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

**KHOA CÔNG NGHỆ THÔNG TIN**

**ĐỒ ÁN VISUAL BASIC**

**QUẢN LÝ ĐIỂM SINH VIÊN**

**Sinh viên: Hoàng Mạnh Sỹ**

**Mã sinh viên: 19120418**

**Lớp: TH24.15**

**Hà Nội – Thứ 4, 20/10/2021**

**MỤC LỤC**

**Chương I: GIỚI THIỆU** 2

**Chương II: TỔNG QUAN**

1: Tổng quan về Visual Basic 4

1.1: Visual Basic là gì? 4

1.2: Mục đích sử dụng Visual Basic 5

2: Công cụ Visual Basic.NET(VB.NET) 5

2.1: Công cụ Visual Basic.NET là gì? 5

2.2: Hình ảnh giao diện VB.NET . 6

**Chương III: THIẾT KẾ CHƯƠNG TRÌNH**

1: Cấu trúc chương trình 8

1.1: Cấu trúc của một ứng dụng Visual Basic 8

1.2: Các bước xây dựng chương trình 8

2: Xây dựng chương trình 9

2.1: Phân tích thiết kế hệ thống 9

2.2: Tạo cơ sở dữ liệu 10

2.3: Thiết kế giao diện và liên kết với database 12

2.4: Lập trình code để giao diện tương tác truy vấn với cơ sở dữ liệu 14

**Chương IV: KẾT LUẬN**

1: Kết luận chung 20

2: Bản thân 23

**CHƯƠNG I**

GIỚI THIỆU

Tại Việt Nam, số lượng học sinh – sinh viên mỗi năm ngày càng nhiều, điều này cũng vô tình tạo khó khăn cho các trường học, trường đại học trong việc lưu trữ và quản lý kết quả. Trong khi đó việc ứng dụng công nghệ thông tin trong nhiều lĩnh vực ngày càng được khuyến khích. Vì vậy, ứng dụng công nghệ 4.0 vào việc quản lý, phần mềm quản lý điểm sinh viên làm trên ngôn ngữ Visual Basic .Net ra đời

Phần mềm quản lý điểm là công cụ cho phép nhập, lưu trữ và quản lý tất cả điểm số của sinh viên. Đồng thời các sinh viên cũng có thể truy cập phần mềm để xem điểm số của mình. Với phần mềm này, các giáo viên, phòng ban quản lý đào tạo sẽ tiết kiệm được nhiều thời gian trong việc tìm kiếm điểm thi của học viên cũng như là cải thiện sự hiệu quả trong việc quản lý điểm số. Tùy theo nhà cung cấp, dịch vụ phát triển phần mềm mà phần mềm có thể được sử dụng trên máy tính và trên điện thoại, tạo điều kiện cho cả nhà trường và sinh viên có thể sử dụng mọi lúc mọi nơi

**CHƯƠNG II**

**TỔNG QUAN VỀ VISUAL BASIC**

**1. TỔNG QUAN VỀ VISUAL BASIC**

**1.1 Visual Basic là gì?**

Visual Basic (viết tắt VB) là một ngôn ngữ lập trình hướng sự kiện (event-driven) và môi trường phát triển tích hợp (IDE) kết bó. Là sản phẩm của Microsoft, một thành phần phần của bộ Visual Studio. Chức năng: Là một ngôn ngữ lập trình dùng để xây dựng các ứng dụng chạy trên môi trường Windows.

Phần Visual (có nghĩa là hình ảnh trực quan) đề cập đến phương pháp được sử dụng để tạo giao diện đồ họa người dùng (Graphical User Interface hay viết tắt là GUI). Với những bộ phận hình ảnh (gọi là controls), bạn có thể sắp đặt vị trí và quyết định các đặc tính của chúng trên một khung màn hình, gọi là form (giống form trong Access). Nếu bạn đă từng sử dụng chương trình vẽ chẳng hạn như Paint, bạn đă có sẵn các kỹ năng cần thiết để tạo một GUI cho VB6.

Phần Basic đề cập đến ngôn ngữ BASIC (viết tắt của Beginner's All-purpose Symbolic Instruction Code), một ngôn ngữ lập trình đơn giản, dễ học, được tạo ra cho các khoa học gia (những người không có thời gian để học lập trình điện toán) dùng.

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

Visual Basic là một ngôn ngữ lập trình dùng để xây dựng các ứng dụng chạy trên môi trường Windows.

Visual Basic, con đường nhanh nhất và đơn giản nhất để tạo những ứng dụng cho Microsoft Windows. Bất kể bạn là một nhà chuyên nghiệp hay là một người mới lập trình Windows, Visual Basic cung cấp cho bạn một tập hợp các công cụ hoàn chỉnh để nhanh chóng phát triển các ứng dụng.

**2. Công cụ Visual Basic.NET(VB.NET)**

**2.1. Công cụ Visual Basic.NET là gì?**

Visual Basic.NET là một công cụ lập trình hỗ trợ ngôn ngữ lập trình Visual Basic.NET trong bộ công cụ lập trình Microsoft Visual Studio

