**TRƯỜNG ĐẠI HỌC THỦ DẦU MỘT**

**KHOA KỸ THUẬT - CÔNG NGHỆ**



**BÁO CÁO GIỮA KÌ MÔN HỌC  
CÔNG NGHỆ XML VÀ ỨNG DỤNG**

**ĐỀ TÀI: CLOUD SERVICE MODELS**

**AND**

**OPENNEBULA**

**Giảng viên giảng dạy: Th.S Nguyễn Đình Thọ**

**Nhóm thực hiện: Nhóm 1**

***Bình Dương, tháng 11 năm 2017***

MỤC LỤC

[PHẦN 1: CLOUD COMPUTING 1](#_Toc501923568)

[Chương I: TỔNG QUAN VỀ CLOUD COMPUTING 1](#_Toc501923569)

[1. Đặt vấn đề 1](#_Toc501923570)

[2. Định nghĩa 2](#_Toc501923571)

[3. Mô hình Cloud Computing 3](#_Toc501923572)

[4. Các giải pháp về Cloud computing 5](#_Toc501923573)

[CHƯƠNG II: THỰC TRẠNG VÀ CÁC MÔ HÌNH CỦA CLOUD COMPUTING 7](#_Toc501923574)

[1. Hiện thực Cloud computing và các mô hình 7](#_Toc501923575)

[1.1.1 Infrastructure as a service (IaaS dịch vụ hạ tầng) 7](#_Toc501923576)

[1.1.2 Platform-as-a-Service (PaaS - Dịch vụ nền tảng) 9](#_Toc501923577)

[1.1.3 Software-as-a-Service (SaaS - Dịch vụ phần mềm): 11](#_Toc501923578)

[2. Một số ví dụ về cách tính chi phí trong Cloud 13](#_Toc501923579)

[3. Mô hình Cloud computing theo mô hình triển khai 13](#_Toc501923580)

[PHẦN 2: OPENEBULLA 14](#_Toc501923581)

[CHƯƠNG I: TỔNG QUAN VỀ OPENNEBULLA 14](#_Toc501923582)

[1. OpenNebula 14](#_Toc501923583)

[2. Các nền tảng OpenNebula 15](#_Toc501923584)

[3. Các thành phần chính của OpenNebula 15](#_Toc501923585)

[CHƯƠNG II. XÂY DỰNG OPENNEBULA VỚI GIAO DIỆN SUNSTONE 19](#_Toc501923586)

[1. Cài đặt Opennebula trên hệ điều hành Ubuntu 19](#_Toc501923587)

[2. Triển khai các dịch vụ trên Opennebula 22](#_Toc501923588)

[2.1 Cấu hình KVM Node 22](#_Toc501923589)

[2.2 Thêm Host vào OpenNebula 22](#_Toc501923590)

[2.3 Thiết lập mạng trên Opennebula 23](#_Toc501923591)

[2.4 Tạo Images máy ảo trong Opennebula 24](#_Toc501923592)

[2.5 Tạo VM Templates 25](#_Toc501923593)

[2.6 Instances máy ảo 26](#_Toc501923594)

[PHẦN 3: TỔNG KẾT 26](#_Toc501923595)

[1. Xu hướng phát triển của điện toán đám mây (cloud computing) 27](#_Toc501923596)

[2. Ưu điểm của điện toán đám mây 27](#_Toc501923597)

[3. Nhược điểm của điện toán đám mây 27](#_Toc501923598)

[TÀI LIỆU THAM KHẢO 29](#_Toc501923599)

