**TRƯỜNG ĐẠI HỌC KINH DOANH VÀ CÔNG NGHỆ HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

🙣🙜🕮🙞🙡



**ĐỒ ÁN VB**

**QUẢN LÝ SINH VIÊN**

**Sinh viên thực hiện : Nguyễn Tuấn Vũ**

**Mã sinh viên : 2924202735**

**Lớp : TH29.14**

**Hà Nội – 2025**

**MỤC LỤC**

[**LỜI CẢM ƠN** 2](https://docs.google.com/document/d/1NCBta47QEBjvrOQHVbi9fg2w_ZXjYBjT/edit#heading=h.mg7ahjpwa238)

[**CHƯƠNG 1: TỔNG QUAN VỀ VISUAL BASIC 3**](https://docs.google.com/document/d/1NCBta47QEBjvrOQHVbi9fg2w_ZXjYBjT/edit#heading=h.yin8szg2tcta)

[1.1. Visual Basic là gì? 3](https://docs.google.com/document/d/1NCBta47QEBjvrOQHVbi9fg2w_ZXjYBjT/edit#heading=h.li3h103gsg3c)

[1.2. Mục đích sử dụng Visual Basic 3](https://docs.google.com/document/d/1NCBta47QEBjvrOQHVbi9fg2w_ZXjYBjT/edit#heading=h.bqrpuzrs4klj)

[1.3. Hệ Quản Trị Cơ Sở Dữ Liệu SQL Server 4](https://docs.google.com/document/d/1NCBta47QEBjvrOQHVbi9fg2w_ZXjYBjT/edit#heading=h.xfqvfkgtvcra)

[**CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ 5**](https://docs.google.com/document/d/1NCBta47QEBjvrOQHVbi9fg2w_ZXjYBjT/edit#heading=h.e8xnjmafn10r)

[2.1  Phân tích các chức năng 5](https://docs.google.com/document/d/1NCBta47QEBjvrOQHVbi9fg2w_ZXjYBjT/edit#heading=h.hyhr6vln2q2f)

[2.2 Cơ sở dữ liệu 6](https://docs.google.com/document/d/1NCBta47QEBjvrOQHVbi9fg2w_ZXjYBjT/edit#heading=h.lndv6l64nn7t)

[2.3 Thiết kế các bảng 6](https://docs.google.com/document/d/1NCBta47QEBjvrOQHVbi9fg2w_ZXjYBjT/edit#heading=h.abs3r2l0v2pw)

[**CHƯƠNG 3: CHƯƠNG TRÌNH QUẢN LÝ NHÂN VIÊN 8**](https://docs.google.com/document/d/1NCBta47QEBjvrOQHVbi9fg2w_ZXjYBjT/edit#heading=h.naq37748rk3n)

[3.1 Giao diện và mô tả 8](https://docs.google.com/document/d/1NCBta47QEBjvrOQHVbi9fg2w_ZXjYBjT/edit#heading=h.2k9gdnq6rjhm)

[3.2 Một số đoạn mã code cơ bản trong chương trình 11](https://docs.google.com/document/d/1NCBta47QEBjvrOQHVbi9fg2w_ZXjYBjT/edit#heading=h.aoahvhrt904e)

[**CHƯƠNG 4: KẾT LUẬN 22**](https://docs.google.com/document/d/1NCBta47QEBjvrOQHVbi9fg2w_ZXjYBjT/edit#heading=h.5r9p2272n491)

**LỜI CẢM ƠN**

Lời đầu tiên, em xin chân thành gửi lời cảm ơn sâu sắc đến quý thầy cô trong khoa Công nghệ thông tin – Trường đại học Kinh Doanh và Công Nghệ Hà Nội, đặc biệt là giảng viên hướng dẫn đã tận tình chỉ bảo và tạo điều kiện thuận lợi để em có thể hoàn thành tốt đồ án môn Công nghệ Thông tin.

Em cũng xin trân trọng cảm ơn quý thầy cô đã truyền đạt cho em những kiến thức quý báu trong suốt quá trình học tập. Đây là nền tảng vững chắc để em vận dụng vào quá trình nghiên cứu và thực hiện đồ án này.

Bên cạnh đó, em xin cảm ơn đến gia đình và bạn bè đã luôn đồng hành, động viên và hỗ trợ em về cả tinh thần lẫn vật chất trong suốt quá trình học tập và thực hiện đồ án.

Do thời gian hạn chế và kinh nghiệm thực tế chưa nhiều, chắc chắn không thể tránh khỏi những thiếu sót. Em rất mong nhận được sự đóng góp ý kiến và chỉ bảo từ quý thầy cô và các bạn để em có thể hoàn thiện tốt hơn trong tương lai.

Một lần nữa, em xin chân thành cảm ơn!

**CHƯƠNG 1: TỔNG QUAN VỀ VISUAL BASIC**

**1.1. Visual Basic là gì?**

Visual Basic (VB) là một ngôn ngữ lập trình được phát triển bởi Microsoft, dựa trên ngôn ngữ Basic truyền thống. Đây là ngôn ngữ lập trình hướng sự kiện, cho phép người dùng xây dựng ứng dụng bằng cách kết hợp giữa việc viết mã lệnh và sử dụng công cụ thiết kế giao diện trực quan. Với môi trường phát triển tích hợp Visual Studio, Visual Basic trở thành một trong những công cụ lập trình mạnh mẽ, dễ học và dễ sử dụng.

