**TRƯỜNG ĐẠI HỌC TÀI NGUYÊN VÀ MÔI TRƯỜNG THÀNH PHỐ HỒ CHÍ MINH**

**🙠🙠🕮🙢🙢**



**BÁO CÁO CUỐI KHOA**

**Môn học: PHÁT TRIỂN PHẦN MỀM MÃ NGUỒN MỞ**

**Đề tài: QUẢN LÝ SINH VIÊN**

|  |  |  |
| --- | --- | --- |
| Giảng viên hướng dẫn | **:** **ThS. Trần Văn Định** |  |
| Sinh viên thực hiện | **: Võ Xuân Anh Tuấn**  **: Trần Hồng Hoàng**  **: Bùi Henry** | **0950080116**  **0950080107**  **0950080103** |
| Lớp | **: THMT** | **Nhóm 3** |
| Khóa | **: 09** |  |

**Thành phố Hồ Chí Minh, ngày 23 tháng 5 năm 2023**

Nội dung

[**Phần 1: Tóm tắt lý thuyết** 5](#_Toc136123690)

[**Phần 2: Đặt tả dự án** 12](#_Toc136123691)

[**Phần 3: Thiết kế dự án** 13](#_Toc136123692)

[**Phần 4: Cài đặt** 37](#_Toc136123693)

[**Phần 5. Kết quả đạt được** 46](#_Toc136123694)

[**Phần 6. Kết luận** 47](#_Toc136123695)

**NHẬN XÉT CỦA GIẢNG VIÊN**

....................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................

LỜI NÓI ĐẦU

Trong bối cảnh này, phần mềm mã nguồn mở đã trở thành một lựa chọn phổ biến cho việc phát triển các ứng dụng quản lý sinh viên hiệu quả. Phần mềm mã nguồn mở không chỉ cung cấp các giải pháp kỹ thuật tiên tiến, mà còn giúp giảm thiểu chi phí phát triển và đảm bảo tính bảo mật, linh hoạt và khả năng mở rộng của hệ thống.

Đồ án này nhằm tìm hiểu và nghiên cứu về việc sử dụng phần mềm mã nguồn mở để phát triển hệ thống quản lý sinh viên. Chúng em đã tiến hành nghiên cứu các công nghệ phù hợp và lựa chọn một hệ thống phát triển phần mềm mã nguồn mở để triển khai quản lý sinh viên.

Bài báo cáo này sẽ trình bày chi tiết về quá trình phát triển hệ thống quản lý sinh viên dựa trên phần mềm mã nguồn mở. Chúng em, sẽ giới thiệu kiến thức nền tảng về phần mềm mã nguồn mở, công nghệ sử dụng và quy trình phát triển. Ngoài ra, chúng em cũng sẽ thảo luận về các lợi ích và thách thức của việc sử dụng phần mềm mã nguồn mở trong quản lý sinh viên.

# **Phần 1: Tóm tắt lý thuyết**

**1. Giới thiệu**

**1.1 Mục đích của tài liệu**

Tài liệu Đặc tả Yêu cầu này phác thảo các mục tiêu và thông số kỹ thuật cho Hệ thống Quản lý Sinh viên. Nó cung cấp một tổng quan toàn diện về các yêu cầu hệ thống và dành cho các nhà phát triển phần mềm và các bên liên quan tham gia vào dự án.

**1.2 Tóm tắt dự án**

Tên dự án: Hệ thống quản lý sinh viên

Người quản lý dự án: [Bùi Henry]

Nhà phân tích chịu trách nhiệm: [Trần Hồng Hoàng]

Người dùng chịu trách nhiệm: [Võ Xuân Anh Tuấn]

**1.3 Phạm vi dự án**

Tài liệu này xác định phạm vi của Hệ thống quản lý sinh viên và mối quan hệ của nó với các dự án liên quan khác. Nó phác thảo các chức năng và tính năng sẽ được đưa vào hệ thống và xác định ranh giới trách nhiệm của nó.

**1.4 Bối cảnh**

Hệ thống quản lý thông tin sinh viên hiện tại không hiệu quả và thiếu sự tích hợp giữa các phòng ban khác nhau. Hệ thống Quản lý Sinh viên được đề xuất nhằm mục đích giải quyết những thiếu sót này bằng cách cung cấp một nền tảng tập trung để quản lý hồ sơ sinh viên, đăng ký khóa học, chấm điểm và các nhiệm vụ liên quan khác. Hệ thống này sẽ nâng cao hiệu quả quản lý và nâng cao trải nghiệm tổng thể của sinh viên.

**1.5 Mục đích của hệ thống**

Hệ thống quản lý sinh viên được thiết kế để phục vụ các mục đích sau:

- Cung cấp một nền tảng tập trung để quản lý thông tin và hồ sơ sinh viên.

- Hợp lý hóa quy trình đăng ký khóa học và lên lịch trình.

- Tạo điều kiện giao tiếp hiệu quả giữa sinh viên, giảng viên và quản trị viên.

- Tạo báo cáo và phân tích để ra quyết định tốt hơn.

- Nâng cao trải nghiệm người dùng tổng thể và khả năng truy cập cho tất cả các bên liên quan.

**1.6 Tổng quan tài liệu**

Tài liệu này được tổ chức thành các phần sau:

- Giới thiệu: Cung cấp một cái nhìn tổng quan về tài liệu và nội dung của nó.

- Điều lệ: Mục tiêu chức năng: Mô tả các mục tiêu, yêu cầu chức năng của Hệ thống quản lý sinh viên.

- Điều lệ: Mục tiêu phi chức năng: Xác định các yêu cầu phi chức năng, bao gồm độ tin cậy, khả năng sử dụng, hiệu suất, bảo mật, khả năng hỗ trợ, ràng buộc thiết kế và các cân nhắc liên quan khác.