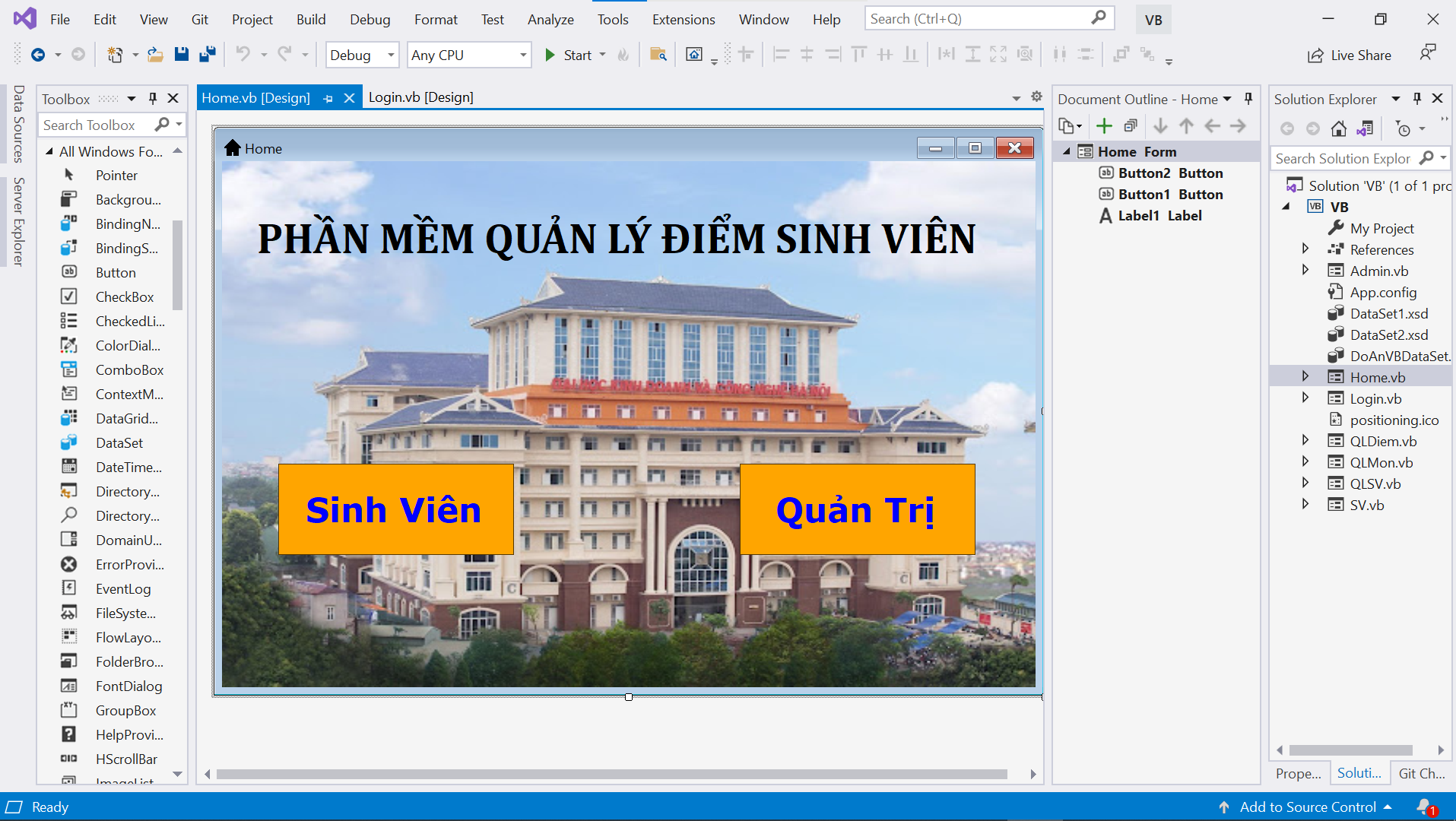
VB2005 được chạy trên nền .NET Framework 2 đến .NET Framework 4.8, khi đóng gói chương trình cũng cần bộ Framework này để chạy chương trình. Cũng như VB 6.0, VB2005 cũng là ngôn ngữ lập trình hướng đối tượng.

Mặc dù mục đích của chúng ta là tạo ra những ứng dụng nhỏ cho bản thân hay một nhóm, một hệ thống các công ty lớn, hoặc thậm chí phân phối những ứng dụng ra toàn cầu qua Internet. Visual Basic là cung cụ mà bạn cần:

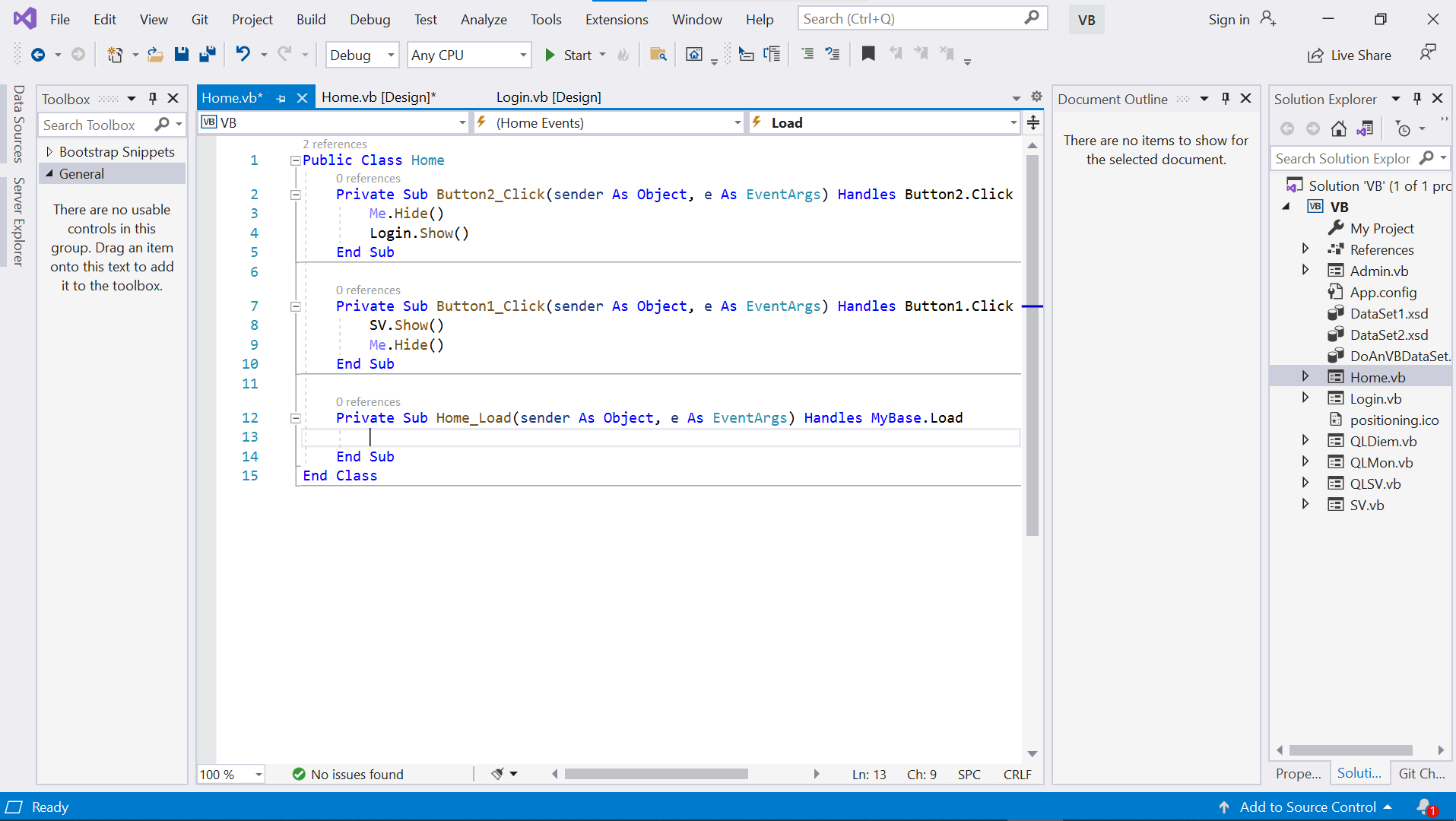
* Những chức năng truy xuất dữ liệu cho phép ta tạo ra những cơ sở dữ liệu, những ứng dụng front-end, và những thành phần phạm vi server-side cho hầu hết các dạng thức cơ sở dữ liệu phổ biến, bao gồm Microsoft SQL Server và những cơ sở dữ liệu mức enterprise khác.
* Những kỹ thuật ActiveX cho phép ta dùng những chức năng được cung cấp từ những ứng dụng khác, như là chương trình xử lý văn bản Microsoft Word, bảng tính Microsoft Excel và những ứng dụng Windows khác.
* Khả năng Internet làm cho nó dễ dàng cung cấp cho việc thêm vào những tài liệu và ứng dụng qua Internet hoặc intranet từ bên trong ứng dụng của bạn, hoặc tạo những ứng dụng Internet server.
* Ưng dụng của bạn kết thúc là một file .exe thật sự. Nó dùng một máy ảo Visual Basic để bạn tự do phân phối ứng dụng.

