dung được chọn cập nhật lên 3 ô cbChonTaiKhoanNganHang\_36\_Nguyen, txtTenTaiKhoanNganHang\_36\_Nguyen và txtMatKhauNganHang\_36\_Nguyen 71](#_Toc164902009)

**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\_36\_Nguyen | Form | Autosize: true  AcceptButton: btnLogin  Icon: nhập từ máy  BackgroundImageL nhập từ máy  StartPosition: Centergreen  BorderStyle: Fix3D  Text: Login |  |
| **2** | picbanner\_36\_Nguyen | PictureBox | BackColor: Limegreen  SizeMode: Normal  BoderStyle: FixSingle |  |
| **3** | lbCauNoi\_36\_Nguyen | Label | Text: Cần cù thì bù siêng năng, có tiết kiệm thì mới có ăn  BackColor: PeachPuff  AutoSize: true |  |
| **4** | panelLogin\_36\_Nguyen | Panel | BackColor: AntiqueWhite  AutoSize: true  BoderStyle: FixSingle |  |
| **5** | lbLogin\_36\_Nguyen | Label | BackColor: Lime  AutoSize: true  BoderStyle: FixSingle  Text: LOGIN |  |
| **6** | lbUser\_36\_Nguyen | Label | Text: Tên đăng nhập  AutoSize: true |  |
| **7** | txtUser\_36\_Nguyen | TextBox | BorderStyle: Fix3D  TextAlign: Center  AcceptsTab: true  AcceptReturn: true | txtUser\_36\_Nguyen\_MouseDown  txtUser\_36\_Nguyen\_MouseLeave |
| **8** | lbPassWord\_36\_Nguyen | Label | Text: Mật khẩu  AutoSize: true |  |
| **9** | txtPassWord\_36\_Nguyen | TextBox | BorderStyle: Fix3D  TextAlign: Center  AcceptsTab: true  AcceptReturn: true | txtPassWork\_36\_Nguyen\_MouseDown  txtPassWork\_MouseLeave |
| **10** | picEye\_36\_Nguyen | PictureBox | BackColor: AntiqueWhite  Cursor: hand  SizeMode: zoom | picEye\_36\_Nguyen\_Click |
| **12** | lbQuenMK\_36\_Nguyen | Label | Text: Quên mật khẩu?  Italic: true | lbQuenMK\_36\_Nguyen\_Click  lbQuenMK\_36\_Nguyen\_MouseHover  lbQuenMK\_MouseLeave |
| **13** | btnSignUp\_36\_Nguyen | Button | Text: Đăng kí  AutoSize: true  Bold: true  TextAlign: MiddleCenter  BackColor: : AntiqueWhite | btnSignUp\_36\_Nguyen\_Click |
| **14** | btnLogin\_36\_Nguyen | Button | Text: Đăng nhập  AutoSize: true  Bold: true  TextAlign: MiddleCenter  BackColor: : AntiqueWhite | btnLogin\_36\_Nguyen\_Click |

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

private bool isEyeOpen\_36\_Nguyen = false;

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

private void ChangeEyeImage\_36\_Nguyen()

{

if (isEyeOpen\_36\_Nguyen)

{

picEye\_36\_Nguyen.Image = Properties.Resources.EyeOpen;

}

else

{

picEye\_36\_Nguyen.Image = Properties.Resources.Eye;

}

}

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

{

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

if (txtPassWord\_36\_Nguyen.PasswordChar == '\*')

txtPassWord\_36\_Nguyen.PasswordChar = '\0'; // Hiển thị mật khẩu

else

txtPassWord\_36\_Nguyen.PasswordChar = '\*'; // Ẩn mật khẩu

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

isEyeOpen\_36\_Nguyen = !isEyeOpen\_36\_Nguyen;

ChangeEyeImage\_36\_Nguyen();

}

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

private void lbQuenMK\_36\_Nguyen\_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);

}

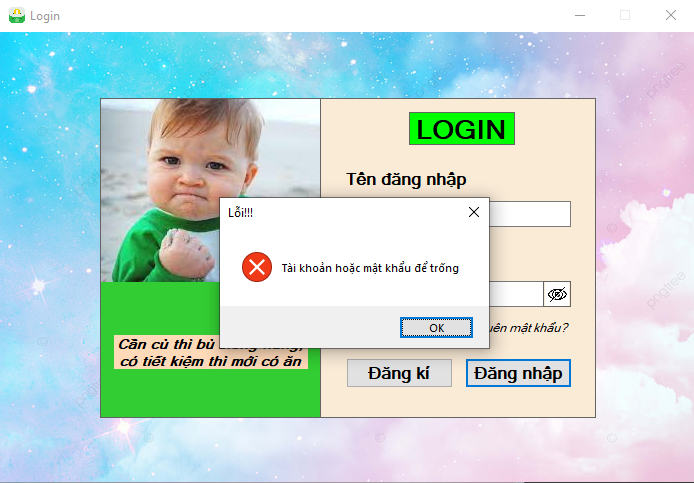
## Yêu cầu 3:

## 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

## Tài khoản mặc định là Nguyen, Mật khẩu là 36

A screenshot of a computer screen

Description automatically generated

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

private bool CheckCredentials\_36\_Nguyen(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\_36\_Nguyen = File.ReadAllLines(filePath);

foreach (string line in lines\_36\_Nguyen)

{

string[] parts\_36\_Nguyen = line.Split(',');

if (parts\_36\_Nguyen.Length == 2 && parts\_36\_Nguyen[0].Trim() == username && parts\_36\_Nguyen[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 btn

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

{

string username\_36\_Nguyen = txtUser\_36\_Nguyen.Text;

string password\_36\_Nguyen = txtPassWord\_36\_Nguyen.Text;

bool isAuthenticated = CheckCredentials\_36\_Nguyen(username\_36\_Nguyen, password\_36\_Nguyen);

if (isAuthenticated)

{

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

this.Hide();

FormMain\_36\_Nguyen frmMain = new FormMain\_36\_Nguyen();

frmMain.ShowDialog();

this.Close();

}

else if (txtUser\_36\_Nguyen.Text == "Nguyen" && txtPassWord\_36\_Nguyen.Text == "36")

{

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

this.Hide();

FormMain\_36\_Nguyen frmMain = new FormMain\_36\_Nguyen();

frmMain.ShowDialog();

this.Close();

}

else if (txtUser\_36\_Nguyen.Text == "" || txtPassWord\_36\_Nguyen.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\_36\_Nguyen.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\_36\_Nguyen\_Click(object sender, EventArgs e)

{

string username\_36\_Nguyen = txtUser\_36\_Nguyen.Text;

string password\_36\_Nguyen = txtPassWord\_36\_Nguyen.Text;

bool isAuthenticated\_36\_Nguyen = CheckCredentials\_36\_Nguyen(username\_36\_Nguyen, password\_36\_Nguyen);

if (isAuthenticated\_36\_Nguyen)

{

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

return;