- Mô hình ngữ cảnh: Trình bày tuyên bố mục tiêu, sơ đồ ngữ cảnh và các phần bên ngoài hệ thống của Hệ thống quản lý sinh viên.

- Mô hình Ca sử dụng: Minh họa sơ đồ ca sử dụng chính và cung cấp các mô tả chi tiết về từng ca sử dụng.

- The Class Model: Phác thảo mô hình lớp và các thành phần của nó.

- Phần phụ lục: Bao gồm bảng chú giải các thuật ngữ và danh mục các tài liệu tham khảo được sử dụng trong tài liệu.

**2. Điều lệ: Mục tiêu chức năng**

**2.1 Ghi Danh Học Sinh**

- Mục tiêu: Cho phép sinh viên đăng ký học qua cổng thông tin trực tuyến.

- Mục đích công việc: Cung cấp một phương pháp thuận tiện và hiệu quả để sinh viên lựa chọn và đăng ký các khóa học mong muốn.

- Ưu tiên: Cao

- Kết quả đo lường được: Hệ thống sẽ cho phép sinh viên hoàn thành quy trình đăng ký khóa học trong một khung thời gian xác định.

**2.2 Hệ thống chấm điểm**

- Mục tiêu: Tự động hóa quy trình chấm điểm và cung cấp sổ điểm trực tuyến.

- Mục đích Công việc: Cải thiện độ chính xác và hiệu quả trong việc chấm điểm, đồng thời cung cấp cho sinh viên và giảng viên khả năng tiếp cận điểm số dễ dàng.

- Ưu tiên: Trung bình

- Kết quả đo lường được: Hệ thống sẽ tính điểm chính xác và cung cấp cho sinh viên và giảng viên trong thời gian thực.

**3. Điều lệ: Mục tiêu phi chức năng**

**3.1 Độ tin cậy**

- Mục tiêu: Đảm bảo tính sẵn sàng của hệ thống và giảm thiểu thời gian chết.

- Kết quả đo lường được: Hệ thống phải có tỷ lệ khả dụng là 99% và thời gian trung bình giữa các lần hỏng hóc (MTBF) ít nhất là 100 giờ.

**3.2 Khả năng sử dụng**

- Mục tiêu: Tạo giao diện thân thiện với người dùng để dễ dàng điều hướng và hoàn thành nhiệm vụ.

- Kết quả đo lường được: Hệ thống nên có thời gian đào tạo trung bình là một giờ đối với người dùng thông thường và 30 phút đối với người sử dụng điện.

**3.3 Hiệu năng**

- Mục tiêu: Đảm bảo hệ thống có thể xử lý đồng thời các yêu cầu của người dùng và cung cấp trải nghiệm đáp ứng.

- Kết quả đo lường được: Hệ thống có thể hỗ trợ tối thiểu 1000 người dùng đồng thời với thời gian phản hồi không quá 2 giây.

**3.4 Bảo mật**

- Mục tiêu: Bảo vệ thông tin sinh viên và ngăn chặn truy cập trái phép.

- Kết quả đo lường được: Hệ thống phải tuân thủ các giao thức bảo mật tiêu chuẩn của ngành, bao gồm mã hóa dữ liệu nhạy cảm và triển khai các cơ chế xác thực người dùng.

**3.5 Khả năng hỗ trợ**

- Mục tiêu: Đảm bảo tính dễ bảo trì và hỗ trợ cho hệ thống.

- Kết quả đo lường được: Hệ thống nên bao gồm tài liệu toàn diện và cung cấp giao diện trực quan cho quản trị viên để quản lý cấu hình hệ thống.

**3.6 Ràng buộc thiết kế**

- Mục tiêu: Tuân thủ các ràng buộc thiết kế cụ thể do tổ chức áp đặt.

- Kết quả đo lường được: Hệ thống phải đáp ứng các hướng dẫn của tổ chức về thương hiệu, khả năng tiếp cận và tiêu chuẩn công nghệ.

**3.7 Các cân nhắc khác**

- Mục tiêu: Giải quyết mọi yêu cầu phi chức năng bổ sung dành riêng cho Hệ thống quản lý sinh viên.

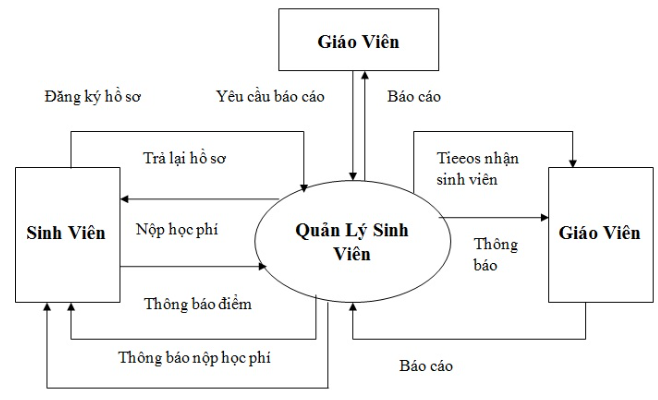
- Kết quả đo lường được: Hệ thống phải tuân thủ các luật và quy định liên quan quản lý quyền riêng tư và khả năng truy cập dữ liệu của học sinh.

**4. Mô hình bối cảnh**

**4.1 Tuyên bố mục tiêu**

Mục tiêu của Hệ thống quản lý sinh viên là cung cấp một nền tảng tập trung để quản lý thông tin sinh viên, đăng ký khóa học, chấm điểm và liên lạc, từ đó cải thiện hiệu quả quản trị và nâng cao trải nghiệm tổng thể của sinh viên.