Ngôn ngữ này hỗ trợ đầy đủ các đặc điểm của lập trình hướng đối tượng như lớp, kế thừa và đa hình. Ngoài ra, Visual Basic còn có khả năng tích hợp chặt chẽ với .NET Framework, giúp phát triển ứng dụng nhanh chóng, linh hoạt và có thể kết nối với nhiều hệ quản trị cơ sở dữ liệu khác nhau, đặc biệt là SQL Server.

**1.2. Mục đích sử dụng Visual Basic**

Mục đích chính của Visual Basic là hỗ trợ xây dựng các ứng dụng quản lý và ứng dụng chạy trên nền tảng Windows. Với ưu điểm dễ sử dụng và khả năng phát triển nhanh, Visual Basic được ứng dụng rộng rãi trong nhiều lĩnh vực khác nhau.

Cụ thể, Visual Basic thường được sử dụng để:

* Phát triển các phần mềm quản lý như quản lý nhân sự, quản lý bán hàng, quản lý kho, quản lý trường học…
* Thiết kế giao diện người dùng trực quan, dễ thao tác, thân thiện với người sử dụng.
* Xử lý các nghiệp vụ tính toán, thống kê, lập báo cáo trong doanh nghiệp.
* Kết nối và làm việc trực tiếp với cơ sở dữ liệu thông qua SQL Server hoặc các hệ quản trị cơ sở dữ liệu khác.
* Hỗ trợ phát triển phần mềm nhanh chóng, dễ dàng bảo trì và mở rộng.

Chính vì những ưu điểm trên, Visual Basic được lựa chọn nhiều trong các đồ án, dự án thực tế có tính chất quản lý dữ liệu.

**1.3. Hệ Quản Trị Cơ Sở Dữ Liệu SQL Server**

SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System – RDBMS) do Microsoft phát triển. Đây là công cụ mạnh mẽ, được sử dụng rộng rãi trong việc lưu trữ, quản lý và truy xuất dữ liệu.

SQL Server tổ chức dữ liệu theo mô hình quan hệ gồm các bảng, cột và hàng, đồng thời hỗ trợ ngôn ngữ SQL (Structured Query Language) để thực hiện các thao tác thêm, sửa, xóa và tìm kiếm dữ liệu. Với khả năng xử lý khối lượng dữ liệu lớn và nhiều người dùng truy cập đồng thời, SQL Server đảm bảo được tính toàn vẹn, bảo mật và hiệu suất trong quá trình hoạt động.

Trong thực tế, SQL Server thường được sử dụng để lưu trữ dữ liệu của các hệ thống quản lý như thông tin nhân viên, khách hàng, sản phẩm… Khi kết hợp với Visual Basic, SQL Server đóng vai trò là nơi lưu trữ dữ liệu trung tâm, còn Visual Basic đảm nhận việc xây dựng giao diện và xử lý các nghiệp vụ quản lý. Sự kết hợp này mang lại cho người dùng một hệ thống vừa mạnh mẽ, vừa dễ sử dụng và có thể áp dụng hiệu quả trong nhiều bài toán quản lý.

**CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ**

***2.1 Phân tích các chức năng***

*Chức năng quản trị hệ thống*

* Cho phép người quản trị hệ thống quản lý người sử dụng, phân quyền truy cập theo vai trò (quản trị, giáo vụ, giảng viên, sinh viên).
* Quản lý danh mục hệ thống (khoa, lớp, môn học).
* Giám sát và vận hành hệ thống ổn định, an toàn.

*Chức năng quản lý thông tin*

* **Quản lý sinh viên:** thêm mới, chỉnh sửa, xóa sinh viên; quản lý mã sinh viên, khoa, lớp, thông tin cá nhân.
* **Quản lý thông tin sinh viên:** lưu trữ và cập nhật hồ sơ cá nhân (họ tên, ngày sinh, giới tính, địa chỉ, số điện thoại, email, lớp, khoa).
* **Quản lý khoa:** thêm mới, chỉnh sửa, xóa khoa; gán sinh viên và giảng viên thuộc khoa đó.
* **Quản lý môn học:** thêm mới, chỉnh sửa, xóa môn học; theo dõi số tín chỉ và điểm trung bình của sinh viên.
* **Quản lý kết quả học tập:** nhập điểm, sửa điểm, thống kê điểm trung bình từng môn hoặc từng sinh viên.

*Chức năng tra cứu thông tin*

* Cho phép người dùng (theo quyền hạn được phân) tra cứu thông tin sinh viên, khoa, môn học.
* Tìm kiếm nhanh sinh viên theo mã, tên hoặc khoa.
* Xem báo cáo thống kê: số lượng sinh viên theo khoa, danh sách sinh viên có điểm cao, danh sách sinh viên cần cải thiện điểm.

*Yêu cầu miền ứng dụng*

* Chạy được trên nhiều hệ điều hành khác nhau (Windows, Linux).
* Hệ quản trị cơ sở dữ liệu tập trung (SQL Server hoặc tương đương).
* Giao diện thiết kế thống nhất, thân thiện, dễ sử dụng.