}

else if (txtUser\_36\_Nguyen.Text == "" || txtPassWord\_36\_Nguyen.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\_36\_Nguyen = File.AppendText(filePath))

{

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

sw\_36\_Nguyen.WriteLine(/\*"\n" + \*/username\_36\_Nguyen + "," + password\_36\_Nguyen);

}

this.Hide();

IsRobot\_36\_Nguyen robot = new IsRobot\_36\_Nguyen();

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\_36\_Nguyen\_MouseHover(object sender, EventArgs e)

{

lbQuenMK\_36\_Nguyen.Font = new Font(lbQuenMK\_36\_Nguyen.Font, FontStyle.Underline);

}

private void lbQuenMK\_36\_Nguyen\_MouseLeave(object sender, EventArgs e)

{

lbQuenMK\_36\_Nguyen.Font = new Font(lbQuenMK\_36\_Nguyen.Font, FontStyle.Italic);

}

//Hover vào có màu ở ô Đăng nhập

private void txtUser\_36\_Nguyen\_MouseDown(object sender, MouseEventArgs e)

{

txtUser\_36\_Nguyen.BackColor = Color.LightGreen;

}

private void txtUser\_36\_Nguyen\_MouseLeave(object sender, EventArgs e)

{

txtUser\_36\_Nguyen.BackColor = Color.White;

}

//Hover vào có màu ở ô Mật khẩu

private void txtPassWork\_36\_Nguyen\_MouseDown(object sender, MouseEventArgs e)

{

txtPassWord\_36\_Nguyen.BackColor = Color.LightGreen;

}

private void txtPassWork\_36\_Nguyen\_MouseLeave(object sender, EventArgs e)

{

txtPassWord\_36\_Nguyen.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\_36\_Nguyen | Form | Autosize: true  Icon: nhập từ máy  BackgroundImageL nhập từ máy |  |
| **2** | picIsRobot\_36\_Nguyen | PictureBox | BorderStyle: FixedSingle  Cursor: Hand  SizeMode: Zoom | picIsRobot\_36\_Nguyen\_MouseDown  picIsRobot\_36\_Nguyen\_MouseMove  picIsRobot\_36\_Nguyen\_MouseUp |
| **3** | panIsRobot\_36\_Nguyen | Panel | BackColor: Pink  AutoSize: false |  |
| **4** | lbIsRobot\_36\_Nguyen | 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 point\_36\_Nguyen; //Biến lưu giữ vị trí chuột

private void picIsRobot\_36\_Nguyen\_MouseDown(object sender, MouseEventArgs e)

{

point\_36\_Nguyen = e.Location;

}

private void picIsRobot\_36\_Nguyen\_MouseMove(object sender, MouseEventArgs e)

{

if (e.Button == MouseButtons.Left)

{

int dx = e.X - point\_36\_Nguyen.X;

int dy = e.Y - point\_36\_Nguyen.Y;

picIsRobot\_36\_Nguyen.Left += dx;

picIsRobot\_36\_Nguyen.Top += dy;

}

}

private void picIsRobot\_36\_Nguyen\_MouseUp(object sender, MouseEventArgs e)

{

if(panIsRobot\_36\_Nguyen.Bounds.Contains(picIsRobot\_36\_Nguyen.Bounds))

{

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

this.Hide();

FormLogin\_36\_Nguyen login = new FormLogin\_36\_Nguyen();

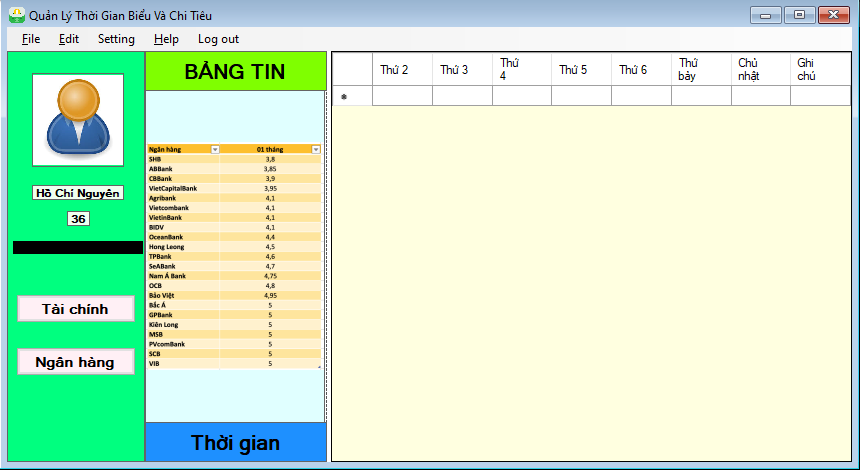
login.ShowDialog();

this.Close();

}

}

1. **Form Chính (Trang chủ)**

****

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Control** | **Loại** | **Thuộc tính** | **Hàm xử lí** |
| **1** | FormMain\_36\_Nguyen | Form | WindowState: Maximized  Text: Quản lí thời gian biểu và chi tiêu  AutoSize: true  Icon: nhập từ máy  StartPosition: Centergreen | FormMain\_36\_Nguyen\_Load |
| **2** | TaskBar\_36\_Nguyen | Panel | BackColor: SpringGreen  AutoSize: False  BorderStyle: FixSingle  Dock: Left |  |
| **3** | picAvatar\_36\_Nguyen | PictureBox | BorderStyle: FixSingle  Cursor: Hand  Image: nhập từ máy  SizeMode: Zoom  Archor: Top, Bottom, Left, Right | picAvatar\_36\_Nguyen\_Click |
| **4** | lbHoTen\_36\_Nguyen | Label | BackColor: Honeydew  BorderStyle: FixSingle  AutoSize: true  Text: Hồ Chí Nguyên |  |
| **5** | lbId\_36\_Nguyen | Label | BackColor: Honeydew  BorderStyle: FixSingle  AutoSize: true  Text: 36 |  |
| **6** | lbDuongKe\_36\_Nguyen | Label | BackColor: InactiveCaptionText |  |
| **7** | btnTaiChinh\_36\_Nguyen | Button | BackColor: LavenderBlush  Text: Tài chính  TextAlign: MiddleCenter  Cursor: Hand | btnTaiChinh\_36\_Nguyen\_Click |
| **8** | btnTaiKhoanNganHang\_36\_Nguyen | Button | BackColor: LavenderBlush  Text: Ngân hàng  TextAlign: MiddleCenter  Cursor: Hand | btnTaiKhoanNganHang\_36\_Nguyen\_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** | gridBangThoiKhoaBieu\_36\_Nguyen | DataGridView | BackColor: LightYellow  BorderStyle: FixSingle  GridColor: ControlDark  Dock: Fill |  |
| **13** | menuStrip1\_36\_Nguyen | MenuStrip | AutoSize: true  Dock: top |  |
| **14** | fileToolStripMenuItem1\_36\_Nguyen | MenuStripItem | Text: New  TextAlign: MiddleCenter  BackColor: Control |  |
| **15** | newToolStripMenuItem2\_36\_Nguyen | newToolStripMenuItem2\_36\_Nguyen\_Click |
| **16** | openToolStripMenuItem2\_36\_Nguyen | openToolStripMenuItem2\_36\_Nguyen\_Click |
| **17** | saveToolStripMenuItem2\_36\_Nguyen | saveToolStripMenuItem2\_36\_Nguyen\_Click |
| **18** | exitToolStripMenuItem2\_36\_Nguyen | exitToolStripMenuItem2\_36\_Nguyen\_Click |
| **19** | editToolStripMenuItem1\_36\_Nguyen |  |
| **20** | settinToolStripMenuItem\_36\_Nguyen |  |
| **21** | fontToolStripMenuItem\_36\_Nguyen |  |
| **22** | fontEditToolStripMenuItem1\_36\_Nguyen | fontEditToolStripMenuItem1\_36\_Nguyen\_Click |
| **23** | fontColorToolStripMenuItem1\_36\_Nguyen | fontColorToolStripMenuItem1\_36\_Nguyen\_Click |
| **24** | colorToolStripMenuItem1\_36\_Nguyen |  |
| **25** | formColorToolStripMenuItem1\_36\_Nguyen | formColorToolStripMenuItem1\_36\_Nguyen\_Click |
| **26** | controlColorToolStripMenuItem1\_36\_Nguyen | controlColorToolStripMenuItem1\_36\_Nguyen\_Click |
| **27** | soundToolStripMenuItem1\_36\_Nguyen |  |
| **28** | playToolStripMenuItem1\_36\_Nguyen | playToolStripMenuItem1\_36\_Nguyen\_Click |
| **29** | stopToolStripMenuItem1\_36\_Nguyen | stopToolStripMenuItem1\_36\_Nguyen\_Click |
| **30** | helpToolStripMenuItem2\_36\_Nguyen | helpToolStripMenuItem2\_36\_Nguyen\_Click |
| **31** | logOutToolStripMenuItem1\_36\_Nguyen | logOutToolStripMenuItem1\_36\_Nguyen\_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\_36\_Nguyen\_Click(object sender, EventArgs e)