**4.2 Biểu đồ ngữ cảnh**



**4.3 Bên Ngoài Hệ Thống**

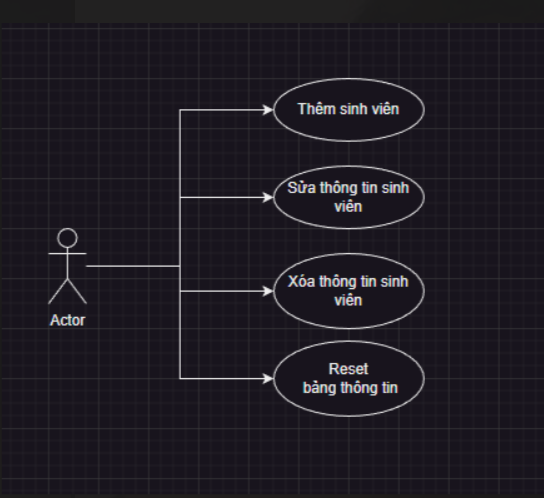
- Cổng thông tin sinh viên: Giao diện mà qua đó sinh viên truy cập vào hệ thống để đăng ký các khóa học, xem điểm và liên lạc với giảng viên và quản trị viên.

- Cổng thông tin của Khoa: Giao diện qua đó các giảng viên quản lý các khóa học, nhập điểm và giao tiếp với sinh viên và quản trị viên.

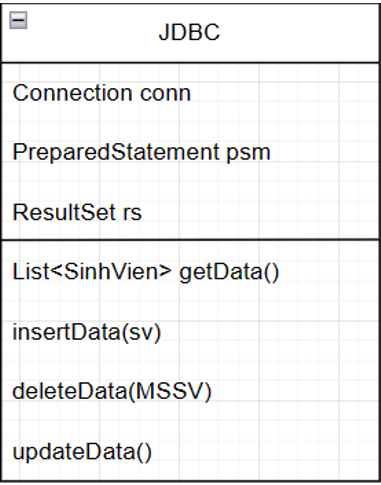
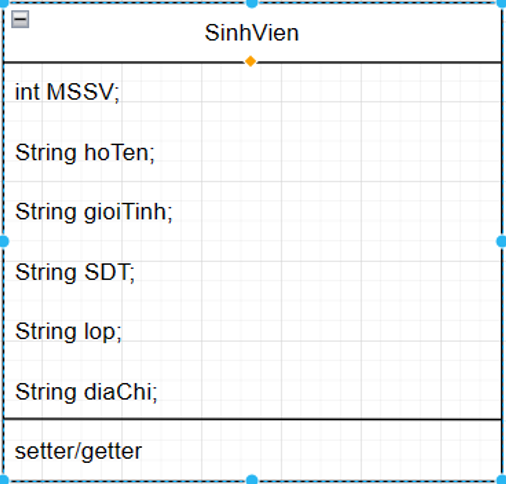
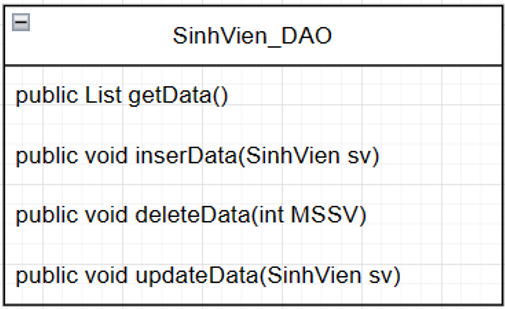
- Cổng quản trị: Giao diện qua đó quản trị viên quản lý hồ sơ học sinh, tạo báo cáo và thực hiện các tác vụ quản trị.

- Hệ thống bên ngoài (ví dụ: Hệ thống quản lý học tập): Điểm tích hợp với các hệ thống bên ngoài để trao đổi dữ liệu và thông tin liên quan.

**5. Mô hình ca sử dụng**



**6. Mô hình lớp**

# **Phần 2: Đặt tả dự án**

Dự án quản lý sinh viên là một hệ thống được thiết kế để quản lý thông tin về sinh viên, lớp học trong một trường đại học hoặc một tổ chức giáo dục tương tự. Mục tiêu chính của dự án này là tạo ra một nền tảng để quản lý thông tin cá nhân của sinh viên, ghi nhận quá trình học tập và cung cấp các chức năng quản lý tiện lợi cho cán bộ quản lý.

Các tính năng chính của hệ thống quản lý sinh viên có thể bao gồm:

**1. Quản lý thông tin sinh viên:**

- Lưu trữ thông tin cá nhân của sinh viên bao gồm tên, giới tính, địa chỉ và thông tin liên hệ khác.

**2. Quản lý danh sách sinh viên:**

- Ghi nhận thông tin về các lớp học, tên lớp, họ tên, số điện thoại, địa chỉ

- Quản lý số lượng sinh viên đăng ký vào từng lớp học.

**3. Tra cứu thông tin:**

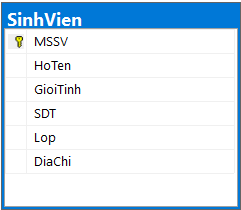
- Cung cấp khả năng tra cứu thông tin sinh viên, thông tin cá nhân,

- Cung cấp công cụ tìm kiếm linh hoạt để tìm kiếm sinh viên theo tiêu chí như tên, mã sinh viên hoặc lớp học.