***2.2 Cơ sở dữ liệu***

1. **Thông tin về Khoa**
   * MaKhoa
   * TenKhoa
   * SDT
   * DiaChi
2. **Thông tin về SinhVien**
   * MaSV
   * HoTen
   * NgaySinh
   * GioiTinh
   * DiaChi
   * SDT
   * Email
   * MaKhoa
3. **Thông tin về MonHoc**
   * MaMH
   * TenMH
   * SoTinChi
4. **Thông tin về KetQua (nếu có)**
   * MaSV
   * MaMH
   * Diem

***2.3 Thiết kế các bảng***

Cơ sở dữ liệu gồm có 3 bảng chính:

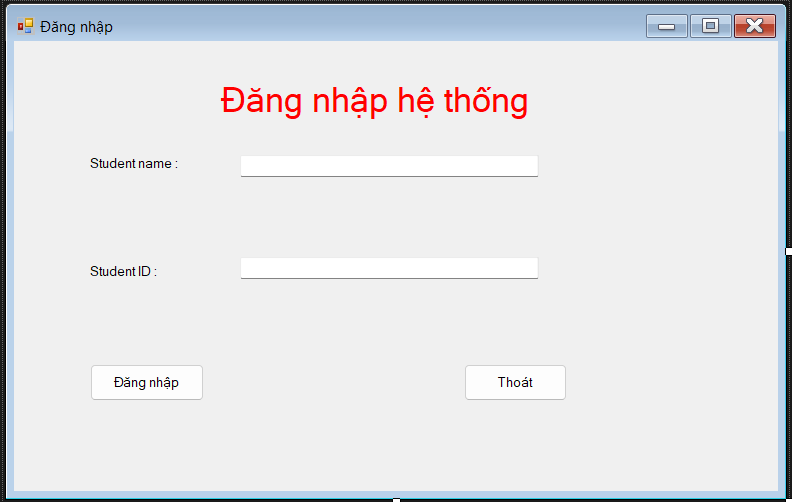
* **Khoa**
* **SinhVien**
* **MonHoc**  
  (Bảng **KetQua** dùng để lưu kết quả học tập của sinh viên)

* CREATE DATABASE StudentDB;
* GO
* -- Sử dụng database vừa tạo
* USE StudentDB;
* GO
* -- Tạo bảng Classes trước
* CREATE TABLE Classes (
* ClassID   VARCHAR(20) PRIMARY KEY,
* ClassName NVARCHAR(100) NOT NULL
* );
* GO
* -- Tạo bảng Students, có ràng buộc khóa ngoại tới Classes
* CREATE TABLE Students (
* StudentID   VARCHAR(20) PRIMARY KEY,
* StudentName NVARCHAR(100) NOT NULL,
* Phone       VARCHAR(20),
* Address     NVARCHAR(100),
* ClassID     VARCHAR(20),
* FOREIGN KEY (ClassID) REFERENCES Classes(ClassID)
* );
* GO
* -- Thêm dữ liệu vào bảng Classes
* INSERT INTO Classes (ClassID, ClassName) VALUES
* ('PT09031', N'LAP TRINH UNG DUNG'),
* ('PT09033', N'LAP TRINH MOBILE');
* GO
* -- Thêm dữ liệu vào bảng Students
* INSERT INTO Students (StudentID, StudentName, Phone, Address, ClassID) VALUES
* ('535', 'retyer', 'fethf', 'ertyev', 'PT09031'),
* ('dfgsd', 'dsfsdfgsdfgsdfgsdfgsdfg', '324534654', 'dfgsdfg', NULL),
* ('dfsdf', 'sdrgssfdg', 'fsdg', 'sdfgsdfg', 'PT09031'),
* ('PH00001', N'Nguyen Trong Tho', '0989787731', N'HCM', 'PT09031'),
* ('PH00002', N'Tran Hop Minh Nghia', '0989787700', N'Ha Gian', 'PT09031'),
* ('PH00003', N'Nguyen Van An', '0989787731', N'Hanoi', 'PT09031'),
* ('PH00004', N'Tran Dinh Trong', '0989787700', N'Ha Gian', 'PT09031'),
* ('PH0010', N'Nguyen Van An', '098988877', N'Hanoi', 'PT09031'),
* ('PH0011', N'Nguyen Van An', '098988877', N'Hanoi', 'PT09031'),
* ('PH1111', N'NGuyen Van Hung', '9088989089', N'Hanoi', 'PT09031');
* GO

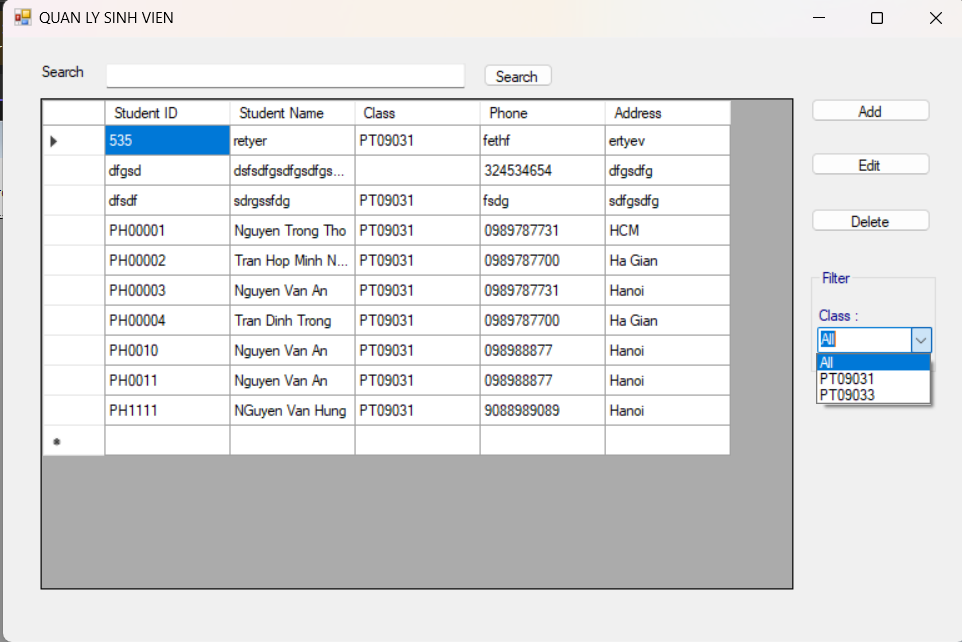
**CHƯƠNG 3: CHƯƠNG TRÌNH QUẢN LÝ NHÂN VIÊN**