**2.2. Hình ảnh giao diện VB.NET trong phần mềm Visual studio 2019.**

Người dùng có thể tải chương trình Visual Studio tại địa chỉ: <https://visualstudio.microsoft.com/fr/downloads/>. Sau khi tải xong, tiến hành cài đặt chương trình vào máy tính theo cách thông thường. Một số hình ảnh của chương trình Visual Studio:



Hình 1: Giao diện thiết kế Form của VB.NET



Hình 2: Giao diện code của VB.NET

**CHƯƠNG III**

**THIẾT KẾ CHƯƠNG TRÌNH**

**1. Cấu trúc chương trình**

**1.1. Cấu trúc của một ứng dụng Visual Basic**

Một ứng dụng thật ra là một tập hợp các chỉ dẫn trực tiếp đến máy tính để thi hành một hay nhiều tác vụ. Cấu trúc của một ứng dụng là phương pháp trong đó các chỉ dẫn được tổ chức, đó là nơi chỉ dẫn được lưu giữ và thi hành những chỉ dẫn trong một trình tự nhất định.

Vì một ứng dụng Visual Basic, trên cơ bản là những đối tượng, cấu trúc mã đóng để tượng trưng cho những mô hình vật lý trên màn hình. Bằng việc định nghĩa, những đối tượng chứa mã và dữ liệu. Form, cái mà chúng ta nhìn thấy trên màn hình là tượng trưng cho những thuộc tính, quy định cách xuất hiện và cách cư xử. Cho mỗi form trong một ứng dụng, có một quan hệ module form (với tên file mở rộng là .frm) dùng để chứa đựng mã của nó.

Mỗi module chứa những thủ tục sự kiện – những đoạn mã, nơi đặt những chỉ dẫn, cái sẽ được thi hành trong việc đáp ứng những sự kiện chỉ định. Form có thể chứa những điều khiển. Tương ứng với mỗi điều khiển trên form, có một tập hợp những thủ tục sự kiện trong module form đó.

Mã không chỉ quan hệ với một form chỉ định hay điều khiển có thể được đặt trong một loại module khác, một module chuẩn (.BAS). Một thủ tục được dùng để đáp ứng những sự kiện trong những đối tượng khác nhau phải được đặt trong cùng một chuẩn, thay vì tạo những bản sao mã trong những thủ tục sự kiện cho mỗi đối tượng. Một lớp module (.cls) được dùng để tạo những đối tượng, cái mà có thể được gọi từ những thủ tục bên trong ứng dụng của bạn. Trong khi một module chuẩn chỉ chứa mã, một lớp module chứa đựng cả mã và dữ liệu. Ta có thể nghĩ nó như một điều khiển.

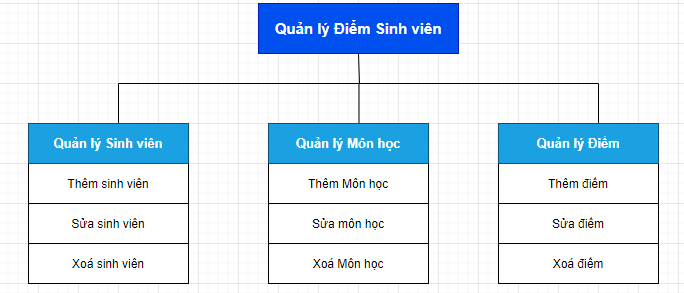
**1.2. Các bước xây dựng chương trình**

Phân tích và thiết kế hệ thống → Tạo Cơ sở dữ liệu → Thiết kế Giao diện(Form) → Lập trình code để giao diện tương tác truy vấn với cơ sở dữ liệu → Hoàn Thành

**2. Xây dựng chương trình**

**2.1. Phân tích thiết kế hệ thống**

Biểu đồ phân cấp chức năng:



**2.2 Tạo cơ sở dữ liệu**

Tạo database bằng querry:

create database DoAnVB

go

use DoAnVB

-Code:

create table Diem(

MaSV varchar(10) not null,

MaMH varchar(50)not null,

DiemT4 float(,

DiemChu varchar(50),

DiemT10 float,

constraint pr\_diem primary key(MaSV,MaMH)

)

Làm tương tự với các bảng khác

Thêm khóa ngoại cho các bảng:

-Code:

GO

ALTER TABLE [dbo].[Diem] WITH CHECK ADD CONSTRAINT [Diem\_MaSV\_fk] FOREIGN KEY([MaSV])

REFERENCES [dbo].[DMSV] ([MaSV])

GO

ALTER TABLE [dbo].[Diem] CHECK CONSTRAINT [Diem\_MaSV\_fk]

GO

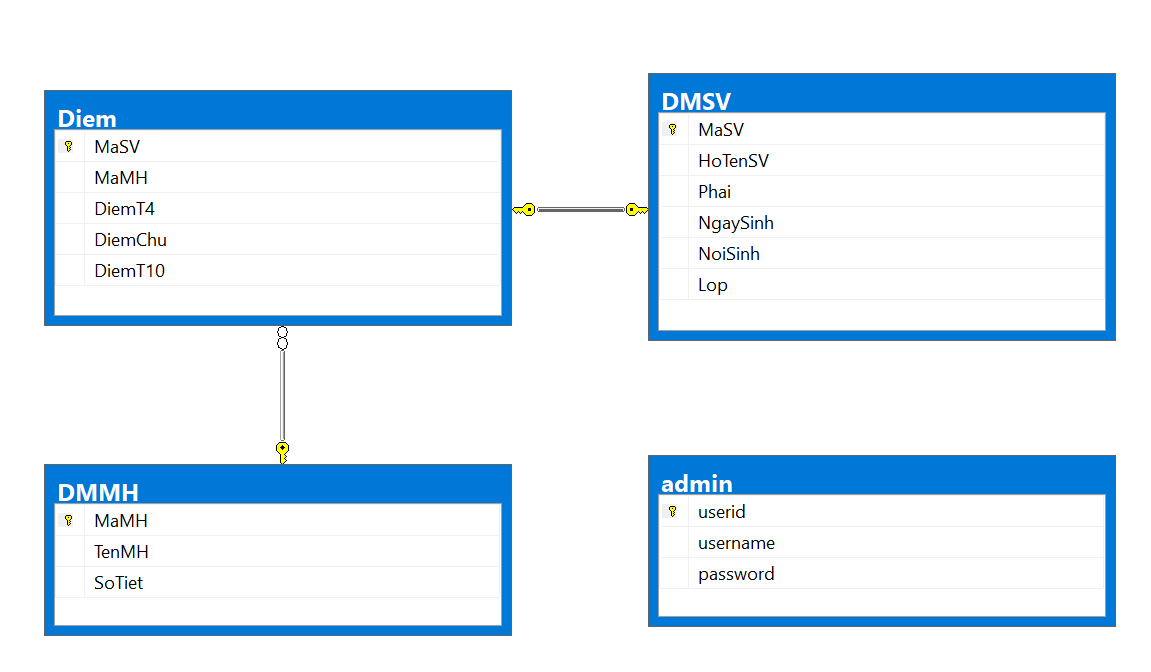
ALTER TABLE [dbo].[Diem] WITH CHECK ADD CONSTRAINT [DMMH\_MaMH\_fk] FOREIGN KEY([MaMH])

REFERENCES [dbo].[DMMH] ([MaMH])

GO

ALTER TABLE [dbo].[Diem] CHECK CONSTRAINT [DMMH\_MaMH\_fk]

GO

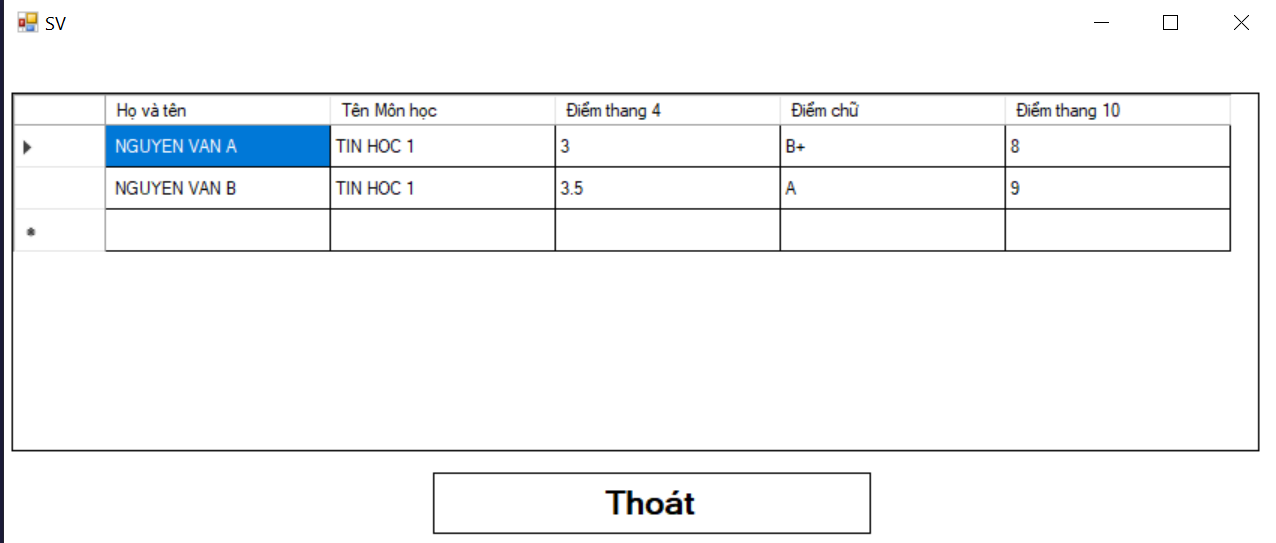
Lược đồ quan hệ:

**2.3. Thiết kế giao diện và liên kết với database**

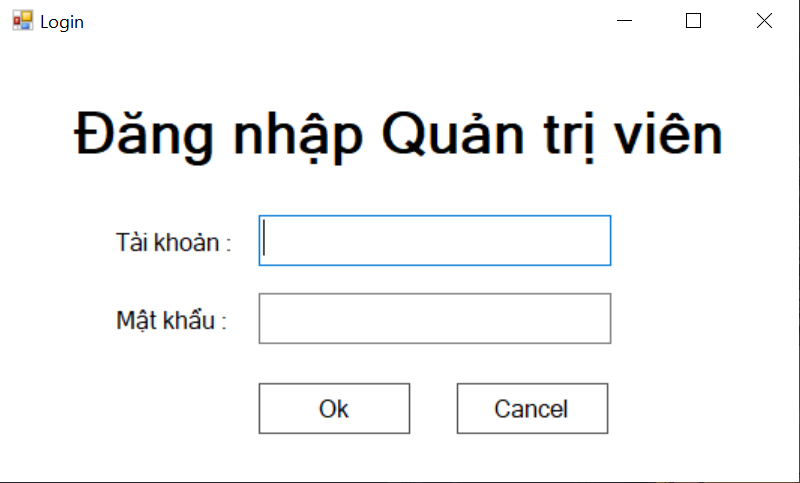
Giao diện Trang chủ



Giao diện Xem điểm



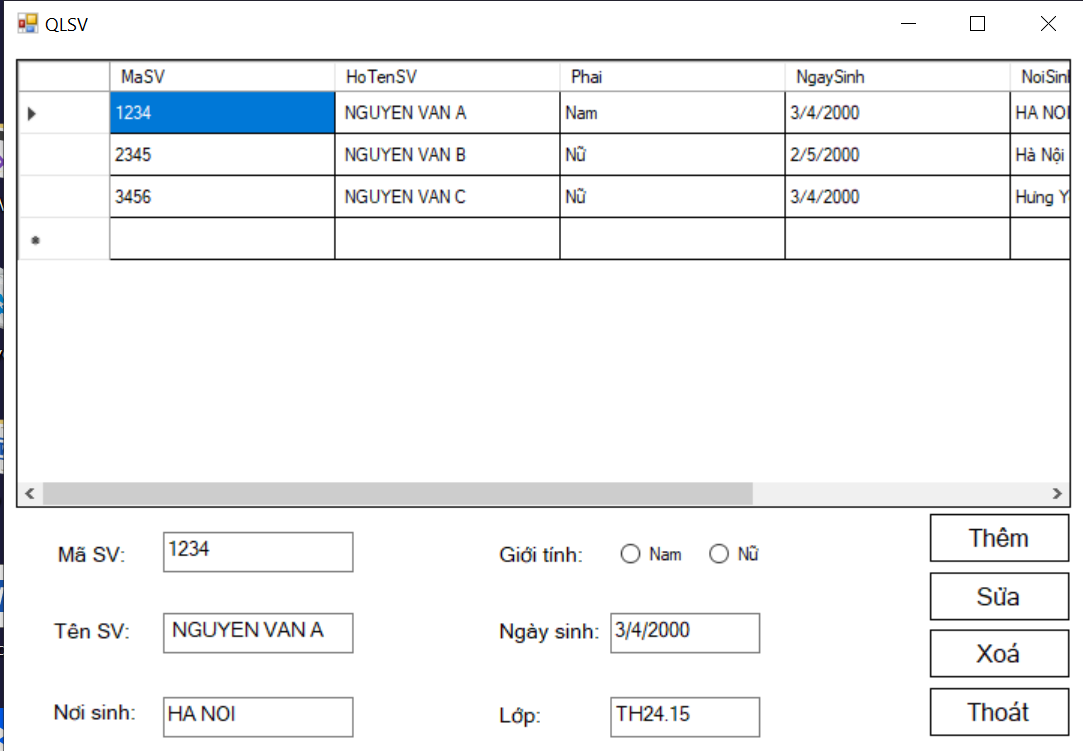
Giao diện Đăng nhập



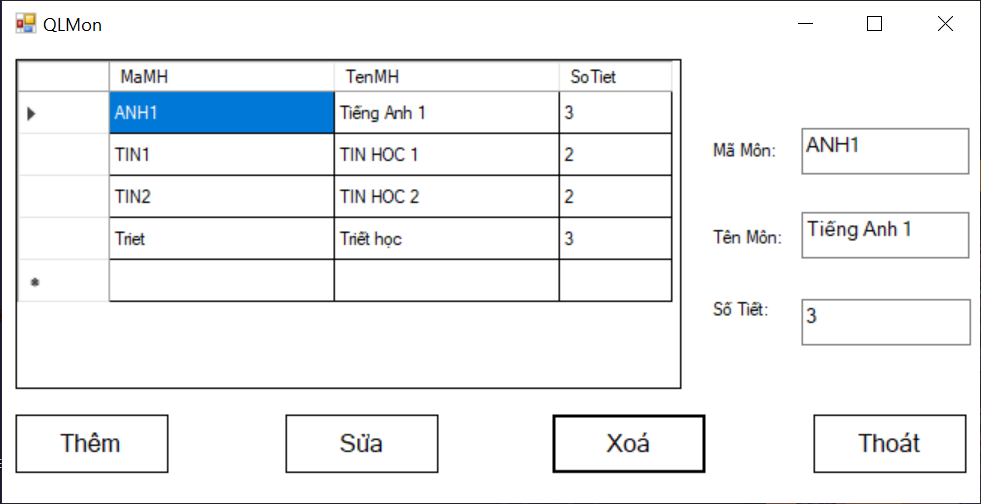
Giao diện trang Quản trị viên



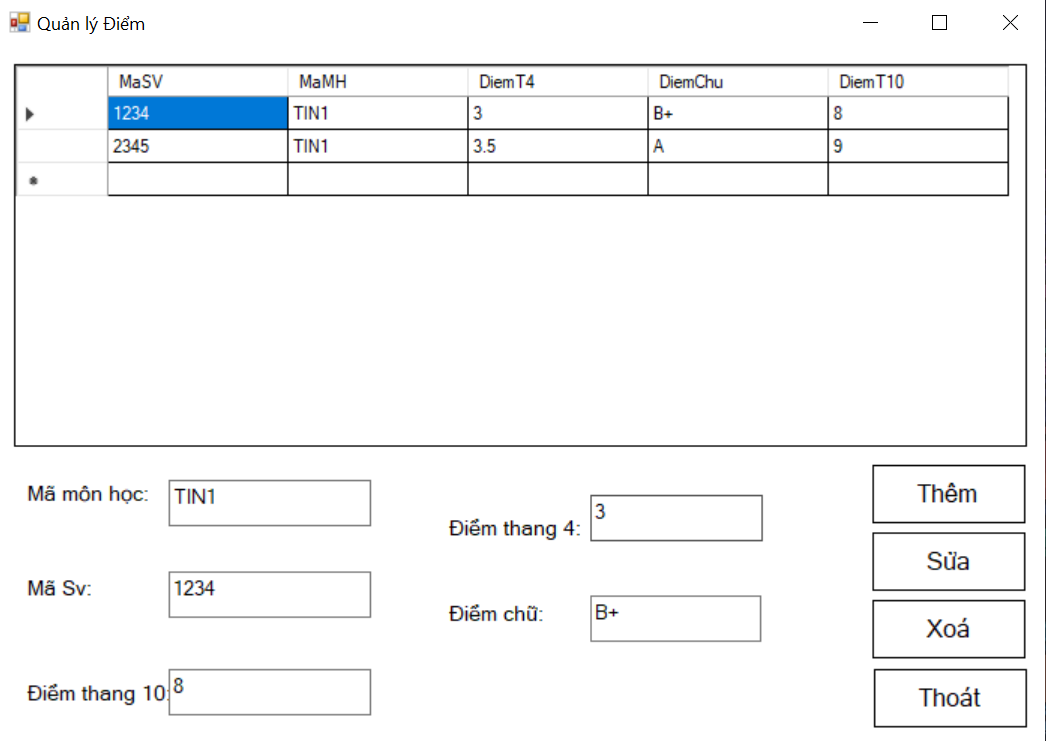
Giao diện Quản lý sinh viên



Giao diẹn quản lý môn học



Giao diện quản lý điểm



**2.4 Lập trình code để giao diện tương tác truy vấn với cơ sở dữ liệu**

-Code chuyển đổi giữa các form thành viên từ form Admin

Public Class Admin

Private Sub Button1\_Click(sender As Object, e As EventArgs) Handles Button1.Click

QLSV.Show()

Me.Hide()

End Sub

Private Sub Button2\_Click(sender As Object, e As EventArgs) Handles Button2.Click

QLMon.Show()

Me.Hide()

End Sub

Private Sub Button3\_Click(sender As Object, e As EventArgs) Handles Button3.Click

QLDiem.Show()

Me.Hide()