# **Phần 3: Thiết kế dự án**

**3.1 DATABASE**

- Tên Database: QLSV

- Bảng dữ liệu:

+ MSSV: Mã số sinh viên

+ HoTen: Họ và tên

+ GioiTinh: Giới tính

+ SDT: Số điện thoại

+ Lop: Lớp

+ DiaChi: Địa chỉ  
- Code Database:

USE [QLSV]

GO

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[SinhVien](

[MSSV] [int] IDENTITY(1,1) NOT NULL,

[HoTen] [nvarchar](50) NOT NULL,

[GioiTinh] [nvarchar](3) NOT NULL,

[SDT] [nchar](11) NULL,

[Lop] [nchar](10) NOT NULL,

[DiaChi] [nvarchar](50) NULL,

CONSTRAINT [PK\_SinhVien] PRIMARY KEY CLUSTERED

(

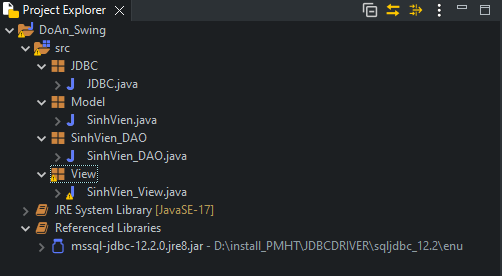
[MSSV] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

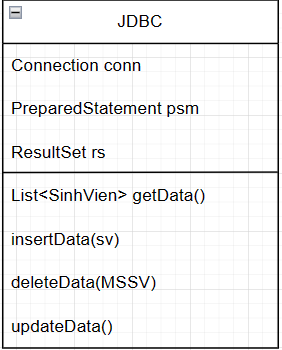
) ON [PRIMARY]

GO

**3.2 THIẾT KẾ CLASS**



3.2.1 JDBC.Java



* Code:

package JDBC;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

public class JDBC {

public static Connection getConnect() {

Connection conn = null;

String url = "jdbc:sqlserver://localhost:1433;databaseName=QLSV;encrypt=true;trustServerCertificate=true";

String user = "sa";

String pass = "123";

try {

conn = DriverManager.getConnection(url, user, pass)

System.out.println("Kết nối thành công !");

} catch (SQLException e) {

System.out.println("Kết nối thất bại !");

e.printStackTrace();

}

return conn;

}

public static void closeConnect(Connection conn) {

if(conn != null) {

try {

conn.close();

System.out.println("Đóng kết nối thành công !");

} catch (SQLException e) {

System.out.println("Đóng kết nối thất bại !");

e.printStackTrace();

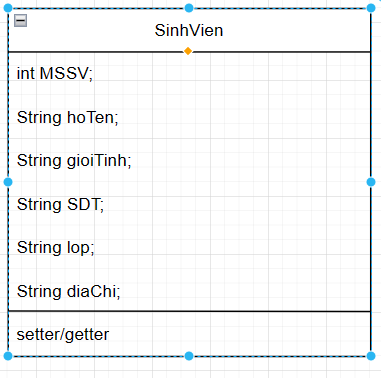
}

}

}

}

**3.2.2 SinhVien.Java**



* Code:

package Model

public class SinhVien {

int MSSV;

String hoTen;

String gioiTinh;

String SDT;

String lop;

String diaChi;

public SinhVien() {

}

public SinhVien(String hoTen, String gioiTinh, String SDT, String lop, String diaChi) {

this.hoTen = hoTen;

this.gioiTinh = gioiTinh;

this.SDT = SDT;

this.lop = lop;

this.diaChi = diaChi;

}

public SinhVien(int MSSV, String hoTen, String gioiTinh, String SDT, String lop, String diaChi) {

this.MSSV = MSSV;

this.hoTen = hoTen;

this.gioiTinh = gioiTinh;

this.SDT = SDT;

this.lop = lop;

this.diaChi = diaChi;

}

public int getMSSV() {

return MSSV;

}

public void setMSSV(int mSSV) {

MSSV = mSSV;

}

public String getHoTen() {

return hoTen;

}

public void setHoTen(String hoTen) {

this.hoTen = hoTen;

}

public String getGioiTinh() {

return gioiTinh;

}

public void setGioiTinh(String gioiTinh) {

this.gioiTinh = gioiTinh;

}

public String getSDT() {

return SDT;

}

public void setSDT(String sDT) {

SDT = sDT;

}

public String getLop() {

return lop;

}

public void setLop(String lop) {

this.lop = lop;

}

public String getDiaChi() {

return diaChi;

}

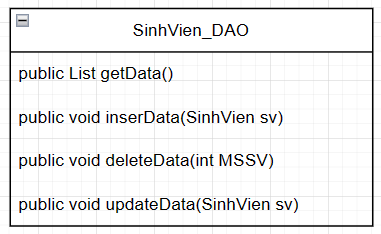
public void setDiaChi(String diaChi) {

this.diaChi = diaChi;

}

}

**3.2.3 SinhVien\_DAO**



* Code:

package SinhVien\_DAO;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.ArrayList;

import java.util.List;

import JDBC.JDBC;

import Model.SinhVien;

public class SinhVien\_DAO {

private Connection conn = null;

private PreparedStatement psm = null;

private ResultSet rs = null;

//getData

public List<SinhVien> getData(){

List<SinhVien> lstSinhVien = new ArrayList<>();

conn = JDBC.getConnect();

String sqlData = "SELECT \* FROM dbo.SinhVien";

try {

psm = conn.prepareStatement(sqlData);

rs = psm.executeQuery();

while(rs.next()) {

int mssv = rs.getInt("MSSV");

String hoTen = rs.getNString("HoTen");

String gioiTinh = rs.getString("gioiTinh");

String sdt = rs.getString("SDT");

String lop = rs.getString("Lop");

String diaChi = rs.getNString("DiaChi");

SinhVien sv = new SinhVien(mssv,hoTen,gioiTinh,sdt,lop,diaChi);

lstSinhVien.add(sv);

}

} catch (SQLException e) {

e.printStackTrace();

} finally {

JDBC.closeConnect(conn);

if(psm != null) {

try {

psm.close();

} catch (SQLException e) {

e.printStackTrace();

}

}

if(rs != null) {

try {

rs.close();

} catch (SQLException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

}

return lstSinhVien;