**PHẦN 1: YÊU CẦU BÁO CÁO**

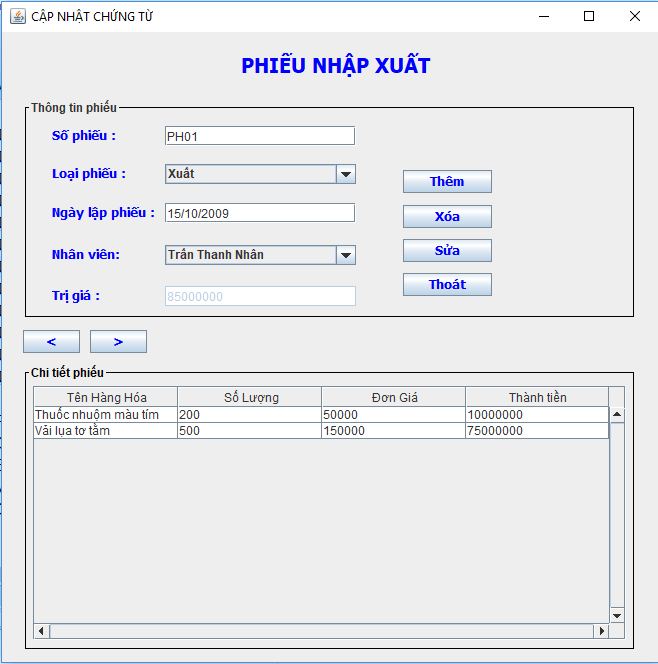
**Yêu cầu đề bài**

**XÂY DỰNG FORM THỰC HIỆN CÁC THAO TÁC VỚI XML**

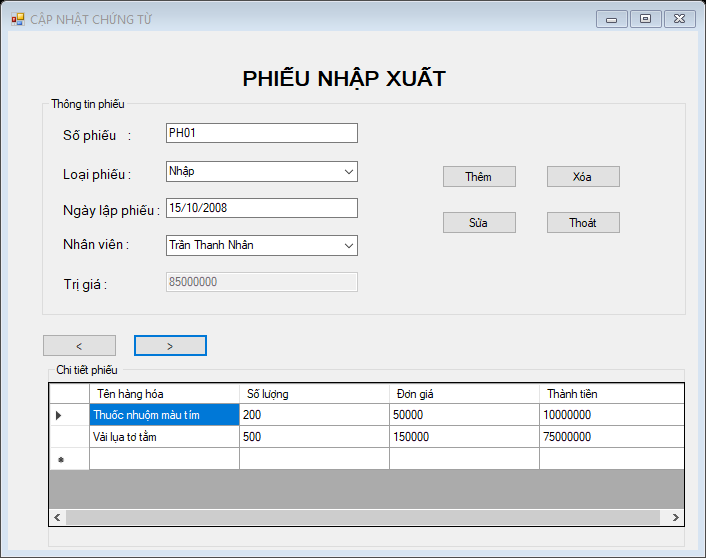
Yêu cầu cụ thể như sau:

* Load  dữ liệu từ file XML và hiển thị trên Frame
* Tạo các Button xử lý các sự kiện như THÊM, SỬA, XÓA, TÌM KIẾM…. các phần tử trong XML,THOÁT…. (Không giới hạn các thao tác)
* Có thể dùng mô hình DOM hoặc SAX để parser dữ liệu.
* Ngôn ngữ lập Trình là tùy ý ( Java hoặc C#....v…v.)
* Form báo cáo file word giống như môn Điện toán đám mây. Gửi file nén kèm theo Code. Gửi lên E-learning khi có yêu cầu .
* Lưu ý : Nội dung Không được trùng nhau
* Ví dụ : như form dước đây (chỉ mang tính tham khảo)

**Jframe- Java**

* 

**Microsoft Visual Studio - CScharp**

* 
* Tài liệu tham khảo:
* <https://www.youtube.com>
* <https://www.youtube.com/watch?v=w09xnvkl6UQ>
* <https://www.youtube.com/watch?v=cVdMFwKu998>

**PHẦN 2: XÂY DỰNG FILE XML**

Tiến hành thiết kế và tạo ra file dữ liệu sinh viên bằng XML cho chương trình.