{

OpenFileDialog openFile\_36\_Nguyen = new OpenFileDialog();

//set tiêu đề

openFile\_36\_Nguyen.Title = "Mở ảnh đại diện";

//check điều kiện lọc file

openFile\_36\_Nguyen.Filter = "File anh|\*.jpg;|All file|\*.\*";

//check xem người dùng đã chọn file hay chưa

if (openFile\_36\_Nguyen.ShowDialog() == DialogResult.OK)

{

picAvatar\_36\_Nguyen.Image = Image.FromFile(openFile\_36\_Nguyen.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\_36\_Nguyen\_Click(object sender, EventArgs e)

{

FormTaiKhoan\_36\_Nguyen f\_36\_Nguyen = new FormTaiKhoan\_36\_Nguyen();

f\_36\_Nguyen.TopLevel = false;

this.Controls.Add(f\_36\_Nguyen); // Thêm form con vào các điều khiển của form cha

f\_36\_Nguyen.Dock = DockStyle.Fill; // Đặt đầy đủ kích thước của form con

f\_36\_Nguyen.BringToFront(); // Đưa form con lên phía trước

f\_36\_Nguyen.Show(); // Hiển thị form con

}

//Mở form tài chính

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

{

FormTaiChinh\_36\_Nguyen f\_36\_Nguyen = new FormTaiChinh\_36\_Nguyen();

f\_36\_Nguyen.TopLevel = false;

this.Controls.Add(f\_36\_Nguyen); // Thêm form con vào các điều khiển của form cha

f\_36\_Nguyen.Dock = DockStyle.Fill; // Đặt đầy đủ kích thước của form con

f\_36\_Nguyen.BringToFront(); // Đưa form con lên phía trước

f\_36\_Nguyen.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\_36\_Nguyen = new List<Image>(); // Danh sách các ảnh cần thay đổi

public FormMain\_36\_Nguyen()

{

InitializeComponent();

// Thêm các ảnh vào danh sách

images\_36\_Nguyen.Add(Properties.Resources.cake);

images\_36\_Nguyen.Add(Properties.Resources.gioiThieuCake);

images\_36\_Nguyen.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\_36\_Nguyen.Count - 1)

{

currentImageIndex\_36\_Nguyen = 0;

}

else

{

currentImageIndex\_36\_Nguyen++;

}

// Hiển thị ảnh tương ứng trong Panel

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

}

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

public FormMain\_36\_Nguyen()

{

InitializeComponent();

// 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\_36\_Nguyen\_Load(object sender, EventArgs e)

{

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

{

gridBangThoiKhoaBieu\_36\_Nguyen.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\_36\_Nguyen\_Click(object sender, EventArgs e)

{

gridBangThoiKhoaBieu\_36\_Nguyen.Rows.Clear();

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

{

gridBangThoiKhoaBieu\_36\_Nguyen.Rows.Add(i + ". ", i + ". ", i + ". ", i + ". ", i + ". ", i + ". ", i + ". ");

}

}

//Mở bảng grid tkb

private void openToolStripMenuItem2\_36\_Nguyen\_Click(object sender, EventArgs e)

{

OpenFileDialog openFileDialog\_36\_Nguyen = new OpenFileDialog();

openFileDialog\_36\_Nguyen.Filter = "Text files (\*.txt)|\*.txt|All files (\*.\*)|\*.\*";

openFileDialog\_36\_Nguyen.FilterIndex = 1; // mặc định chọn đuôi .txt

if (openFileDialog\_36\_Nguyen.ShowDialog() == DialogResult.OK)

{

try

{

gridBangThoiKhoaBieu\_36\_Nguyen.Rows.Clear();

string filePath\_36\_Nguyen = openFileDialog\_36\_Nguyen.FileName;

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

string[] lines\_36\_Nguyen = File.ReadAllLines(filePath\_36\_Nguyen);

// Bỏ qua dòng đầu tiên

lines\_36\_Nguyen = lines\_36\_Nguyen.Skip(1).ToArray();

foreach (string line in lines\_36\_Nguyen)

{

string[] data\_36\_Nguyen = line.Split('\t'); // Giả sử dữ liệu được phân tách bằng tab (\t)

gridBangThoiKhoaBieu\_36\_Nguyen.Rows.Add(data\_36\_Nguyen);

}

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\_36\_Nguyen(DataGridView dataGridView, string filePath)

{

try

{

using (StreamWriter writer\_36\_Nguyen = new StreamWriter(filePath))

{

// Ghi tiêu đề cột

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

{

writer\_36\_Nguyen.Write(dataGridView.Columns[i].HeaderText);

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

writer\_36\_Nguyen.Write("\t");

}

writer\_36\_Nguyen.WriteLine();

// Ghi dữ liệu từ các dòng

foreach (DataGridViewRow row in dataGridView.Rows)