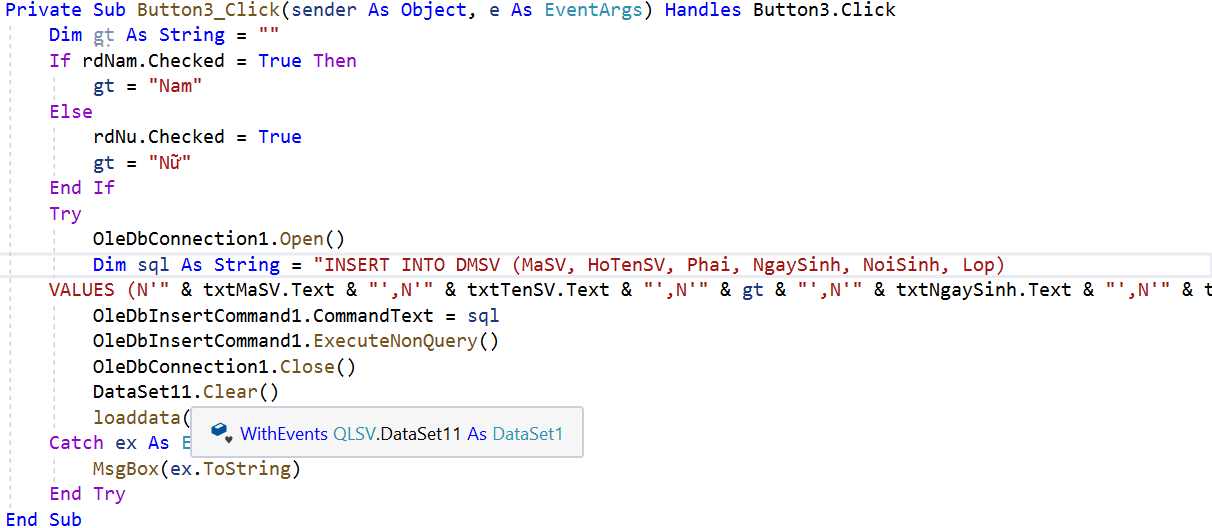
End Sub

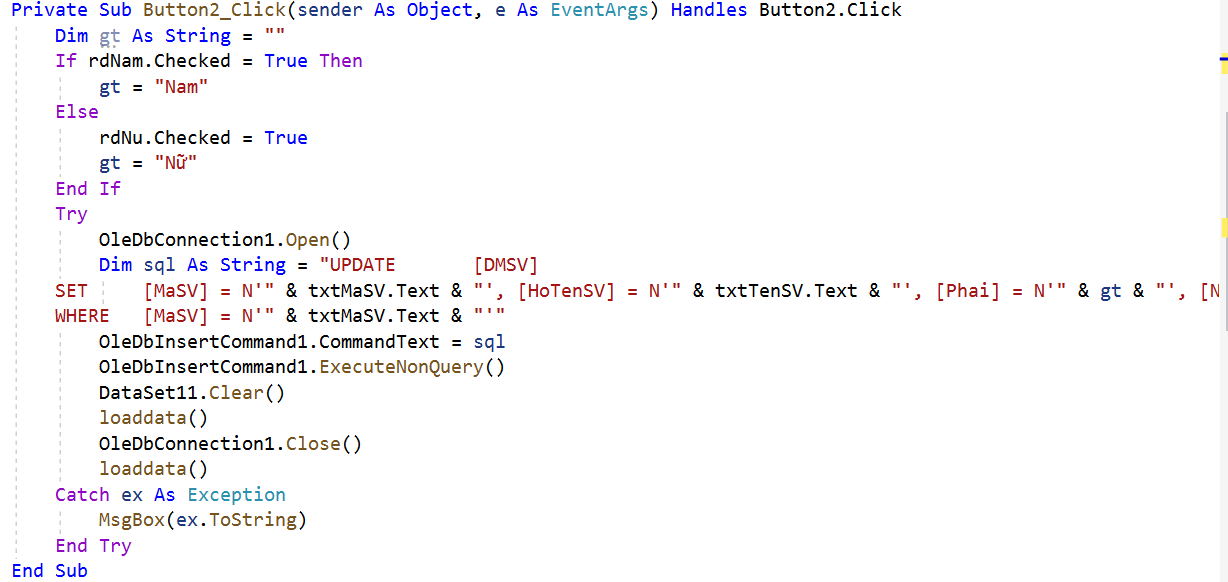
End ClassCode tự động đổ dữ liệu vào các DataGrid khi khởi động chương trình:

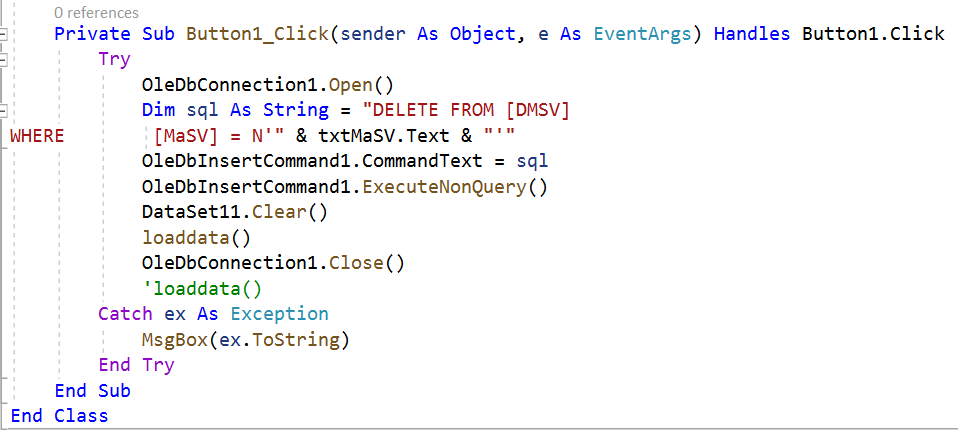
Private Sub QLSV\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

Me.DMSVTableAdapter.Fill(Me.DataSet11.DMSV)

End Sub

- Code thêm một bản ghi từ form QLSV lên database: 

-Code sửa một bản ghi từ form QLSV lên database:

-Code xoá một bản ghi từ form QLSV lên database:

**4. Các kiến thức sử dụng trong bài**

Cách khai báo biến:

Dim <Tên biến> As <Kiểu biến>

Các kiểu biến:

Tên kiểu Kích thước Khoảng gía trị

- Byte 1 byte 0 đến 255 (tức có thể gán cho biến các giá trị nhỏ

nhất là 0 và lớn nhất là 255)

- Integer 2 byte -32768 đến 32767

Long 4 byte -2.147.483.648 đến 2.147.483.647

- Single 4 byte -3,402823E38 đến –1,401298E-45 (các gía trị âm)

1,401298E-45 đến 3,402823E38 (các gía trị dương)

-Double 8 byte -1,79769E308 đến –4,94065E-324 (giá trị âm)

4,94065E-324 đến 1,79769E308 (giá trị dương)

- Currency 8 byte -922337203685477,5808 đến

922337203685477,5807

- Boolean 2 byte True tới False

- Date 2 byte 1 tháng 1 năm 100 đến 31 tháng 12 năm 9999 ;

Thời gian từ 0:00:00 đến 23:59:59:

- String 1 byte cho có thể lên đến 231 ký tự

mỗi ký tự

- Variant 16 byte + Null, Error, bất kỳ kiểu số nào có giá trị trong khoảng

1 byte cho Double hay bất kỳ nội dung text nào

mỗi ký tự

- Ký hiệu Exx phía sau số có nghĩa là nhân với 10xx.

- Kiểu số nguyên dương (không chấp nhận số âm) gồm kiểu Byte.

- Kiểu số nguyên (chấp nhận cả số âm nhưng không chấp nhận phần lẻ thập phân) gồm các kiểu :Integer, Long.

- Kiểu số thực gồm Single, Double, Currency.

- Kiểu Boolean gọi là kiểu luận lý, nó chỉ chấp nhận hai giá trị True là đúng và False là sai.

- Kiểu String dùng để chứa các giá trị chuỗi. Một chuỗi ký tự có thể có nhiều ký tự. - - Khi viết một giá trị chuỗi, ta phải bao hai đầu nó bằng dấu nháy kép.

- Kiểu ngày tháng (Date) để chứa giá trị thời gian. Khi viết một giá trị kiểu Date, ta có thể viết theo bất cứ kiểu ghi giờ nào bao hai đầu bằng dấu #.

Lệnh đóng chương trình:

Me.Close()

Lệnh thu nhỏ giao diện chương trình:

WindowState = WindowState.Minimezed

Cấu trúc tuyển If:

Cú pháp 1 :

If <biểu thức luận lý> Then

… ‘Nếu biểu thức luận lý trên là True

… ‘thì thực hiện đoạn lệnh này

End If

Cú pháp 2 :

If <biểu thức luận lý> Then

… ‘Nếu biểu thức luận lý trên là True

… ‘thì thực hiện đoạn lệnh này

Else

… ‘Ngược lại nếu biểu thức luận lý là False

… ‘thì thực hiện đoạn lệnh này.