**3.1 Giao diện và mô tả**

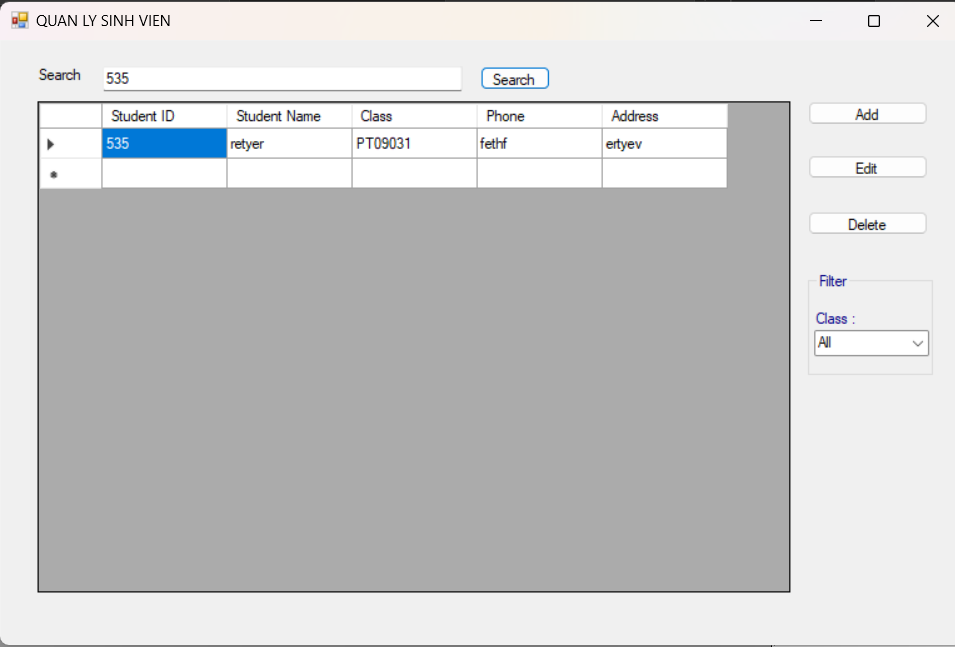
* **Thiết kế giao diện và liên kết với Database**
* Giao diện đăng nhập