{

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

{

writer\_36\_Nguyen.Write(row.Cells[i].Value);

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

writer\_36\_Nguyen.Write("\t");

}

writer\_36\_Nguyen.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\_36\_Nguyen\_Click(object sender, EventArgs e)

{

SaveFileDialog saveFileDialog\_36\_Nguyen = new SaveFileDialog();

saveFileDialog\_36\_Nguyen.Filter = "Text files (\*.txt)|\*.txt|File pdf|\*.pdf |All files (\*.\*)|\*.\*";

saveFileDialog\_36\_Nguyen.FilterIndex = 1; //Xác định chỉ số của bộ lọc mặc định được chọn trong danh sách các bộ lọc

saveFileDialog\_36\_Nguyen.RestoreDirectory = true; //Khôi phục thư mục hiện tại khi hộp thoại đóng lại

if (saveFileDialog\_36\_Nguyen.ShowDialog() == DialogResult.OK)

{

SaveToTextFile\_36\_Nguyen(gridBangThoiKhoaBieu\_36\_Nguyen, saveFileDialog\_36\_Nguyen.FileName);

}

}

//Nhấn Exit để đóng app

private void exitToolStripMenuItem2\_36\_Nguyen\_Click(object sender, EventArgs e)

{

DialogResult traLoi\_36\_Nguyen;

traLoi\_36\_Nguyen = MessageBox.Show("Bạn có chắc muốn thoát ứng dụng không?", "Trả lời", MessageBoxButtons.OKCancel, MessageBoxIcon.Question);

if (traLoi\_36\_Nguyen == DialogResult.OK)

{

Application.Exit();

}

}

//FontDialog menu

//Chỉnh font chữ

private void fontEditToolStripMenuItem1\_36\_Nguyen\_Click(object sender, EventArgs e)

{

FontDialog dlg\_36\_Nguyen = new FontDialog();

dlg\_36\_Nguyen.ShowEffects = true;

//dlg.ShowColor = true;

dlg\_36\_Nguyen.ShowApply = true;

if (dlg\_36\_Nguyen.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, v.v.

control.Font = dlg\_36\_Nguyen.Font;

}

}

}

//ColorDialog menu

//Màu chữ

private void fontColorToolStripMenuItem1\_36\_Nguyen\_Click(object sender, EventArgs e)

{

ColorDialog dlg\_36\_Nguyen = new ColorDialog();

dlg\_36\_Nguyen.FullOpen = true; // Cho phép hiển thị tất cả các màu sắc

if (dlg\_36\_Nguyen.ShowDialog() == DialogResult.OK)

{

foreach (Control control in this.Controls)

{

control.ForeColor = dlg\_36\_Nguyen.Color;

}

}

}

//Màu form

private void formColorToolStripMenuItem1\_36\_Nguyen\_Click(object sender, EventArgs e)

{

ColorDialog dlg\_36\_Nguyen = new ColorDialog();

dlg\_36\_Nguyen.FullOpen = true; // Cho phép hiển thị tất cả các màu sắc

if (dlg\_36\_Nguyen.ShowDialog() == DialogResult.OK)

{

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

this.BackColor = dlg\_36\_Nguyen.Color;

}

}

//Màu control

private void controlColorToolStripMenuItem1\_36\_Nguyen\_Click(object sender, EventArgs e)

{

ColorDialog dlg\_36\_Nguyen = new ColorDialog();

dlg\_36\_Nguyen.FullOpen = true;

if (dlg\_36\_Nguyen.ShowDialog() == DialogResult.OK)

{

foreach (Control control in this.Controls)

{

control.BackColor = dlg\_36\_Nguyen.Color;

}

}

}

//Phát nhạc

private SoundPlayer music\_36\_Nguyen; // thuộc tính để lưu nhạc

private void playToolStripMenuItem1\_36\_Nguyen\_Click(object sender, EventArgs e)

{

try

{

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

if (music\_36\_Nguyen != null)

{

// Dừng phát âm thanh nếu đang phát

music\_36\_Nguyen.Stop();

// Giải phóng tài nguyên

music\_36\_Nguyen.Dispose();

}

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

music\_36\_Nguyen = 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\_36\_Nguyen.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\_36\_Nguyen != null)

{

// Dừng phát âm thanh nếu đang phát

music\_36\_Nguyen.Stop();

// Giải phóng tài nguyên

music\_36\_Nguyen.Dispose();

}

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

music\_36\_Nguyen = 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\_36\_Nguyen.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\_36\_Nguyen\_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\_36\_Nguyen\_Click(object sender, EventArgs e)

{

DialogResult traLoi\_36\_Nguyen;

traLoi\_36\_Nguyen = MessageBox.Show("Bạn có chắc muốn về màn hình Login không?", "Trả lời", MessageBoxButtons.OKCancel, MessageBoxIcon.Question);

if (traLoi\_36\_Nguyen == DialogResult.OK)

{

this.Hide();

FormLogin\_36\_Nguyen login = new FormLogin\_36\_Nguyen();

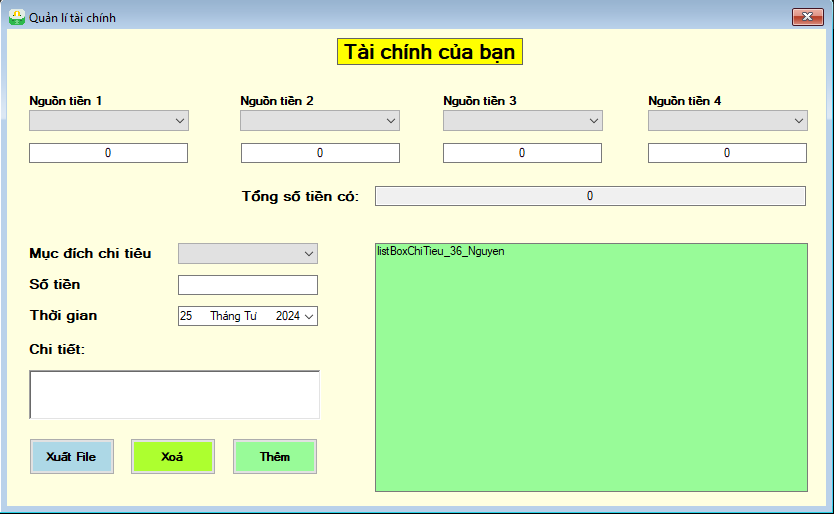
login.ShowDialog();

this.Close();