End If

Lệnh hiển thị và bỏ hiển thị thành phần trong giao diện:

Hiển thị thành phần:

<Tên thành phần>.Visible = true

Bỏ hiển thị thành phần:

<Tên thành phần>.Visible = False

Câu lệnh mở kết nối với database:

<Tên kết nối>.Open()

Câu lệnh đóng kết nối database:

<Tên kết nối>.Close()

Câu lệnh bắt lỗi **Try**:

**Try**

<câu\_lệnh1>

**catch**

<câu\_lệnh2>

**End try**

**CHƯƠNG IV**

**KẾT LUẬN**

**1. KẾT LUẬN CHUNG**

**Tạo giao diện người sử dụng**

Giao diện người sử dụng có lẽ là thành phần quan trọng nhất của một ứng dụng. Đối với người sử dụng, giao diện chính là ứng dụng ; họ không cần chú ý đến thành phần mã thực thi bên dưới. Ứng dụng của chúng ta có thể phổ biến được hay không phụ thuộc vào giao diện.

**Sử dụng những điều khiển chuẩn của Visual Basic**

Ta dùng những điều khiển để lấy thông tin mà người sử dụng nhập vào, và để hiển thị kết xuất. Những điều khiển mà ta có thể dùng trong ứng dụng bao gồm hộp văn bản, nút lệnh, và hộp danh sách,…. Những điều khiển khác cho ta truy xuất những ứng dụng khác, xử lý dữ liệu của nó như là một thành phần mã trong ứng dụng của bạn.

**Lập trình với những đối tượng.**

Những đối tượng là thành phần chủ yếu để lập tình Visual Basic. Những đối tượng có thể là form, những điều khiển hay cơ sở dữ liệu.

**Lập trình với phần hợp thành**

Chúng ta đôi khi cần sử dụng khả năng tính toán của Microsoft Excel trong ứng dụng Visual Basic, hay định dạng một tài liệu sử dụng thanh công cụ định dạng của Microsoft Word, hoặc lưu trữ và xử lý dữ liệu dùng Microsoft Jet… Tất cả những điều này có thể thực hiện được bằng cách xây dựng những ứng dụng của chúng ta sử dụng những thành phần ActiveX. Thêm vào đó, Visual Basic có thể giúp chúng ta tạo ra những điều khiển ActiveX riêng.

**Đáp ứng những sự kiện chuột và bàn phím**

Những ứng dụng Visual Basic có thể đáp ứng một lượng lớn sự kiện chuột và bàn phím. Ví dụ form, hộp ảnh, và những điều khiển ảnh có thể phát hiện vị trí con trỏ chuột, có thể quyết định phím trái hay phím phải được nhấn, và có thể đáp ứng những tổ hợp của phím chuột với phím Shift, Ctrl, hay Alt. Sử dụng những điều khiển phím, ta có thể lập trình những điều khiển và form để đáp ứng các hành động phím hoặc phiên dịch và xử lý mã Ascii của ký tự.

Thêm vào đó, những ứng dụng Visual Basic có thể hỗ trợ sự kiện rê và thả cũng như tính năng rê và thả OLE.

**Làm việc với văn bản và đồ họa.**

Visual Basic cung cấp khả năng đồ họa và văn bản phức tạp trong ứng dụng. Những thuộc tính văn bản có thể giúp ta nhấn mạnh các khái niệm quan trọng và các chi tiết cần quan tâm. Thêm vào đó, Visual Basic cung cấp khả năng đồ họa cho phép ta linh động trong thiết kế, bao hàm các hình ảnh động bằng cách hiển thị một loạt các hình ảnh liên tiếp nhau.

**Gỡ rối mã và quản lý lỗi**

Đôi khi có những lỗi xảy ra bên trong mã của ứng dụng. Những lỗi nghiêm trọng có thể là nguyên nhân một ứng dụng không đáp ứng lệnh, thông thường yêu cầu người sử dụng khởi động lại ứng dụng, và không lưu lại những gì ta đã làm. Quá trình tìm ra và sửa lỗi gọi là gỡ rối. Visual Basic cung cấp nhiều công cụ giúp chúng ta phân tích ứng dụng làm việc như thế nào. Những công cụ gỡ rối đặt biệt hữu ích trong việc tìm ra nguồn gốc lỗi, nhưng chúng ta cũng có thể dùng những công cụ này để kiểm tra chương trình hoặc tìm hiểu những ứng dụng khác nhau làm việc như thế nào.

**Xử lý ổ đĩa, thư mục và file**

Khi lập trình trong Windows, nó rất quan trọng để có khả năng thêm, di chuyển, tạo mới hoặc xóa những thư mục và file, lấy thông tin về và xử lý ổ đĩa. Visual Basic cho phép chúng ta xử lý ổ đĩa, thư mục và file bằng hai phương pháp : qua những phương htức cũ như là điều lệnh Open, Write#, và qua một tập hợp các công cụ mới như FSO (File System Object)

**Thiết kế cho việc thi hành và tính tương thích**

Visual Basic chia xẻ hầu hết những tính năng ngôn ngữ trong Visual Basic cho những ứng dụng, bao gồm trong Microsoft Office và nhiều ứng dụng khác. Visual Basic, VBScript, một ngôn ngữ script Internet, đều là tập hợp con của ngôn ngữ Visual Basic.

**Phân phối những ứng dụng**

Sau khi tạo một ứng dụng Visual Basic, ta có thể tự do phân phối bất kỳ ứng dụng nào đã tạo bằng Visual Basic đến bất cứ ai dùng Microsoft Windows. Ta có thể phân phối ứng dụng trên đĩa, trên CD, qua mạng, trên intranet hoặc Internet.

**2. BẢN THÂN**

Qua bộ môn lập trình Visual Basic này, em đã có thể tạo cho riêng mình một sản phẩm thực sự thông qua lập trình, qua đó em cũng có thể dễ dàng hiểu được mối quan hệ giữa các cấu trúc, câu lệnh lập trình và hành vi của các đối tượng thông qua các hình ảnh giao diện. Bằng cách này, em đã hiểu rõ và rút ra được nhiều kinh nghiệm với các câu lệnh, cấu trúc, chương trình (liên quan đến lập trình trực quan) mà em sẽ được giảng dạy sau này.