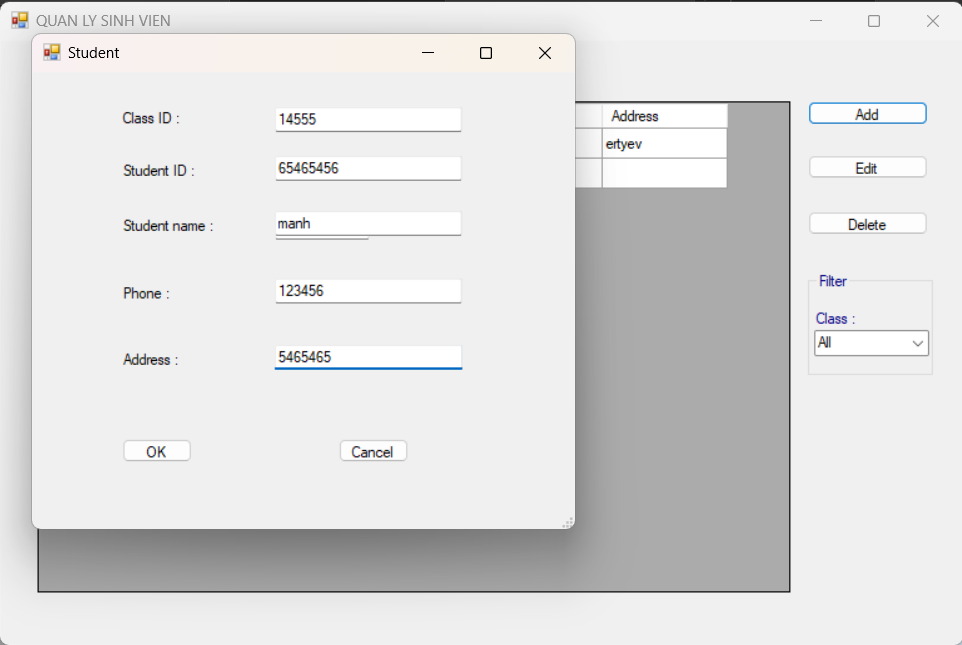
Giao diện quản lý sinh viên



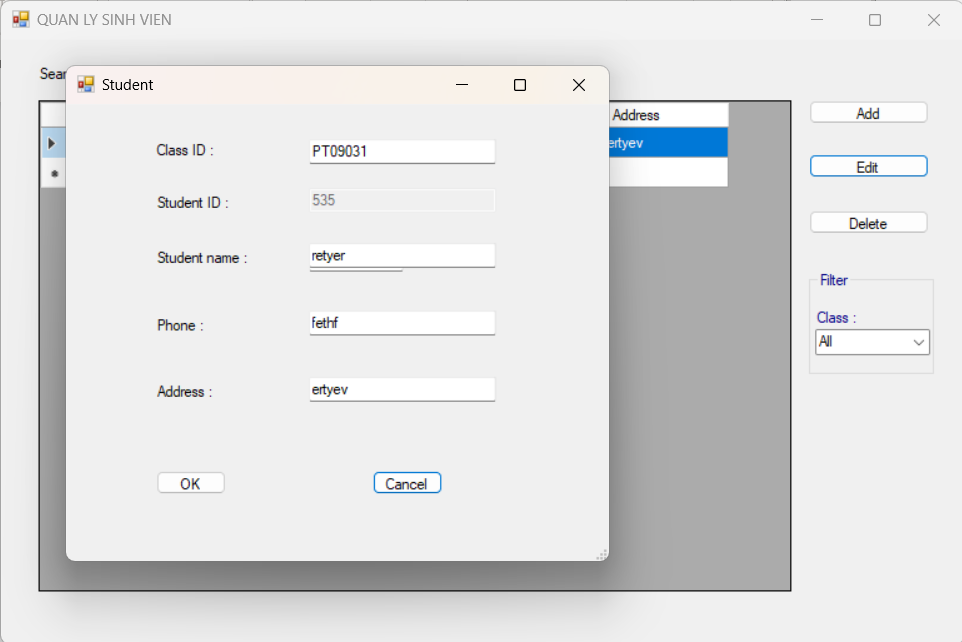
Chức năng tìm kiếm



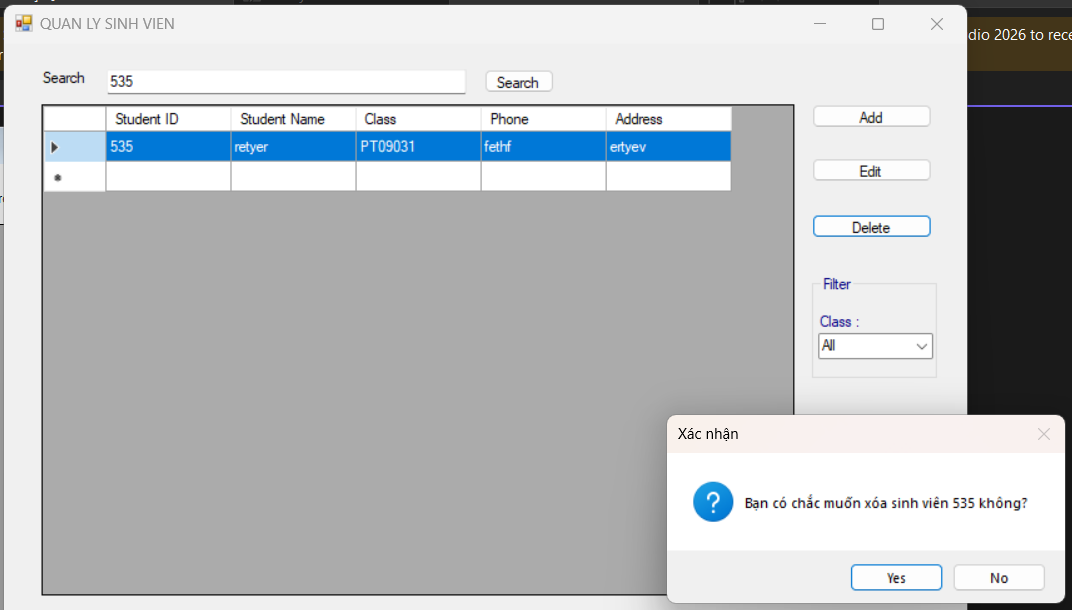
Chức năng thêm



Chức năng sửa



Chức năng xoá



**3.2 Một số đoạn mã code cơ bản trong chương trình**

**1. Giao diện chính quản lý sinh viên**

Imports System.Data.SqlClient

Public Class frmQLSV

    ' Kết nối SQL Server

    Private conn As New SqlConnection("Data Source=DESKTOP-LS6SVQ2\SQLEXPRESS01;Initial Catalog=StudentDB;Integrated Security=True;Encrypt=False")

    Private Sub frmMain\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

        Me.StartPosition = FormStartPosition.CenterScreen

        LoadData()

        LoadClassList()

    End Sub

    ' =======================================================

    ' ======= TẢI DỮ LIỆU SINH VIÊN LÊN BẢNG (LoadData) =======

    ' =======================================================

    Private Sub LoadData(Optional ByVal search As String = "", Optional ByVal classFilter As String = "")

        Try

            ' 🟢 SỬA LỖI TÊN CỘT: Sử dụng tên cột chính xác trong SQL (StudentID, StudentName, ClassID,...)