}

}

1. **Form quản lí tài chính chi tiêu**

****

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Control** | **Loại** | **Thuộc tính** | **Hàm xử lí** |
| **1** | FormTaiChinh\_36\_Nguyen | Form | BackColor: LightYellow  Icon: nhập từ máy  Text: Quản lí tài chính  Startposition: CenterParent | FormTaiChinh\_36\_Nguyen\_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\_36\_Nguyen | Combobox | Archor: top  backColor: Window  DropDownStyle: DropDownList |  |
| **12** | cbNguonTien2\_36\_Nguyen | Combobox | Archor: top  backColor: Window  DropDownStyle: DropDownList |  |
| **13** | cbNguonTien3\_36\_Nguyen | Combobox | Archor: top  backColor: Window  DropDownStyle: DropDownList |  |
| **14** | cbNguonTien4\_36\_Nguyen | Combobox | Archor: top  backColor: Window  DropDownStyle: DropDownList |  |
| **15** | txtNguonTien1\_36\_Nguyen | TextBox | Archor: top  backColor: Window  BorderStyle: Fix3D  Text: 0  TextAlign: Center | txtNguonTien1\_36\_Nguyen\_KeyPress  txtNguonTien1\_36\_Nguyen\_TextChanged |
| **16** | txtNguonTien2\_36\_Nguyen | TextBox | Archor: top  backColor: Window  BorderStyle: Fix3D  Text: 0  TextAlign: Center | txtNguonTien2\_36\_Nguyen\_KeyPress  txtNguonTien2\_36\_Nguyen\_TextChanged |
| **17** | txtNguonTien3\_36\_Nguyen | TextBox | Archor: top  backColor: Window  BorderStyle: Fix3D  Text: 0  TextAlign: Center | txtNguonTien3\_36\_Nguyen\_KeyPress  txtNguonTien3\_36\_Nguyen\_TextChanged |
| **18** | txtNguonTien4\_36\_Nguyen | TextBox | Archor: top  backColor: Window  BorderStyle: Fix3D  Text: 0  TextAlign: Center | txtNguonTien4\_36\_Nguyen\_KeyPress  txtNguonTien4\_36\_Nguyen\_TextChanged |
| **19** | txtTong\_36\_Nguyen | 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\_36\_Nguyen\_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\_36\_Nguyen\_Load(object sender, EventArgs e)

{

cbNguonTien1\_36\_Nguyen.SelectedIndex = 0;

cbNguonTien2\_36\_Nguyen.SelectedIndex = 1;

cbNguonTien3\_36\_Nguyen.SelectedIndex = 2;

cbNguonTien4\_36\_Nguyen.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\_36\_Nguyen()

{

decimal tong\_36\_Nguyen = 0;

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

if (decimal.TryParse(txtNguonTien1\_36\_Nguyen.Text, out decimal source1\_36\_Nguyen))

{

tong\_36\_Nguyen += source1\_36\_Nguyen;

}

if (decimal.TryParse(txtNguonTien2\_36\_Nguyen.Text, out decimal source2\_36\_Nguyen))

{

tong\_36\_Nguyen += source2\_36\_Nguyen;

}

if (decimal.TryParse(txtNguonTien3\_36\_Nguyen.Text, out decimal source3\_36\_Nguyen))

{

tong\_36\_Nguyen += source3\_36\_Nguyen;

}

if (decimal.TryParse(txtNguonTien4\_36\_Nguyen.Text, out decimal source4\_36\_Nguyen))

{

tong\_36\_Nguyen += source4\_36\_Nguyen;

}

// Hiển thị tổng trong TextBox tổng

txtTong\_36\_Nguyen.Text = tong\_36\_Nguyen.ToString();

}

private void txtNguonTien1\_36\_Nguyen\_TextChanged(object sender, EventArgs e)

{

tinhTong\_36\_Nguyen();

}

**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\_36\_Nguyen(string filePath, ListBox listBox)

{

try

{

// Mở tệp văn bản để ghi

using (StreamWriter writer\_36\_Nguyen = new StreamWriter(filePath))

{

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

foreach (var item in listBox.Items)

{

writer\_36\_Nguyen.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\_36\_Nguyen = new SaveFileDialog();

saveFileDialog\_36\_Nguyen.Filter = "Text files (\*.txt)|\*.txt|All files (\*.\*)|\*.\*";

if (saveFileDialog\_36\_Nguyen.ShowDialog() == DialogResult.OK)

{

string filePath\_36\_Nguyen = saveFileDialog\_36\_Nguyen.FileName;

// Gọi phương thức xuất ListBox

xuatGhiFile\_36\_Nguyen(filePath\_36\_Nguyen, 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\_36\_Nguyen.Text) ||

string.IsNullOrEmpty(txtNguonTien2\_36\_Nguyen.Text) ||

string.IsNullOrEmpty(txtNguonTien3\_36\_Nguyen.Text) ||

string.IsNullOrEmpty(txtNguonTien4\_36\_Nguyen.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\_36\_Nguyen.Text);

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

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

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

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

string mucDich\_36\_Nguyen = cbMucDich\_36\_Nguyen.Text;

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

DateTime ngayThucHien\_36\_Nguyen = dateTimeChiTieu\_36\_Nguyen.Value;

string moTa\_36\_Nguyen = richTextBoxChiTiet\_36\_Nguyen.Text;

//Tính tiền còn lại

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

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

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

string item\_36\_Nguyen = $"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\_36\_Nguyen} - Số tiền chi: {soTien\_36\_Nguyen.ToString("C")} - {ngayThucHien\_36\_Nguyen.ToShortDateString()} - {moTa\_36\_Nguyen} - " +

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

// Thêm mục vào ListBox

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

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

txtSoTien\_36\_Nguyen.Clear();

richTextBoxChiTiet\_36\_Nguyen.Clear();

}

## Yêu cầu 6: Nhấn vào Button btnXoa\_36\_Nguyen thì xoá nội dung đang được chọn trong listBox listBoxChiTieu\_36\_Nguyen, nếu chưa chọn báo lỗi

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

{

if (listBoxChiTieu\_36\_Nguyen.SelectedIndex != -1)

{

// Lấy chỉ số của mục được chọn

int mucChon\_36\_Nguyen = listBoxChiTieu\_36\_Nguyen.SelectedIndex;

// Xóa mục được chọn từ ListBox

listBoxChiTieu\_36\_Nguyen.Items.RemoveAt(mucChon\_36\_Nguyen);