}

//insertData

public void inserData(SinhVien sv) {

String insertSQL = "INSERT INTO dbo.SinhVien (HoTen, GioiTinh, SDT, Lop, DiaChi) "

+ " Values(? , ? , ? , ?, ?)";

conn = JDBC.getConnect();

try {

psm = conn.prepareStatement(insertSQL, Statement.RETURN\_GENERATED\_KEYS);

psm.setString(1, sv.getHoTen());

psm.setString(2, sv.getGioiTinh());

psm.setString(3, sv.getSDT());

psm.setString(4, sv.getLop());

psm.setString(5, sv.getDiaChi());

int affectedRows = psm.executeUpdate();

if (affectedRows > 0) {

ResultSet generatedKeys = psm.getGeneratedKeys();

if (generatedKeys.next()) {

int generatedMSSV = generatedKeys.getInt(1);

sv.setMSSV(generatedMSSV);

}

}

} catch (SQLException e) {

e.printStackTrace();

} finally {

JDBC.closeConnect(conn);

if(psm != null) {

try {

psm.close();

} catch (SQLException e) {

}

}

if(rs != null) {

try {

rs.close();

} catch (SQLException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

}

}

//DeleteData

public void deleteData(int MSSV) {

conn = JDBC.getConnect();

String deleteSQL = "DELETE FROM dbo.SinhVien WHERE MSSV = ?";

try {

psm = conn.prepareStatement(deleteSQL);

psm.setInt(1, MSSV);

int rowDelete = psm.executeUpdate();

if(rowDelete > 0) {

System.out.println("Xóa thành công !");

} else {

System.out.println("Xóa thất bại ! Không tìm thấy MSSV !");

}

} catch (SQLException e) {

System.out.println("Xóa thất bại !");

e.printStackTrace();

} finally {

JDBC.closeConnect(conn);

if(psm != null) {

try {

psm.close();

} catch (SQLException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

}

}

//UpdateData

public void updateData(SinhVien sv) {

try {

// Kết nối cơ sở dữ liệu và chuẩn bị truy vấn SQL

Connection conn = JDBC.getConnect();

String query = "UPDATE SinhVien SET HoTen = ?, GioiTinh = ?, SDT = ?, Lop = ?, DiaChi = ? WHERE MSSV = ?";

psm = conn.prepareStatement(query);

// Đặt các tham số truy vấn SQL từ đối tượng SinhVien

psm.setString(1, sv.getHoTen());

psm.setString(2, sv.getGioiTinh());

psm.setString(3, sv.getSDT());

psm.setString(4, sv.getLop());

psm.setString(5, sv.getDiaChi());

psm.setInt(6, sv.getMSSV());

int rowsAffected = psm.executeUpdate();

if (rowsAffected > 0) {

System.out.println("Dữ liệu sinh viên đã được cập nhật thành công!");

} else {

System.out.println("Không tìm thấy sinh viên để cập nhật hoặc dữ liệu không thay đổi!");

}

psm.close();

conn.close();

} catch (SQLException e) {

e.printStackTrace();

}

}

//Tìm kiếm theo tên

public List<SinhVien> findAllByName(String name){

List<SinhVien> kq = new ArrayList<>();

conn = JDBC.getConnect();

String fiSQL = "SELECT \* FROM dbo.SinhVien WHERE HoTen LIKE ?";

try {

psm = conn.prepareStatement(fiSQL);

psm.setString(1, "%" + name + "%");

rs = psm.executeQuery();

while(rs.next()) {

int MSSV = rs.getInt("MSSV");

String hoTen = rs.getString("HoTen");

String gioiTinh = rs.getString("gioiTinh");

String sdt = rs.getString("SDT");

String lop = rs.getString("Lop");

String diaChi = rs.getNString("DiaChi");

SinhVien sv = new SinhVien(MSSV,hoTen,gioiTinh,sdt,lop,diaChi);

kq.add(sv);

}

} catch (SQLException e) {

e.printStackTrace();

} finally {

JDBC.closeConnect(conn);

if(psm != null) {

try {

psm.close();

} catch (SQLException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

if(rs != null) {

try {

rs.close();

} catch (SQLException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

}

return kq;

}

}

3.2.4 SinhVien\_View

- Code:

# package View;

# import java.awt.BorderLayout;

# import java.awt.Dimension;

# import java.awt.EventQueue;

# import java.awt.Toolkit;

# import java.util.ArrayList;

# import java.util.List;

# import javax.swing.JFrame;

# import javax.swing.JTable;

# import javax.swing.table.DefaultTableModel;

# import Model.SinhVien;

# import SinhVien\_DAO.SinhVien\_DAO;

# import javax.swing.JScrollPane;

# import java.awt.Font;

# import javax.swing.JLabel;

# import javax.swing.JOptionPane;

# import javax.swing.SwingConstants;

# import javax.swing.JPanel;

# import javax.swing.JTextField;

# import javax.swing.JRadioButton;

# import javax.print.attribute.standard.JobMessageFromOperator;

# import javax.swing.ButtonGroup;

# import java.awt.Color;

# import javax.swing.border.TitledBorder;

# import javax.swing.JButton;

# import java.awt.event.ActionListener;

# import java.awt.event.ActionEvent;

# import java.awt.event.MouseAdapter;

# import java.awt.event.MouseEvent;

# import javax.swing.JComboBox;

# public class SinhVien\_View {

# private JFrame frmQunLDanh;

# private JTable table;

# private JTextField txtHoTen;

# private JTextField txtSDT;

# private final ButtonGroup buttonGroup = new ButtonGroup();

# private JTextField txtLop;

# private JTextField txtDiaChi;

# private List<SinhVien> lstSinhVien;

# private JTextField txtSrcTen;

# //main

# public static void main(String[] args) {

# EventQueue.invokeLater(new Runnable() {

# public void run() {

# try {

# SinhVien\_View window = new SinhVien\_View();

# window.frmQunLDanh.setVisible(true);

# } catch (Exception e) {

# e.printStackTrace();

# }

# }

# });

# }

# //hiển thị app

# public SinhVien\_View() {

# initialize();

# showData();

# 

# }

# //cài đặt app hiển thị

# @SuppressWarnings("serial")

# 

# 

# private void initialize() {

# frmQunLDanh = new JFrame();

# frmQunLDanh.setTitle("QUẢN LÝ DANH SÁCH SINH VIÊN");

# frmQunLDanh.setSize(838, 493);

# 

# setupFarme(frmQunLDanh);

# frmQunLDanh.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

# frmQunLDanh.getContentPane().setLayout(null);

# frmQunLDanh.getContentPane().setLayout(null);

# 

# JScrollPane scrollPane = new JScrollPane();

# scrollPane.setBounds(244, 59, 568, 296);

# frmQunLDanh.getContentPane().add(scrollPane);

# 

# 

# int doCaoCot = 20 ;

# 

# table = new JTable();

# 

# table.setRowHeight(doCaoCot);

# 

# table.setModel(new DefaultTableModel(

# new Object[][] {

# },

# new String[] {

# "MSSV", "H\u1ECD v\u00E0 t\u00EAn", "Gi\u1EDBi t\u00EDnh", "S\u0110T", "L\u1EDBp", "\u0110\u1ECBa ch\u1EC9"

# }

# ) {

# boolean[] columnEditables = new boolean[] {

# false, false, false, false, false, false

# };

# public boolean isCellEditable(int row, int column) {

# return columnEditables[column];

# }

# });

# table.getColumnModel().getColumn(0).setPreferredWidth(50);

# table.getColumnModel().getColumn(0).setMinWidth(50);

# table.getColumnModel().getColumn(1).setPreferredWidth(160);

# table.getColumnModel().getColumn(1).setMinWidth(100);

# table.getColumnModel().getColumn(2).setPreferredWidth(60);

# table.getColumnModel().getColumn(2).setMinWidth(20);

# table.getColumnModel().getColumn(3).setMinWidth(50);

# table.getColumnModel().getColumn(4).setMinWidth(50);

# table.getColumnModel().getColumn(5).setPreferredWidth(96);

# table.getColumnModel().getColumn(5).setMinWidth(50);

# 

# table.setFont(new Font("Tahoma", Font.PLAIN, 15));

# scrollPane.setViewportView(table);

# 

# 

# JLabel lblNewLabel = new JLabel("DANH SÁCH SINH VIÊN KHOA HỆ THỐNG THÔNG TIN VÀ VIỄN THÁM");

# lblNewLabel.setHorizontalAlignment(SwingConstants.CENTER);

# lblNewLabel.setFont(new Font("Tahoma", Font.PLAIN, 20));

# lblNewLabel.setBounds(0, 0, 822, 25);

# frmQunLDanh.getContentPane().add(lblNewLabel);

# frmQunLDanh.getContentPane().add(lblNewLabel, BorderLayout.NORTH);

# 

# JPanel panel = new JPanel();

# panel.setBorder(new TitledBorder(null, "Nh\u1EADp th\u00F4ng tin sinh vi\u00EAn", TitledBorder.LEADING, TitledBorder.TOP, null, null));