            ' Chú ý: Tên bảng trong FROM đã là Students (đúng)

            Dim sql As String = "SELECT StudentID AS [Student ID], StudentName AS [Student Name], ClassID AS [Class], Phone AS [Phone], Address AS [Address] FROM Students WHERE 1=1"

            If search <> "" Then

                ' 🟢 SỬA LỖI TÊN CỘT: MaSV -> StudentID, TenSV -> StudentName

                sql &= " AND (StudentID LIKE @Search OR StudentName LIKE @Search)"

            End If

            If classFilter <> "" And classFilter <> "All" Then

                ' 🟢 SỬA LỖI TÊN CỘT: Lop -> ClassID

                sql &= " AND ClassID = @Class"

            End If

            Using cmd As New SqlCommand(sql, conn)

                If search <> "" Then cmd.Parameters.AddWithValue("@Search", "%" & search & "%")

                If classFilter <> "" And classFilter <> "All" Then cmd.Parameters.AddWithValue("@Class", classFilter)

                Dim da As New SqlDataAdapter(cmd)

                Dim dt As New DataTable()

                da.Fill(dt)

                dgvStudents.DataSource = dt

            End Using

        Catch ex As Exception

            MessageBox.Show("Lỗi tải dữ liệu: " & ex.Message)

        End Try

    End Sub

    ' =========================================================

    ' ======= NẠP DANH SÁCH LỚP VÀO COMBOBOX (LoadClassList) =======

    ' =========================================================

    Private Sub LoadClassList()

        Try

            cboClass.Items.Clear()

            cboClass.Items.Add("All")

            ' 🟢 SỬA TÊN BẢNG: Students -> Classes để lấy TẤT CẢ các lớp

            Using cmd As New SqlCommand("SELECT ClassID FROM Classes", conn)

                conn.Open()

                Using rdr As SqlDataReader = cmd.ExecuteReader()

                    While rdr.Read()

                        ' Tên cột trong bảng Classes là ClassID

                        cboClass.Items.Add(rdr("ClassID").ToString())

                    End While

                End Using

                conn.Close()

            End Using

            cboClass.SelectedIndex = 0

        Catch ex As Exception

            If conn.State = ConnectionState.Open Then conn.Close()

        End Try

    End Sub

    ' ===================================

    ' ======= NÚT THÊM SINH VIÊN =======

    ' ===================================

    Private Sub btnAdd\_Click(sender As Object, e As EventArgs) Handles btnAdd.Click

        Dim f As New frmStudent()

        f.IsEdit = False

        If f.ShowDialog() = DialogResult.OK Then

            LoadData()

            LoadClassList() ' Có thể cần nạp lại danh sách lớp nếu có lớp mới được thêm

        End If

    End Sub

    ' ===================================

    ' ======= NÚT SỬA SINH VIÊN =======

    ' ===================================

    Private Sub btnEdit\_Click(sender As Object, e As EventArgs) Handles btnEdit.Click

        If dgvStudents.SelectedRows.Count = 0 Then

            MessageBox.Show("Vui lòng chọn sinh viên để sửa!", "Thông báo")

            Return

        End If

        ' Tên Cell trong DataGridView đã được đặt alias là "Student ID"

        Dim maSV As String = dgvStudents.SelectedRows(0).Cells("Student ID").Value.ToString()

        Dim f As New frmStudent()

        f.IsEdit = True

        f.MaSVCanSua = maSV

        If f.ShowDialog() = DialogResult.OK Then

            LoadData()

            LoadClassList() ' Có thể cần nạp lại danh sách lớp nếu lớp bị sửa

        End If

    End Sub

    ' ===================================

    ' ======= NÚT XÓA SINH VIÊN =======

    ' ===================================

    Private Sub btnDelete\_Click(sender As Object, e As EventArgs) Handles btnDelete.Click

        If dgvStudents.SelectedRows.Count = 0 Then

            MessageBox.Show("Vui lòng chọn sinh viên để xóa!", "Thông báo")

            Return

        End If

        Dim maSV As String = dgvStudents.SelectedRows(0).Cells("Student ID").Value.ToString()

        If MessageBox.Show("Bạn có chắc muốn xóa sinh viên " & maSV & " không?", "Xác nhận", MessageBoxButtons.YesNo, MessageBoxIcon.Question) = DialogResult.Yes Then

            Try

                ' 🟢 SỬA LỖI TÊN BẢNG VÀ TÊN CỘT TRONG DELETE

                ' SinhVien -> Students, MaSV -> StudentID

                Using cmd As New SqlCommand("DELETE FROM Students WHERE StudentID = @MaSV", conn)

                    cmd.Parameters.AddWithValue("@MaSV", maSV)

                    conn.Open()

                    cmd.ExecuteNonQuery()

                    conn.Close()

                End Using

                MessageBox.Show("Đã xóa sinh viên thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information)

                LoadData()

                LoadClassList() ' Nạp lại danh sách lớp sau khi xóa

            Catch ex As Exception

                If conn.State = ConnectionState.Open Then conn.Close()

                MessageBox.Show("Lỗi khi xóa: " & ex.Message)

            End Try

        End If

    End Sub

    ' ======================================

    ' ======= TÌM KIẾM & LỌC DỮ LIỆU =======

    ' ======================================

    Private Sub btnSearch\_Click(sender As Object, e As EventArgs) Handles btnSearch.Click

        LoadData(txtSearch.Text.Trim(), cboClass.Text)

    End Sub

    Private Sub cboClass\_SelectedIndexChanged(sender As Object, e As EventArgs) Handles cboClass.SelectedIndexChanged

        LoadData(txtSearch.Text.Trim(), cboClass.Text)

    End Sub

End Class

**2. Giao diện đăng nhập**

Imports System.Data.SqlClient

Public Class frmDangNhap

    ' 🟢 1. KHAI BÁO KẾT NỐI (dùng lại chuỗi kết nối đã sửa)