}

else

{

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

}

}

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

****

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên control** | **Loại** | **Thuộc tính** | **Hàm xử lí** |
| 1 | FormTaiKhoan\_36\_Nguyen | Form | AcceptButton: btnThemNganHang  BackColor: LightYellow  Icon: nhập từ máy  Text: Tài khoản ngân hàng  StartPosition: CenterParent | FormTaiKhoan\_36\_Nguyen\_Load |
| 2 | lbQuanLyTaiKhoanBank\_36\_Nguyen | 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 | lbSoTaiKhoanNganHang\_36\_Nguyen | Label | Archor: Top, Bottom  AutoSize: true  Text: Nhập số tài khoản  BackColor: LightYellow  Bold: true |  |
| 5 | lbMatKhauNganHang\_36\_Nguyen | Label | Archor: Top, Bottom  AutoSize: true  Text: Nhập mật khẩu ngân hàng  BackColor: LightYellow  Bold: true |  |
| 6 | cbChonTaiKhoanNganHang\_36\_Nguyen | ComboBox | Archor: top, Bottom  backColor: Window  DropDownStyle: DropDownList |  |
| 7 | txtSoTaiKhoanNganHang\_36\_Nguyen | TextBox | Archor: top, Bottom  backColor: Window  BorderStyle: Fix3D  TextAlign: Center |  |
| 8 | txtMatKhauNganHang\_36\_Nguyen | TextBox | Archor: top, Bottom  backColor: Window  BorderStyle: Fix3D  TextAlign: Center |  |
| 9 | txtTimKiem\_36\_Nguyen | TextBox | Archor: top, Bottom  backColor: Window  BorderStyle: Fix3D  TextAlign: Center |  |
| 10 | btnThemNganHang\_36\_Nguyen | Button | Archor: top  BackColor: LightYellow Text: Xuất File  TextAlign: MiddleCenter | btnThemNganHang\_36\_Nguyen\_Click |
| 11 | btnTimKiem\_36\_Nguyen | Button | Archor: top  BackColor: LightYellow Text: Xuất File  TextAlign: MiddleCenter | btnTimKiem\_36\_Nguyen\_Click |
| 12 | groupBoxDs\_36\_Nguyen | GroupBox | Archor: Top, Bottom  AutoSize: false  Text: Danh sách tài khoản  BackColor: LightYellow  Bold: true |  |
| 13 | lvDanhSachNganHang\_36\_Nguyen | ListView | BackColor: Window  Archor: Top, Bottom |  |
| 14 | btnXuatFile\_36\_Nguyen | Button | Archor: bottom  BackColor: PaleGreen  Text: Xuất danh sách  TextAlign: MiddleCenter | btnXuatFile\_36\_Nguyen\_Click |
| 15 | btnMoFile\_36\_Nguyen | Button | Archor: bottom  BackColor: PaleGreen  Text: Mở tệp  TextAlign: MiddleCenter | btnMoFile\_36\_Nguyen\_Click |
| 16 | btnXoaAll\_36\_Nguyen | Button | Archor: bottom  BackColor: Lime  Text: Xoá tất cả  TextAlign: MiddleCenter | btnXoaAll\_36\_Nguyen\_Click |
| 17 | btnXoa\_36\_Nguyen | Button | Archor: bottom  BackColor: Lime  Text: Xoá  TextAlign: MiddleCenter | btnXoa\_36\_Nguyen\_Click |
| 18 | btnSua\_36\_Nguyen | Button | Archor: bottom  BackColor: SlateBlue  Text: Xoá  TextAlign: MiddleCenter | btnSua\_36\_Nguyen\_Click |

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

A screenshot of a computer

Description automatically generated

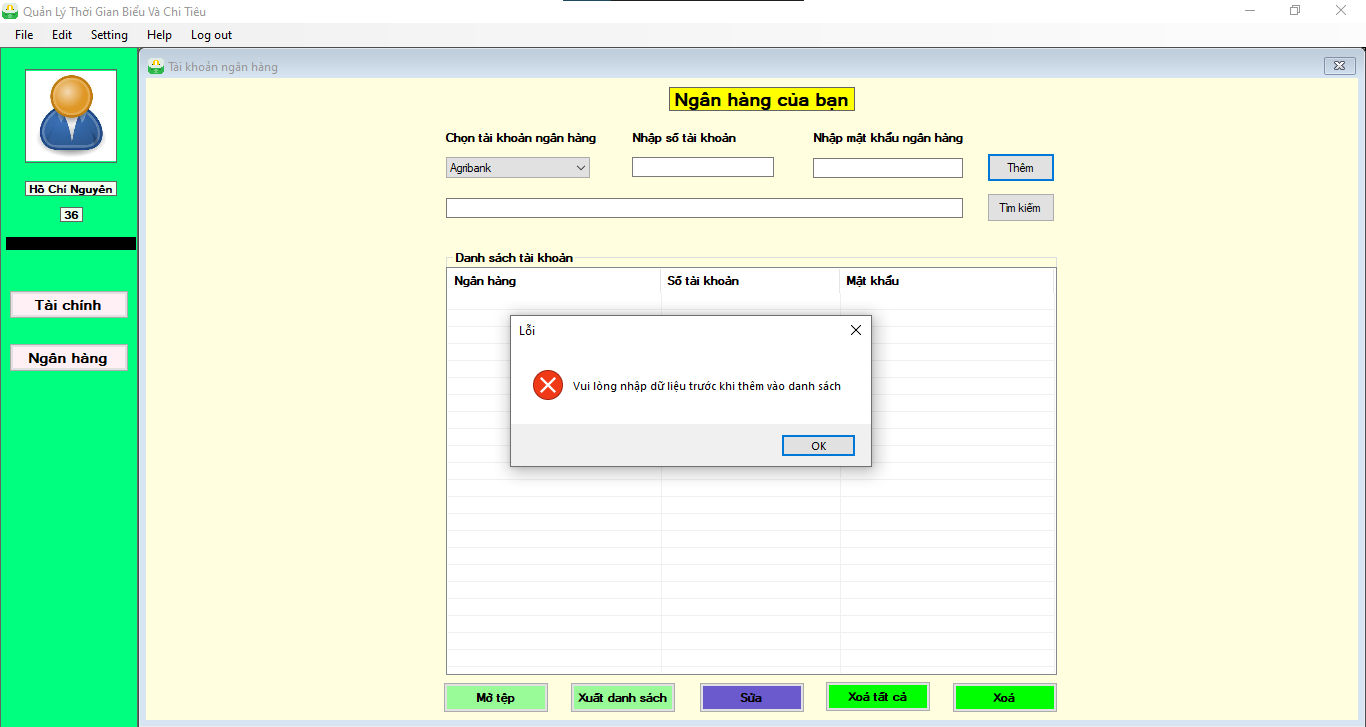
private void FormTaiKhoan\_36\_Nguyen\_Load(object sender, EventArgs e)

{

cbChonTaiKhoanNganHang\_36\_Nguyen.SelectedIndex = 0;

}

## Yêu cầu 2: Nhấn vào Button btnThemNganHang thì thêm nội dung từ cbChonTaiKhoanNganHang\_36\_Nguyen, txtTenTaiKhoanNganHang\_36\_Nguyen và txtMatKhauNganHang\_36\_Nguyen vào ListView lvDanhSachNganHang\_36\_Nguyen. Nếu 1 trong 3 ô không có nội dung thì báo lỗi

****

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

{

String NganHang\_36\_Nguyen = cbChonTaiKhoanNganHang\_36\_Nguyen.SelectedItem.ToString();

String taiKhoan\_36\_Nguyen = txtSoTaiKhoanNganHang\_36\_Nguyen.Text;

String matKhau\_36\_Nguyen = txtMatKhauNganHang\_36\_Nguyen.Text;