File sinhvien.xml

<?xml version="1.0" standalone="yes"?>

<documentelement>

<sinhvien>

<masinhvien>1</masinhvien>

<tensinhvien>Nguyen Van teo</tensinhvien>

<lop>D15PM02</lop>

<noisinh>Binh Duong</noisinh>

</sinhvien>

<sinhvien>

<masinhvien>2</masinhvien>

<tensinhvien>Nguyen Van Khoi</tensinhvien>

<lop>D15PM02</lop>

<noisinh>Binh Duong</noisinh>

</sinhvien>

<sinhvien>

<masinhvien>3</masinhvien>

<tensinhvien>Nguyen Minh Thien</tensinhvien>

<lop>D15PM02</lop>

<noisinh>Binh Duong</noisinh>

</sinhvien>

<sinhvien>

<masinhvien>4</masinhvien>

<tensinhvien>Nguyen Hoang Duy</tensinhvien>

<lop>D15PM02</lop>

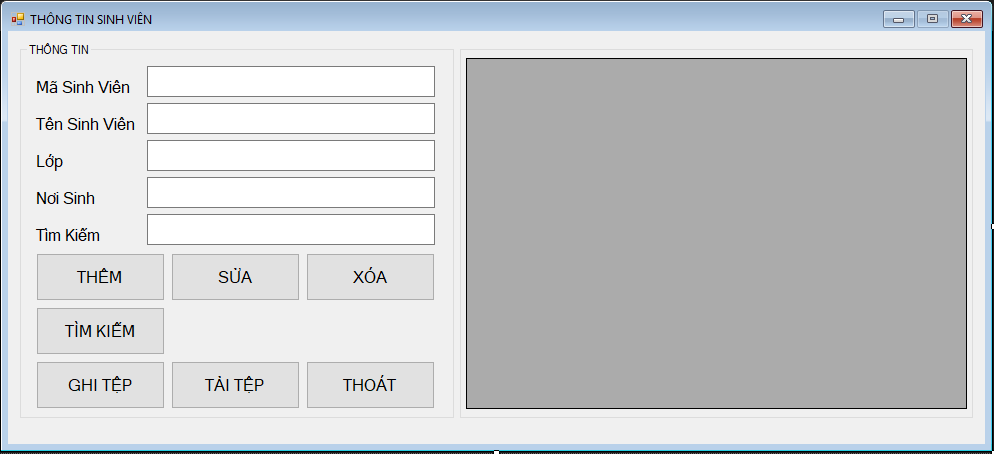
<noisinh>Binh Duong</noisinh>

</sinhvien>

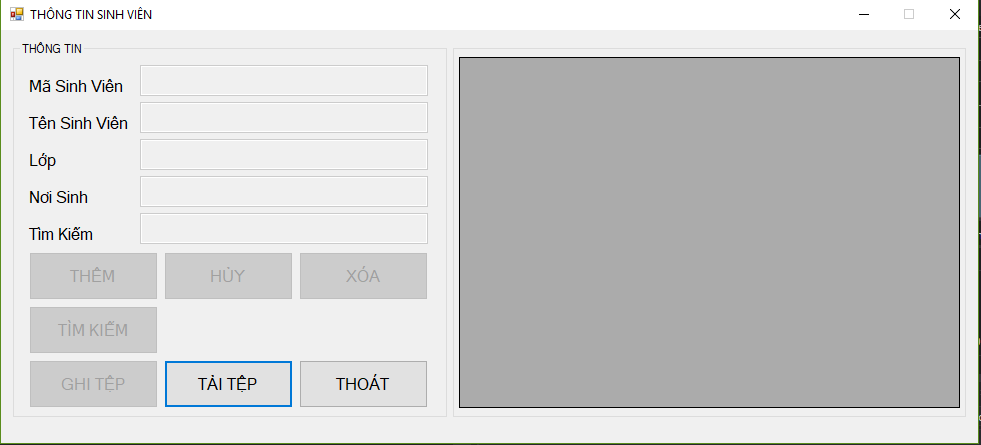
</documentelement>

**PHẦN 3: THIẾT KẾ GIAO DIỆN FORM**

Thiết kế giao diện chính form thông tin sinh viên

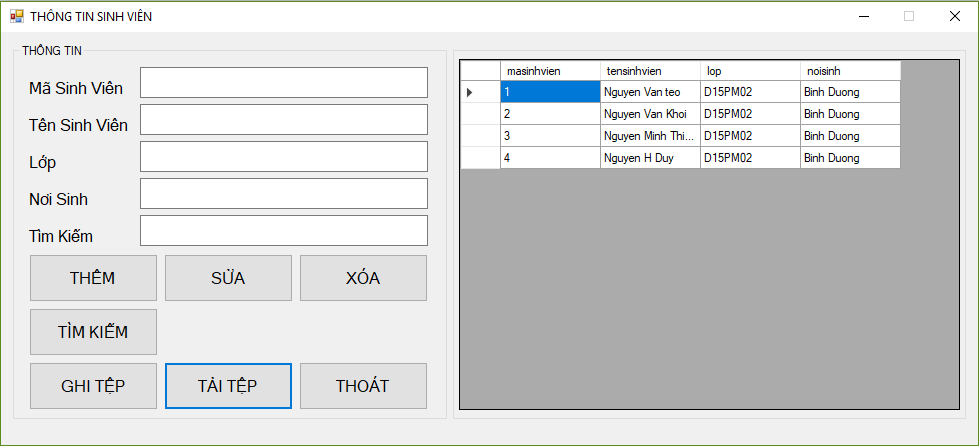


Màn hình chính lúc khởi động



Chọn chức năng “” để load file “sinhvien.xml” đã tạo

Màn hình chính sau khi load file “sinhvien.xml”

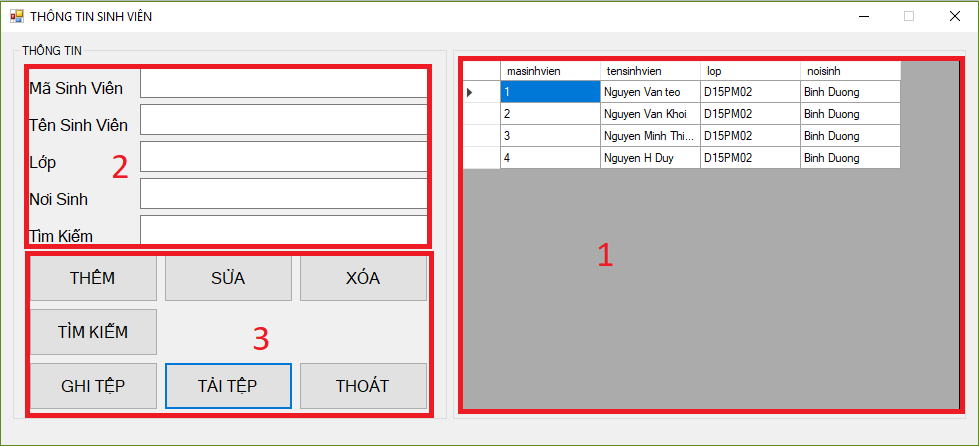


Các nhóm chức năng:

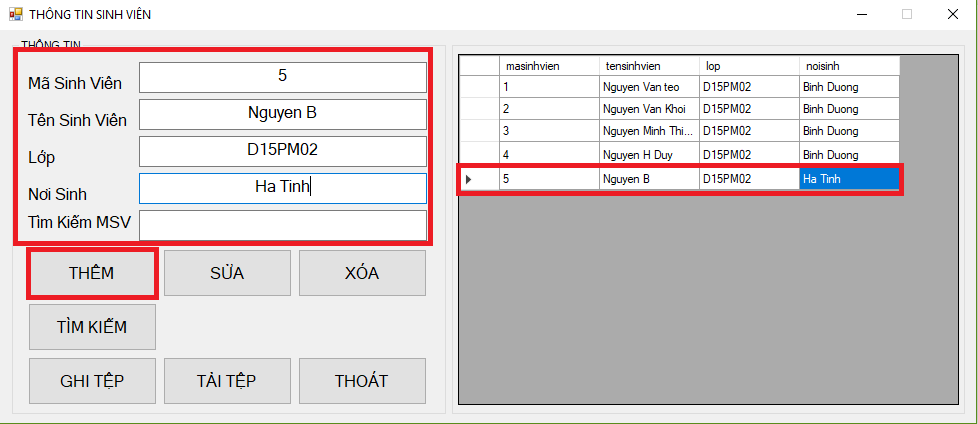
1/ file xml được load thành công

2/ Các ô textbox để chỉnh sửa dữ liệu

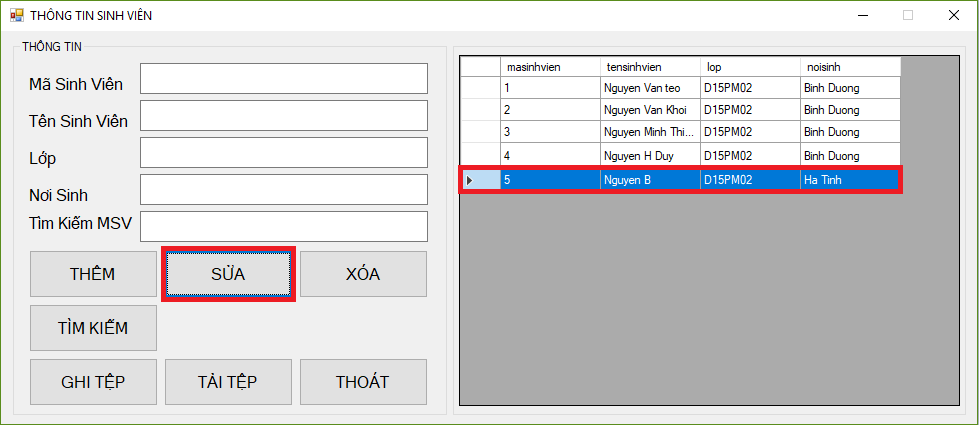
3/ Các nút chức năng

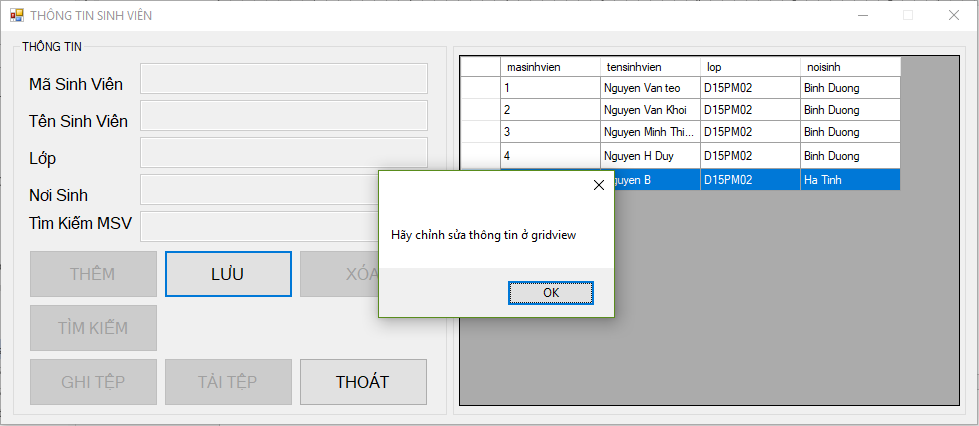


Điền đầy đủ thông tin vào các ô và nhấn nút thêm để thêm một sinh viên mới

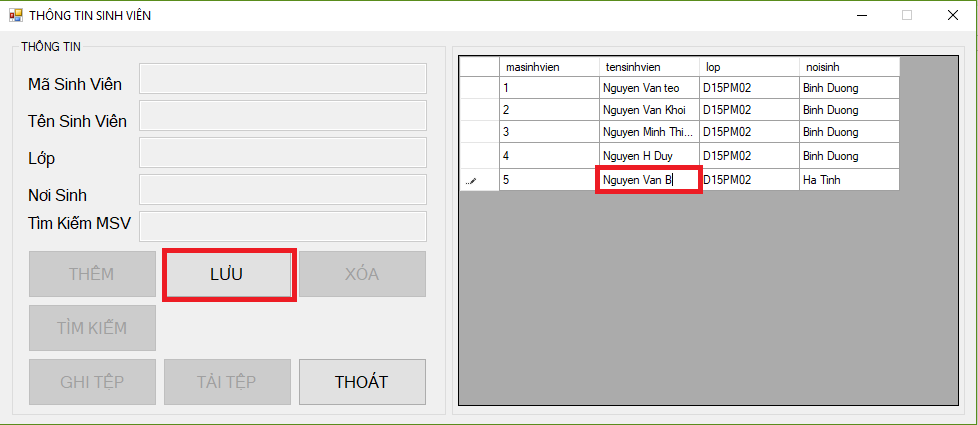


Chọn một dòng trong bảng và chọn “” để tiến hành chỉnh sửa một đối tượng sinh viên