    Private conn As New SqlConnection("Data Source=DESKTOP-LS6SVQ2\SQLEXPRESS01;Initial Catalog=StudentDB;Integrated Security=True;Encrypt=False")

    Private Sub Form1\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

        Me.StartPosition = FormStartPosition.CenterScreen

    End Sub

    Private Sub btnDangNhap\_Click(sender As Object, e As EventArgs) Handles btnDangNhap.Click

        Dim tenSV As String = txtStudentName.Text.Trim()

        Dim maSV As String = txtStudentID.Text.Trim()

        If tenSV = "" Or maSV = "" Then

            MessageBox.Show("Vui lòng nhập đầy đủ thông tin!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning)

            Return ' Thoát khỏi Sub nếu thiếu thông tin

        End If

        ' 🟢 2. THỰC HIỆN XÁC THỰC VỚI CƠ SỞ DỮ LIỆU

        Try

            ' 🟢 Truy vấn bảng Students (tên bảng chính xác trong database)

            ' Kiểm tra sự tồn tại của cặp StudentID và StudentName (giả sử tên là duy nhất)

            Dim sql As String = "SELECT COUNT(\*) FROM Students WHERE StudentID = @MaSV AND StudentName = @TenSV"

            Using cmd As New SqlCommand(sql, conn)

                cmd.Parameters.AddWithValue("@MaSV", maSV) ' Tham số cho StudentID

                cmd.Parameters.AddWithValue("@TenSV", tenSV) ' Tham số cho StudentName

                conn.Open()

                Dim result As Integer = CInt(cmd.ExecuteScalar())

                conn.Close()

                If result > 0 Then

                    ' Đăng nhập thành công

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

                    ' 🟢 Mở Form quản lý sinh viên (frmQLSV) và ẩn form đăng nhập

                    ' Giả sử frmQLSV là Form chính mà bạn muốn hiển thị sau khi đăng nhập

                    Me.Hide()

                    Dim mainForm As New frmQLSV()

                    mainForm.Show()

                Else

                    ' Đăng nhập thất bại

                    MessageBox.Show("Mã sinh viên hoặc Tên sinh viên không đúng!", "Lỗi Đăng nhập", MessageBoxButtons.OK, MessageBoxIcon.Error)

                End If

            End Using

        Catch ex As Exception

            If conn.State = ConnectionState.Open Then conn.Close()

            MessageBox.Show("Lỗi kết nối hoặc truy vấn cơ sở dữ liệu: " & ex.Message, "Lỗi Hệ thống", MessageBoxButtons.OK, MessageBoxIcon.Error)

        End Try

    End Sub

End Class

**3. Giao diện sinh viên**

Imports System.Data.SqlClient

Public Class frmStudent

    ' ► KHAI BÁO CHUNG (phải ở cấp class)

    ' (Chuỗi kết nối đã được sửa lỗi Trust Server Certificate)

    Private conn As New SqlConnection("Data Source=DESKTOP-LS6SVQ2\SQLEXPRESS01;Initial Catalog=StudentDB;Integrated Security=True;Encrypt=False")

    Public IsEdit As Boolean = False

    Public MaSVCanSua As String = ""    ' <-- Tên biến dùng chung cho update (giữ nguyên)

    Private Sub frmStudent\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

        Me.StartPosition = FormStartPosition.CenterScreen

        If IsEdit Then

            LoadStudentToControls()     ' nếu là sửa thì nạp dữ liệu lên textbox

            txtStudentID.Enabled = False ' khóa không cho sửa StudentID khi sửa

        End If

    End Sub

    ' Nạp thông tin sinh viên khi sửa

    Private Sub LoadStudentToControls()

        If String.IsNullOrEmpty(MaSVCanSua) Then Return

        Try

            ' 🟢 SỬA LỖI TÊN BẢNG VÀ TÊN CỘT

            ' Tên bảng: SinhVien -> Students

            ' Tên cột: MaSV, TenSV, Lop, SDT, DiaChi -> StudentID, StudentName, ClassID, Phone, Address

            Using cmd As New SqlCommand("SELECT StudentID, StudentName, ClassID, Phone, Address FROM Students WHERE StudentID = @MaSV", conn)

                cmd.Parameters.AddWithValue("@MaSV", MaSVCanSua)

                conn.Open()

                Using rdr As SqlDataReader = cmd.ExecuteReader()

                    If rdr.Read() Then

                        ' 🟢 Đọc dữ liệu từ tên cột chính xác trong SQL Server

                        txtClassID.Text = rdr("ClassID").ToString()

                        txtStudentID.Text = rdr("StudentID").ToString()

                        txtStudentName.Text = rdr("StudentName").ToString()

                        txtPhone.Text = rdr("Phone").ToString()

                        txtAddress.Text = rdr("Address").ToString()

                    End If

                End Using

                conn.Close()

            End Using

        Catch ex As Exception

            If conn.State = ConnectionState.Open Then conn.Close()

            MessageBox.Show("Lỗi nạp sinh viên: " & ex.Message)

        End Try

    End Sub

    ' Nút OK: thêm hoặc cập nhật

    Private Sub btnOK\_Click(sender As Object, e As EventArgs) Handles btnOK.Click

        Dim classID As String = txtClassID.Text.Trim()

        Dim studentID As String = txtStudentID.Text.Trim()

        Dim studentName As String = txtStudentName.Text.Trim()