# panel.setForeground(new Color(255, 255, 255));

# panel.setBounds(10, 53, 224, 302);

# frmQunLDanh.getContentPane().add(panel);

# panel.setLayout(null);

# 

# JLabel lblHoVaTen = new JLabel("Họ và tên:");

# lblHoVaTen.setBounds(10, 21, 86, 14);

# lblHoVaTen.setFont(new Font("Tahoma", Font.PLAIN, 13));

# panel.add(lblHoVaTen);

# 

# JLabel lblGioiTinh = new JLabel("Giới tính:");

# lblGioiTinh.setBounds(10, 77, 55, 14);

# lblGioiTinh.setFont(new Font("Tahoma", Font.PLAIN, 13));

# panel.add(lblGioiTinh);

# 

# JLabel lblSDT = new JLabel("SDT:");

# lblSDT.setBounds(10, 128, 55, 14);

# lblSDT.setFont(new Font("Tahoma", Font.PLAIN, 13));

# panel.add(lblSDT);

# 

# JLabel lblLop = new JLabel("Lớp:");

# lblLop.setBounds(10, 184, 55, 14);

# lblLop.setFont(new Font("Tahoma", Font.PLAIN, 13));

# panel.add(lblLop);

# 

# JLabel lblDiaChi = new JLabel("Địa chỉ:");

# lblDiaChi.setBounds(10, 240, 55, 14);

# lblDiaChi.setFont(new Font("Tahoma", Font.PLAIN, 13));

# panel.add(lblDiaChi);

# 

# txtHoTen = new JTextField();

# txtHoTen.setBounds(10, 46, 193, 20);

# panel.add(txtHoTen);

# txtHoTen.setColumns(10);

# 

# JRadioButton rdNam = new JRadioButton("Nam");

# rdNam.setBounds(83, 98, 55, 23);

# buttonGroup.add(rdNam);

# panel.add(rdNam);

# 

# JRadioButton rdNu = new JRadioButton("Nữ");

# rdNu.setBounds(10, 98, 55, 23);

# buttonGroup.add(rdNu);

# rdNu.setSelected(true);

# panel.add(rdNu);

# 

# txtSDT = new JTextField();

# txtSDT.setBounds(10, 153, 193, 20);

# panel.add(txtSDT);

# txtSDT.setColumns(10);

# 

# txtLop = new JTextField();

# txtLop.setBounds(10, 209, 193, 20);

# txtLop.setColumns(10);

# panel.add(txtLop);

# 

# txtDiaChi = new JTextField();

# txtDiaChi.setBounds(10, 265, 193, 20);

# txtDiaChi.setColumns(10);

# panel.add(txtDiaChi);

# 

# JButton btnSave\_ = new JButton("Save");

# btnSave\_.addActionListener(new ActionListener() {

# public void actionPerformed(ActionEvent e) {

# 

# String hoTen = txtHoTen.getText();

# String gioiTinh = rdNam.isSelected() ? "Nam" : "Nữ";

# String sdt = txtSDT.getText();

# String lop = txtLop.getText();

# String diaChi = txtDiaChi.getText();

# 

# 

# if(hoTen.isEmpty()) {

# JOptionPane.showMessageDialog(null, "Vui lòng nhập Họ và tên của sinh viên !");

# } else if(lop.isEmpty()) {

# JOptionPane.showMessageDialog(null, "Vui lòng nhập lớp của sinh viên !");

# } else {

# 

# SinhVien sv = new SinhVien(hoTen, gioiTinh, sdt, lop, diaChi);

# SinhVien\_DAO svDAO = new SinhVien\_DAO();

# 

# svDAO.inserData(sv);

# showData();

# 

# txtHoTen.setText("");

# txtSDT.setText("");

# txtLop.setText("");

# 

# txtDiaChi.setText("");

# 

# if(rdNam.isSelected()) {

# rdNu.setSelected(true);

# }

# }

# 

# }

# });

# 

# 

# 

# btnSave\_.setBounds(10, 366, 95, 32);

# frmQunLDanh.getContentPane().add(btnSave\_);

# btnSave\_.setFocusPainted(false);

# 

# 

# 

# JButton btnReset\_ = new JButton("Reset");

# btnReset\_.addActionListener(new ActionListener() {

# public void actionPerformed(ActionEvent e) {

# txtHoTen.setText("");

# txtSDT.setText("");

# txtLop.setText("");

# 

# txtDiaChi.setText("");

# if(rdNam.isSelected()) {

# rdNu.setSelected(true);

# }

# }

# });

# 

# btnReset\_.setBounds(10, 409, 95, 32);

# frmQunLDanh.getContentPane().add(btnReset\_);

# btnReset\_.setFocusPainted(false);

# 

# table.addMouseListener(new MouseAdapter() {

# @Override

# public void mouseClicked(MouseEvent e) {

# 

# int seData = table.getSelectedRow();

# 

# if(seData != -1) {

# 

# int MSSV = (int) table.getValueAt(seData, 0);

# String hoTen = (String) table.getValueAt(seData, 1);

# String gt = (String) table.getValueAt(seData, 2);

# String sdt = (String) table.getValueAt(seData, 3);

# String lop = (String) table.getValueAt(seData, 4);

# String diaChi = (String)table.getValueAt(seData, 5);

# 

# txtHoTen.setText(hoTen);