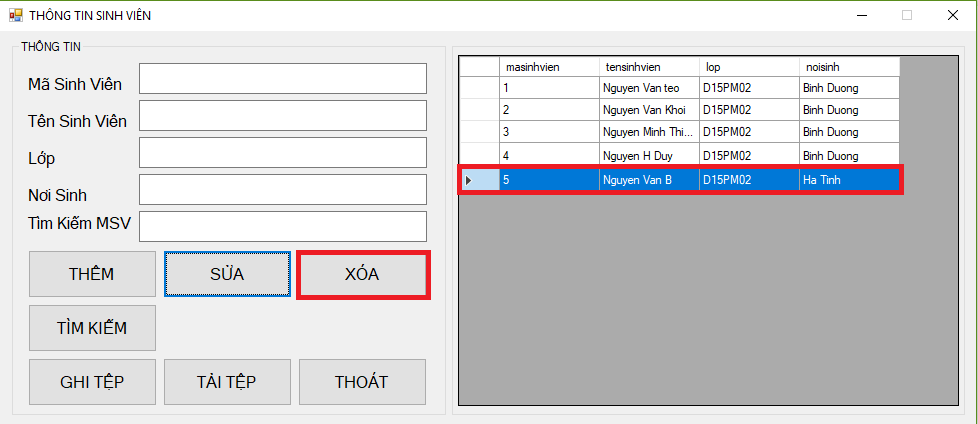


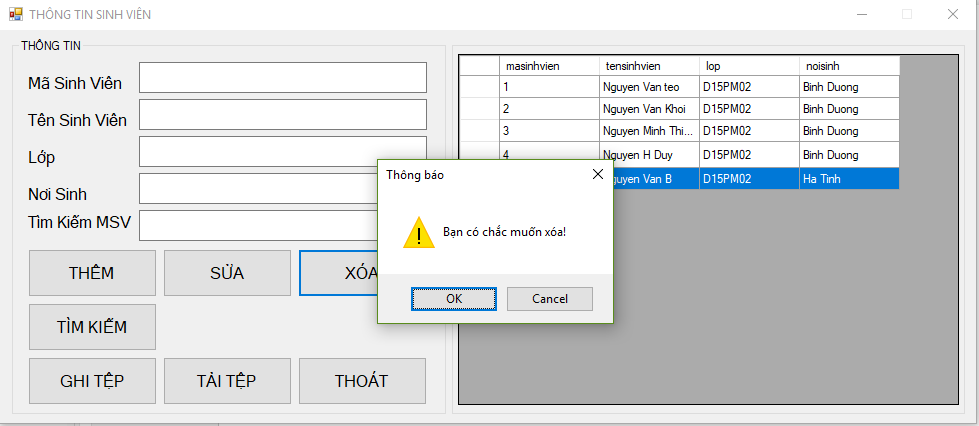


Tiến hành chính sửa và chọn “” để lưu lại

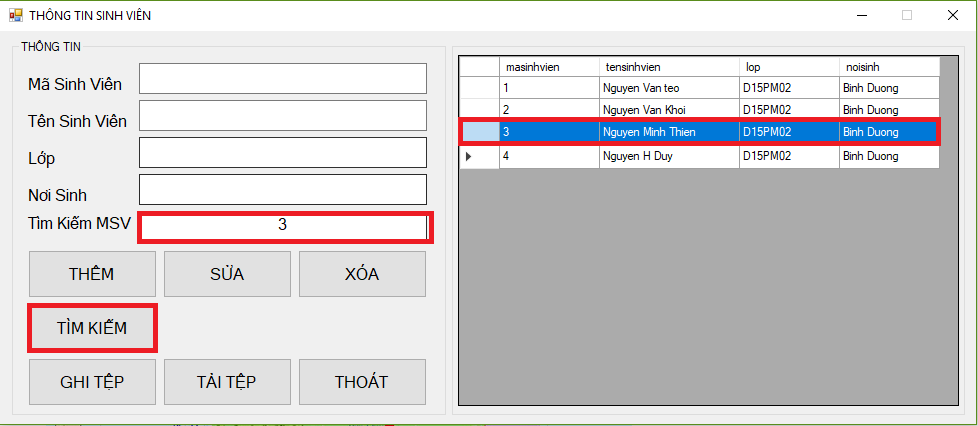


Chọn dòng cần xóa và chọn “” để xóa dòng đã chọn

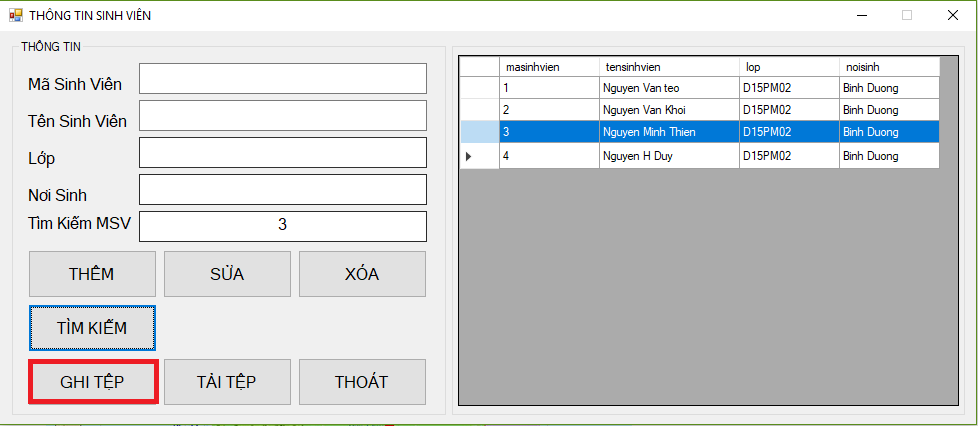