        Dim phone As String = txtPhone.Text.Trim()

        Dim address As String = txtAddress.Text.Trim()

        If classID = "" Or studentID = "" Or studentName = "" Then

            MessageBox.Show("Vui lòng nhập đầy đủ thông tin!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning)

            Return

        End If

        Try

            If Not IsEdit Then

                ' Thêm mới

                ' 🟢 SỬA LỖI TÊN BẢNG VÀ TÊN CỘT TRONG INSERT

                Using cmd As New SqlCommand("INSERT INTO Students (StudentID, StudentName, ClassID, Phone, Address) VALUES (@MaSV, @TenSV, @Lop, @SDT, @DiaChi)", conn)

                    cmd.Parameters.AddWithValue("@MaSV", studentID)

                    cmd.Parameters.AddWithValue("@TenSV", studentName)

                    cmd.Parameters.AddWithValue("@Lop", classID)

                    cmd.Parameters.AddWithValue("@SDT", phone)

                    cmd.Parameters.AddWithValue("@DiaChi", address)

                    conn.Open()

                    cmd.ExecuteNonQuery()

                    conn.Close()

                End Using

                MessageBox.Show("Thêm sinh viên thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information)

            Else

                ' Cập nhật

                ' 🟢 SỬA LỖI TÊN BẢNG VÀ TÊN CỘT TRONG UPDATE

                Using cmd As New SqlCommand("UPDATE Students SET StudentName=@TenSV, ClassID=@Lop, Phone=@SDT, Address=@DiaChi WHERE StudentID=@MaSV", conn)

                    cmd.Parameters.AddWithValue("@MaSV", MaSVCanSua) ' <- dùng biến class-level

                    cmd.Parameters.AddWithValue("@TenSV", studentName)

                    cmd.Parameters.AddWithValue("@Lop", classID)

                    cmd.Parameters.AddWithValue("@SDT", phone)

                    cmd.Parameters.AddWithValue("@DiaChi", address)

                    conn.Open()

                    cmd.ExecuteNonQuery()

                    conn.Close()

                End Using

                MessageBox.Show("Cập nhật sinh viên thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information)

            End If

            Me.DialogResult = DialogResult.OK

            Me.Close()

        Catch ex As Exception

            If conn.State = ConnectionState.Open Then conn.Close()

            MessageBox.Show("Lỗi khi lưu dữ liệu: " & ex.Message)

        End Try

    End Sub

    Private Sub btnCancel\_Click(sender As Object, e As EventArgs) Handles btnCancel.Click

        Me.Close()

    End Sub

End Class

**CHƯƠNG 4: KẾT LUẬN**

Đồ án **“Quản lý sinh viên bằng VB.NET”** là một trải nghiệm quan trọng, giúp em củng cố kiến thức lý thuyết và nâng cao kỹ năng thực hành trong việc lập trình ứng dụng quản lý. Quá trình thực hiện đề tài đã cho em cái nhìn tổng quan về cách xây dựng một hệ thống quản lý thông tin sinh viên hoàn chỉnh, từ việc phân tích yêu cầu, thiết kế cơ sở dữ liệu, xây dựng giao diện cho đến lập trình các chức năng và kiểm thử.

Thông qua đồ án, em đã học hỏi và rèn luyện được nhiều kiến thức, kỹ năng như:

* **Phân tích và thiết kế hệ thống:** biết cách xác định các đối tượng cần quản lý như sinh viên, khoa, môn học; mô hình hóa bằng sơ đồ quan hệ dữ liệu; thiết kế bảng với khóa chính – khóa ngoại để đảm bảo tính toàn vẹn dữ liệu.
* **Lập trình trong VB.NET:** nắm được cách xây dựng giao diện thân thiện, dễ sử dụng cho người dùng; tổ chức mã nguồn theo hướng đối tượng; xử lý các chức năng cơ bản như thêm, sửa, xóa, tìm kiếm thông tin sinh viên, quản lý khoa và môn học.
* **Kết nối cơ sở dữ liệu:** hiểu rõ cách sử dụng SQL Server, viết truy vấn và kết nối ứng dụng VB.NET với cơ sở dữ liệu để quản lý thông tin sinh viên, khoa và kết quả học tập một cách hiệu quả.
* **Kỹ năng mềm:** nâng cao khả năng tự học, tra cứu tài liệu, phân tích và xử lý lỗi, tư duy logic và làm việc có kế hoạch trong quá trình lập trình.

Tuy nhiên, do kiến thức và kinh nghiệm thực tế còn hạn chế, sản phẩm của em vẫn còn một số thiếu sót. Cụ thể, giao diện chương trình chưa thực sự tối ưu, các chức năng nâng cao như thống kê điểm, in báo cáo, xuất dữ liệu hoặc bảo mật người dùng vẫn chưa được triển khai đầy đủ. Em nhận thức rằng để hoàn thiện hơn, em cần tiếp tục trau dồi kỹ năng, nghiên cứu sâu hơn về kỹ thuật lập trình, thiết kế phần mềm và tối ưu trải nghiệm người dùng.

Em rất mong nhận được những ý kiến đóng góp và sự chỉ bảo từ thầy/cô để có thể hoàn thiện và phát triển hệ thống một cách tốt hơn trong tương lai. Đây sẽ là hành trang quý báu giúp em tự tin hơn trong học tập cũng như trong công việc sau này.

**EM XIN TRÂN THÀNH CẢM ƠN!**