// Kiểm tra xem các TextBox có trống hay ko

if (!string.IsNullOrEmpty(NganHang\_36\_Nguyen) && !string.IsNullOrEmpty(taiKhoan\_36\_Nguyen) && !string.IsNullOrEmpty(matKhau\_36\_Nguyen))

{

ListViewItem lvDs\_36\_Nguyen = lvDanhSachNganHang\_36\_Nguyen.Items.Add(cbChonTaiKhoanNganHang\_36\_Nguyen.Text);

lvDs\_36\_Nguyen.SubItems.Add(txtSoTaiKhoanNganHang\_36\_Nguyen.Text);

lvDs\_36\_Nguyen.SubItems.Add(txtMatKhauNganHang\_36\_Nguyen.Text);

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

txtSoTaiKhoanNganHang\_36\_Nguyen.Clear();

txtMatKhauNganHang\_36\_Nguyen.Clear();

txtSoTaiKhoanNganHang\_36\_Nguyen.Focus();

}

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 ListView lvDanhSachNganHang\_36\_Nguyen. Nếu không tìm thấy báo lỗi, nếu tìm thấy thì đánh dấu (Có thể tìm nhiều nội dung giống nhau)

**A screenshot of a computer

Description automatically generated** **A screenshot of a computer

Description automatically generated**

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

{

string searchText\_36\_Nguyen = txtTimKiem\_36\_Nguyen.Text.ToLower(); // Lấy dữ liệu cần tìm kiếm từ TextBox và chuyển thành chữ thường

List<int> mucTimThay\_36\_Nguyen = new List<int>(); // Danh sách để lưu các chỉ số của các mục được tìm thấy

// Duyệt qua từng mục trong ListView

for (int i = 0; i < lvDanhSachNganHang\_36\_Nguyen.Items.Count; i++)

{

ListViewItem item\_36\_Nguyen = lvDanhSachNganHang\_36\_Nguyen.Items[i];

string tenNganHang = item\_36\_Nguyen.SubItems[0].Text.ToLower(); // Cột 0 là tên ngân hàng

string soTaiKhoan = item\_36\_Nguyen.SubItems[1].Text.ToLower(); // Cột 1 là số tài khoản

string matKhau = item\_36\_Nguyen.SubItems[2].Text.ToLower(); // Cột 2 là mật khẩu

// Kiểm tra xem dữ liệu của bất kỳ cột nào chứa dữ liệu tìm kiếm

if (tenNganHang.Contains(searchText\_36\_Nguyen) || soTaiKhoan.Contains(searchText\_36\_Nguyen) || matKhau.Contains(searchText\_36\_Nguyen))

{

mucTimThay\_36\_Nguyen.Add(i); // Thêm chỉ số của mục được tìm thấy vào danh sách

}

}

// Kiểm tra xem có mục nào được tìm thấy hay không

if (mucTimThay\_36\_Nguyen.Count > 0)

{

// Chọn và đảm bảo hiển thị tất cả các mục được tìm thấy trong ListView

foreach (int index in mucTimThay\_36\_Nguyen)

{

lvDanhSachNganHang\_36\_Nguyen.Items[index].Selected = true;

lvDanhSachNganHang\_36\_Nguyen.Items[index].EnsureVisible();

}

}

else

{

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

}

}

## Yêu cầu 4: Nhấn vào button btnXuatFile\_36\_Nguyen thì xuất danh sách nội dung trong ListTaiKhoanNganHang thành file .txt

**A screenshot of a computer

Description automatically generated**

private void xuLiXuatFile\_36\_Nguyen(string filePath, ListView listView)

{

try

{

// Mở tệp văn bản để ghi

using (StreamWriter writer\_36\_Nguyen = new StreamWriter(filePath))

{

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

foreach (ListViewItem item in listView.Items)

{

// Duyệt qua từng cột của mỗi mục và ghi vào tệp

foreach (ListViewItem.ListViewSubItem subItem in item.SubItems)

{

writer\_36\_Nguyen.Write(subItem.Text);

writer\_36\_Nguyen.Write("\t"); // Phân cách giữa các cột

}

writer\_36\_Nguyen.WriteLine(); // Xuống dòng sau khi ghi xong một mục

}

}

MessageBox.Show("Xuất ListView 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\_36\_Nguyen\_Click(object sender, EventArgs e)

{

SaveFileDialog saveFileDialog\_36\_Nguyen = new SaveFileDialog();

saveFileDialog\_36\_Nguyen.Filter = "Text files (\*.txt)|\*.txt|All files (\*.\*)|\*.\*";

if (saveFileDialog\_36\_Nguyen.ShowDialog() == DialogResult.OK)

{

string filePath\_36\_Nguyen = saveFileDialog\_36\_Nguyen.FileName;

// Gọi phương thức xuất ListView

xuLiXuatFile\_36\_Nguyen(filePath\_36\_Nguyen, lvDanhSachNganHang\_36\_Nguyen);

}

}

## Yêu cầu 5: Nhấn vào Button btnMoFile\_36\_Nguyen thì mở file txt danh sách tài khoản có sẵn

private void xuLiMoFile\_36\_Nguyen(string filePath, ListView listView)

{

try

{

// Kiểm tra xem tệp tồn tại trước khi đọc

if (File.Exists(filePath))

{

// Xóa dữ liệu cũ trong ListView trước khi tải dữ liệu mới

listView.Items.Clear();

// Mở tệp văn bản để đọc

using (StreamReader reader = new StreamReader(filePath))

{

string line\_36\_Nguyen;

while ((line\_36\_Nguyen = reader.ReadLine()) != null)

{

// Tách các giá trị trong mỗi dòng bằng dấu tab

string[] values\_36\_Nguyen = line\_36\_Nguyen.Split('\t');

// Tạo một ListViewItem mới

ListViewItem item\_36\_Nguyen = new ListViewItem(values\_36\_Nguyen[0]); // Giá trị đầu tiên là cho cột đầu tiên

for (int i = 1; i < values\_36\_Nguyen.Length; i++)

{

// Thêm các giá trị còn lại vào các SubItem (cột phụ)

item\_36\_Nguyen.SubItems.Add(values\_36\_Nguyen[i]);

}

// Thêm ListViewItem vào ListView

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

}

}

}

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\_36\_Nguyen\_Click(object sender, EventArgs e)

{

OpenFileDialog openFileDialog\_36\_Nguyen = new OpenFileDialog();

openFileDialog\_36\_Nguyen.Filter = "Text files (\*.txt)|\*.txt|All files (\*.\*)|\*.\*";

if (openFileDialog\_36\_Nguyen.ShowDialog() == DialogResult.OK)

{

string filePath = openFileDialog\_36\_Nguyen.FileName;

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

lvDanhSachNganHang\_36\_Nguyen.Items.Clear();