Điền thông tin mã sinh viên cần tìm vào ô tìm kiếm và chọn “” để tiền hành tìm kiếm sinh viên trong danh sách

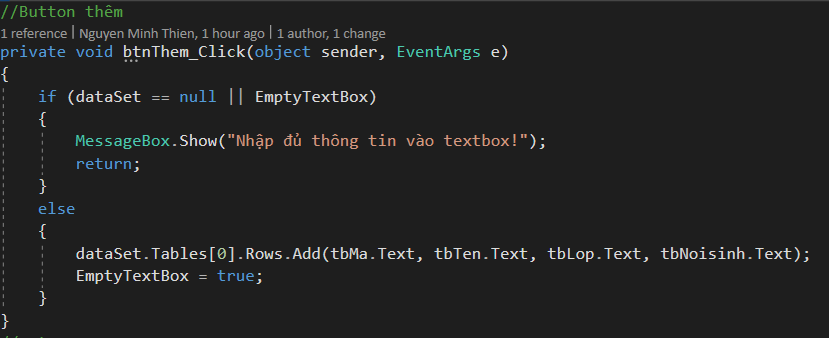


Chọn “” để tiến hành lưu tất cả vào file sinhvien.xml



**PHẦN 4: LẬP TRÌNH CHỨC NĂNG**

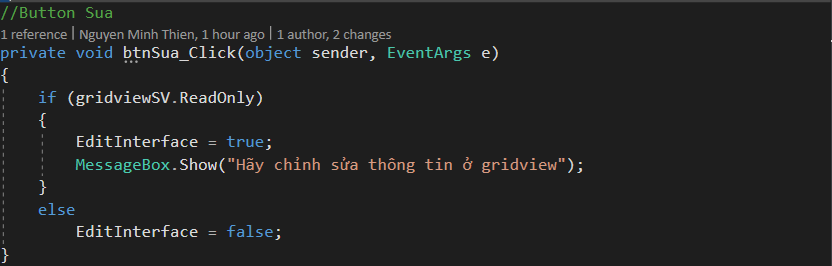
Lập trình chức năng sự kiện cho nút “Thêm”