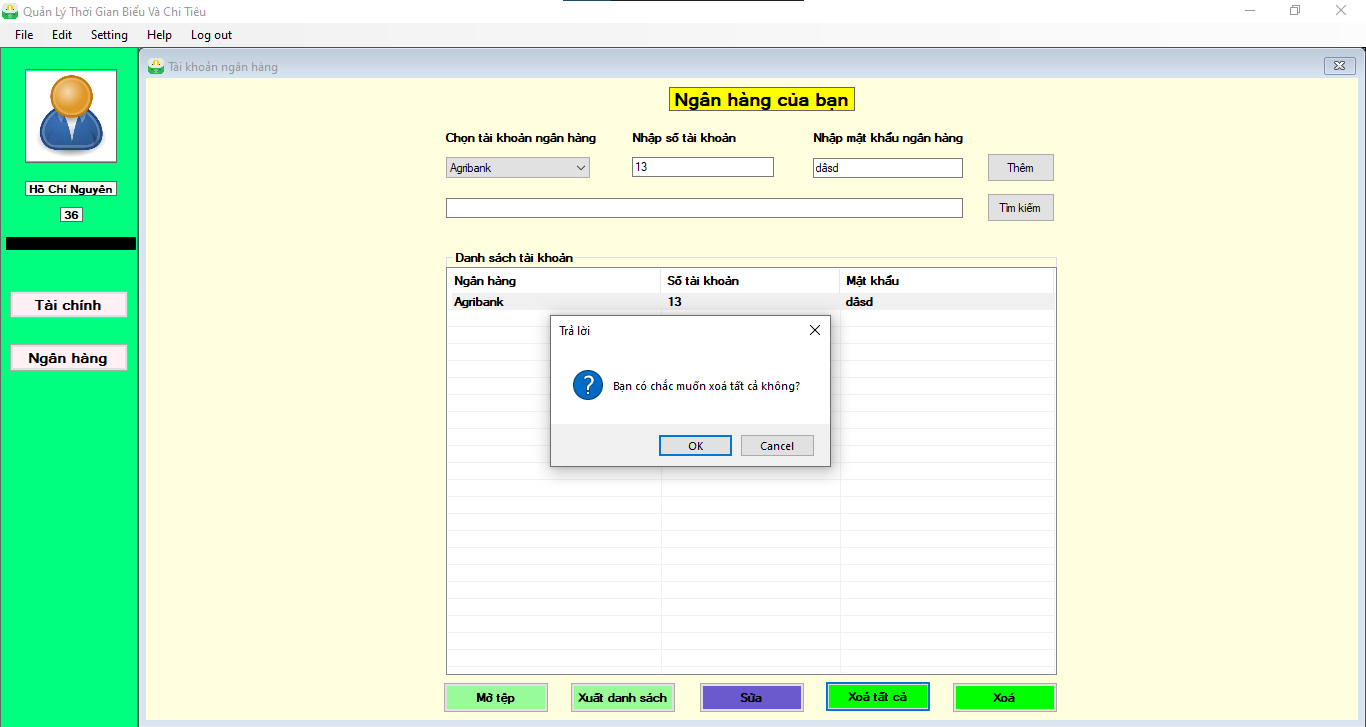
// Gọi phương thức để tải dữ liệu từ tệp vào ListView

xuLiMoFile\_36\_Nguyen(filePath, lvDanhSachNganHang\_36\_Nguyen);

}

}

## Yêu cầu 6: Nhấn vào button btnXoaAll\_36\_Nguyen thì Xoá tất cả nội dung lvDanhSachNganHang\_36\_Nguyen (có hỏi lại chắc chắn không)



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

{

DialogResult traLoiXoaAll\_36\_Nguyen;

traLoiXoaAll\_36\_Nguyen = MessageBox.Show("Bạn có chắc muốn xoá tất cả không?", "Trả lời", MessageBoxButtons.OKCancel, MessageBoxIcon.Question);

if (traLoiXoaAll\_36\_Nguyen == DialogResult.OK)

{

lvDanhSachNganHang\_36\_Nguyen.Items.Clear();

}

}

## Yêu cầu 7: Nhấn vào button btnXoa\_36\_Nguyen thì xoá một nội dung đang chọn trong lvDanhSachNganHang\_36\_Nguyen. Nếu chưa chọn nội dung thì báo lỗi

**A screenshot of a computer

Description automatically generated**

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

{

// Kiểm tra xem có mục nào được chọn không

if (lvDanhSachNganHang\_36\_Nguyen.SelectedItems.Count > 0)

{

// Xóa mục được chọn từ ListBox

lvDanhSachNganHang\_36\_Nguyen.Items.Remove(lvDanhSachNganHang\_36\_Nguyen.SelectedItems[0]);

}

else

{

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

}

}

## Yêu cầu 8: Nhấn vào Button btnSua\_36\_Nguyen thì lấy nội dung từ 3 ô cbChonTaiKhoanNganHang\_36\_Nguyen, txtTenTaiKhoanNganHang\_36\_Nguyen và txtMatKhauNganHang\_36\_Nguyen để cập nhật lại 1 nội dung được chọn trong lvDanhSachNganHang\_36\_Nguyen

## \*Lưu ý phải viết phương thức xử lí khi chọn 1 nội dung trong lvDanhSachNganHang\_36\_Nguyen thì lấy nội dung được chọn cập nhật lên 3 ô cbChonTaiKhoanNganHang\_36\_Nguyen, txtTenTaiKhoanNganHang\_36\_Nguyen và txtMatKhauNganHang\_36\_Nguyen

A screenshot of a computer

Description automatically generated

// chọn 1 nội dung trong lvDanhSachNganHang\_36\_Nguyen thì lấy nội dung được chọn cập nhật lên 3 ô

private void lvDanhSachNganHang\_36\_Nguyen\_SelectedIndexChanged(object sender, EventArgs e)

{

if (lvDanhSachNganHang\_36\_Nguyen.SelectedItems.Count > 0)

{

cbChonTaiKhoanNganHang\_36\_Nguyen.Text = lvDanhSachNganHang\_36\_Nguyen.SelectedItems[0].SubItems[0].Text;

txtSoTaiKhoanNganHang\_36\_Nguyen.Text = lvDanhSachNganHang\_36\_Nguyen.SelectedItems[0].SubItems[1].Text;

txtMatKhauNganHang\_36\_Nguyen.Text = lvDanhSachNganHang\_36\_Nguyen.SelectedItems[0].SubItems[2].Text;

}

}

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

{

if (lvDanhSachNganHang\_36\_Nguyen.SelectedItems.Count > 0)

{

lvDanhSachNganHang\_36\_Nguyen.SelectedItems[0].SubItems[0].Text = cbChonTaiKhoanNganHang\_36\_Nguyen.Text;

lvDanhSachNganHang\_36\_Nguyen.SelectedItems[0].SubItems[1].Text = txtSoTaiKhoanNganHang\_36\_Nguyen.Text;

lvDanhSachNganHang\_36\_Nguyen.SelectedItems[0].SubItems[2].Text = txtMatKhauNganHang\_36\_Nguyen.Text;

}

else

MessageBox.Show("Vui lòng chọn một nội dung trong danh sách tài khoản để sửa", "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);

}