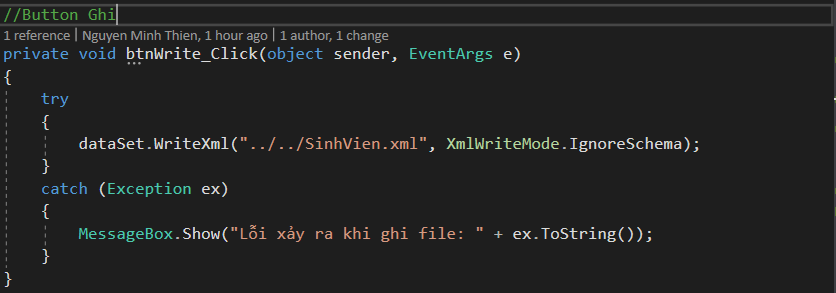
Lập trình chức năng sự kiện cho nút “Xóa”



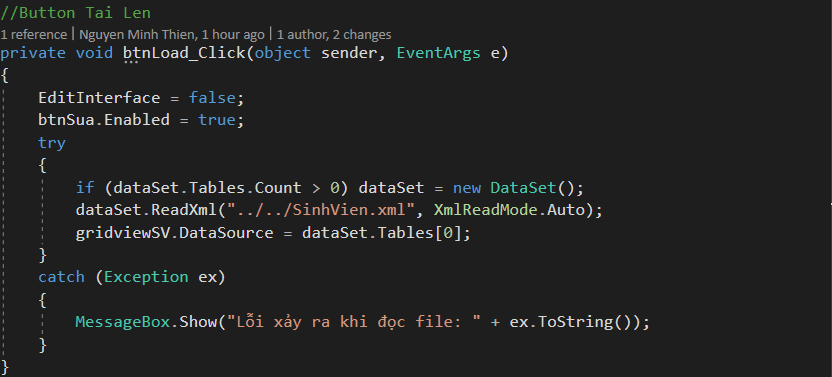
Lập trình chức năng sự kiện cho nút “Sửa”



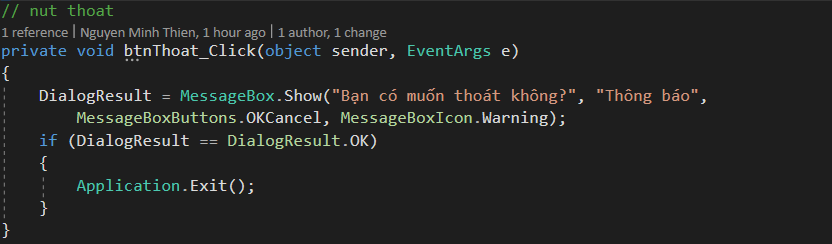
Lập trình chức năng sự kiện cho nút “Ghi Tệp”



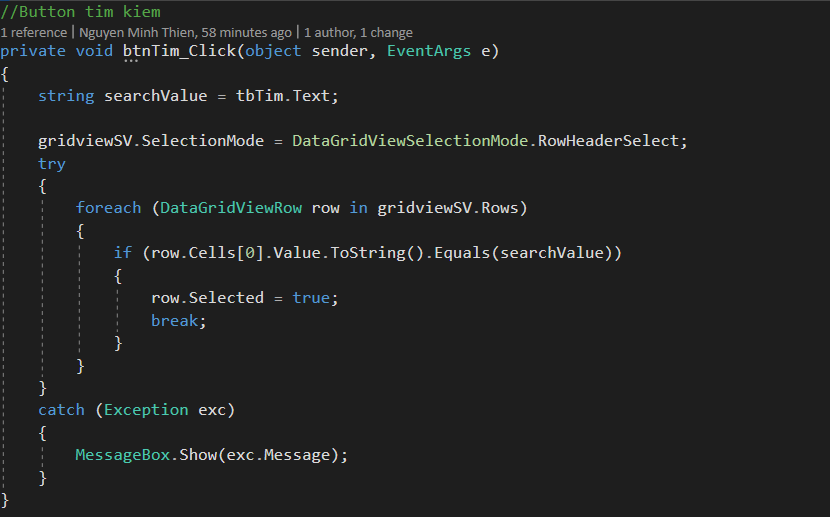
Lập trình chức năng sự kiện cho nút “Tải Lên”



Lập trình chức năng sự kiện cho nút “Thoát”



Lập trình chức năng sự kiện cho nút “Tìm Kiếm”



**PHẦN 5: PHỤ LỤC**

|  |
| --- |
| **CODE ThongTinSinhVien.cs** |
| using System;  using System.Collections.Generic;  using System.ComponentModel;  using System.Data;  using System.Drawing;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  using System.Windows.Forms;  using System.Xml;  namespace nhom1DemoXML  {  public partial class ThongTinSinhVien : Form  {  private DataSet dataSet;  private bool editInterface = false;  private bool EditInterface  {  get { return editInterface; }  set  {  gridviewSV.ReadOnly = !value;  btnThem.Enabled = !value;  btnSua.Text = value ? "LƯU" : "SỬA";  btnXoa.Enabled = !value;  btnTim.Enabled = !value;  btnWrite.Enabled = !value;  btnRead.Enabled = !value;  tbMa.Enabled = !value;  tbTen.Enabled = !value;  tbLop.Enabled = !value;  tbNoisinh.Enabled = !value;  tbTim.Enabled = !value;  }  }  private bool EmptyTextBox  {  get  {  if (tbMa.Text.Length > 0 || tbTen.Text.Length > 0 || tbLop.Text.Length > 0 ||  tbNoisinh.Text.Length > 0) return false;  else return true;  }  set  {  if (value)  {  tbMa.Text = "";  tbTen.Text = "";  tbLop.Text = "";  tbNoisinh.Text = "";  }  }  }  public ThongTinSinhVien()  {  InitializeComponent();  dataSet = new DataSet();  }  private void label1\_Click(object sender, EventArgs e)  {  }  // nut thoat  private void btnThoat\_Click(object sender, EventArgs e)  {  DialogResult = MessageBox.Show("Bạn có muốn thoát không?", "Thông báo",  MessageBoxButtons.OKCancel, MessageBoxIcon.Warning);  if (DialogResult == DialogResult.OK)  {  Application.Exit();  }  }  //Button Tai Len  private void btnLoad\_Click(object sender, EventArgs e)  {  EditInterface = false;  btnSua.Enabled = true;  try  {  if (dataSet.Tables.Count > 0) dataSet = new DataSet();  dataSet.ReadXml("../../SinhVien.xml", XmlReadMode.Auto);  gridviewSV.DataSource = dataSet.Tables[0];  }  catch (Exception ex)  {  MessageBox.Show("Lỗi xảy ra khi đọc file: " + ex.ToString());  }  }  //Button Ghi  private void btnWrite\_Click(object sender, EventArgs e)  {  try  {  dataSet.WriteXml("../../SinhVien.xml", XmlWriteMode.IgnoreSchema);  }  catch (Exception ex)  {  MessageBox.Show("Lỗi xảy ra khi ghi file: " + ex.ToString());  }  }  //Button Sua  private void btnSua\_Click(object sender, EventArgs e)  {  if (gridviewSV.ReadOnly)  {  EditInterface = true;  MessageBox.Show("Hãy chỉnh sửa thông tin ở gridview");  }  else  EditInterface = false;  }  //Button thêm  private void btnThem\_Click(object sender, EventArgs e)  {  if (dataSet == null || EmptyTextBox)  {  MessageBox.Show("Nhập đủ thông tin vào textbox!");  return;  }  else  {  dataSet.Tables[0].Rows.Add(tbMa.Text, tbTen.Text, tbLop.Text, tbNoisinh.Text);  EmptyTextBox = true;  }  }  //Button xoa  private void btnXoa\_Click(object sender, EventArgs e)  {  if (gridviewSV.SelectedRows.Count == 0)  {  MessageBox.Show("Hãy chọn dòng cần xóa!");  return;  }  DialogResult = MessageBox.Show("Bạn có chắc muốn xóa!", "Thông báo",  MessageBoxButtons.OKCancel, MessageBoxIcon.Warning);  if (DialogResult == DialogResult.OK)  {  dataSet.Tables[0].Rows.RemoveAt(gridviewSV.SelectedRows[0].Index);  }  }  private void ThongSinhVien\_Load(object sender, EventArgs e)  {  EditInterface = true;  btnSua.Enabled = false;  btnRead.Enabled = true;  }  //Button tim kiem  private void btnTim\_Click(object sender, EventArgs e)  {  string searchValue = tbTim.Text;  gridviewSV.SelectionMode = DataGridViewSelectionMode.RowHeaderSelect;  try  {  foreach (DataGridViewRow row in gridviewSV.Rows)  {  if (row.Cells[0].Value.ToString().Equals(searchValue))  {  row.Selected = true;  break;  }  }  }  catch (Exception exc)  {  MessageBox.Show(exc.Message);  }  }  }  } |

**TÀI LIỆU THAM KHẢO**

*[1]. Website:* [*http://docs.opennebula.org*](http://docs.opennebula.org) *(lần truy cập cuối: 17/12/2017).*

*[2]. Website: https://vi.wikipedia.org/wiki/Điện\_toán\_đám\_mây (lần truy cập cuối: 17/12/2017).*

*[3]. Website:* [*https://www.ibm.com/cloud/learn/iaas-paas-saas*](https://www.ibm.com/cloud/learn/iaas-paas-saas) *(lần truy cập cuối: 17/12/2017*).