# if(gt.equals("Nam")) {

# rdNam.setSelected(true);

# } else {

# rdNu.setSelected(true);

# }

# 

# txtSDT.setText(sdt);

# txtLop.setText(lop);

# txtDiaChi.setText(diaChi);

# }

# 

# }

# });

# 

# JButton btnUpdate\_ = new JButton("Update");

# btnUpdate\_.addActionListener(new ActionListener() {

# public void actionPerformed(ActionEvent e) {

# 

# int seData = table.getSelectedRow();

# int MSSV = (int) table.getValueAt(seData, 0);

# 

# String hoTen = txtHoTen.getText();

# String gt = rdNam.isSelected() ? "Nam" : "Nữ";

# String lop = txtLop.getText();

# String sdt = txtSDT.getText();

# String diaChi = txtDiaChi.getText();

# 

# SinhVien sv = new SinhVien(MSSV, hoTen, gt, sdt, lop, diaChi);

# SinhVien\_DAO svDAO = new SinhVien\_DAO();

# svDAO.updateData(sv);

# showData();

# 

# }

# });

# btnUpdate\_.setBounds(139, 366, 95, 32);

# frmQunLDanh.getContentPane().add(btnUpdate\_);

# btnUpdate\_.setFocusPainted(false);

# 

# 

# 

# JButton btnDelete\_ = new JButton("Delete");

# btnDelete\_.addActionListener(new ActionListener() {

# public void actionPerformed(ActionEvent e) {

# 

# int viTriData = table.getSelectedRow();

# 

# if(viTriData != -1 ) {//Dữ liệu được chọn

# 

# int option = JOptionPane.showConfirmDialog(null, "Xác nhận xóa sinh viên này ?");

# 

# if(option == 1 || option == 2 ) {

# return;

# }

# int mssv = (int) table.getValueAt(viTriData, 0);

# 

# SinhVien\_DAO svDAO = new SinhVien\_DAO();

# svDAO.deleteData(mssv);

# 

# showData();

# } else {

# 

# JOptionPane.showMessageDialog(null, "Vui lòng chọn sinh viên để xóa !");

# 

# }

# }

# 

# });

# btnDelete\_.setBounds(139, 409, 95, 32);

# frmQunLDanh.getContentPane().add(btnDelete\_);

# btnDelete\_.setFocusPainted(false);

# 

# JPanel panel\_1 = new JPanel();

# panel\_1.setBorder(new TitledBorder(null, "T\u00ECm ki\u1EBFm", TitledBorder.LEADING, TitledBorder.TOP, null, null));

# panel\_1.setBounds(244, 366, 386, 75);

# frmQunLDanh.getContentPane().add(panel\_1);

# panel\_1.setLayout(null);

# 

# JLabel lblSrcTen = new JLabel("Họ và tên:");

# lblSrcTen.setFont(new Font("Tahoma", Font.PLAIN, 13));

# lblSrcTen.setBounds(10, 33, 83, 14);

# panel\_1.add(lblSrcTen);

# 

# txtSrcTen = new JTextField();

# txtSrcTen.setBounds(77, 30, 180, 20);

# panel\_1.add(txtSrcTen);

# txtSrcTen.setColumns(10);

# 

# JButton btnSrc = new JButton("Tìm kiếm");

# btnSrc.addActionListener(new ActionListener() {

# public void actionPerformed(ActionEvent e) {

# 

# String srcTen = txtSrcTen.getText();

# 

# 

# SinhVien\_DAO svDAO = new SinhVien\_DAO();

# List<SinhVien> kq = svDAO.findAllByName(srcTen);

# 

# DefaultTableModel model = (DefaultTableModel) table.getModel();

# model.setRowCount(0);

# for (SinhVien sv : kq) {

# Object[] row = { sv.getMSSV(), sv.getHoTen(), sv.getGioiTinh(), sv.getSDT(), sv.getLop(), sv.getDiaChi() };

# model.addRow(row);

# }

# 

# 

# }

# });

# btnSrc.setBounds(267, 29, 89, 23);

# panel\_1.add(btnSrc);

# 

# 

# }

# 

# //ShowData

# private void showData() {

# SinhVien\_DAO svDAO = new SinhVien\_DAO();

# lstSinhVien = svDAO.getData();

# DefaultTableModel model = (DefaultTableModel) table.getModel();

# model.setRowCount(0);

# for (SinhVien sv : lstSinhVien) {

# Object[] row = { sv.getMSSV(), sv.getHoTen(), sv.getGioiTinh(), sv.getSDT(), sv.getLop(), sv.getDiaChi() };

# model.addRow(row);

# }

# }

# 

# //Frame ở giữa màn hình

# public void setupFarme(JFrame frame) {

# 

# Dimension sizeManHinh = Toolkit.getDefaultToolkit().getScreenSize();

# 

# int manHinh\_cr = sizeManHinh.width;

# int manHinh\_cc = sizeManHinh.height;

# 

# int frame\_cr = frame.getWidth();

# int frame\_cc = frame.getHeight();

# 

# int viTri\_X = (manHinh\_cr - frame\_cr) / 2;

# int viTri\_Y = (manHinh\_cc - frame\_cc) / 2;

# 

# frame.setLocation(viTri\_X,viTri\_Y);

# }

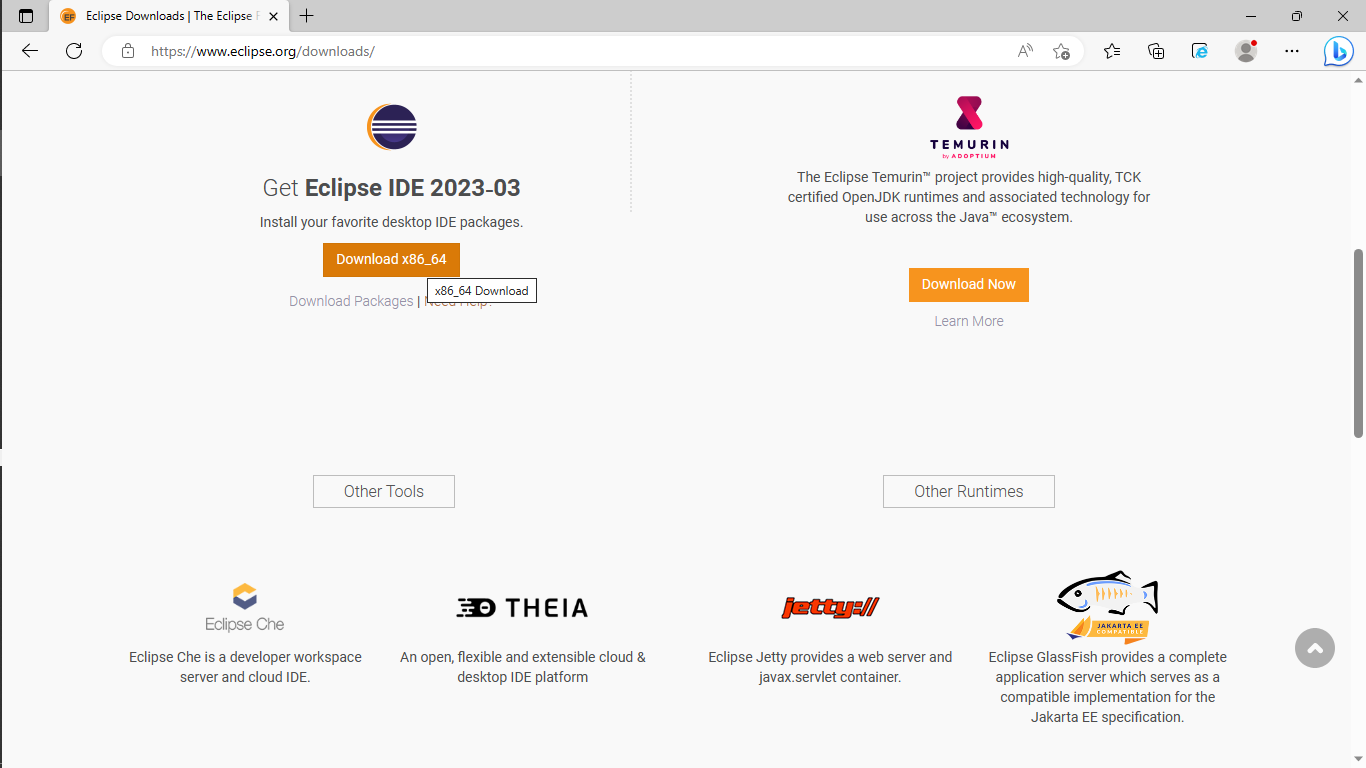
# }

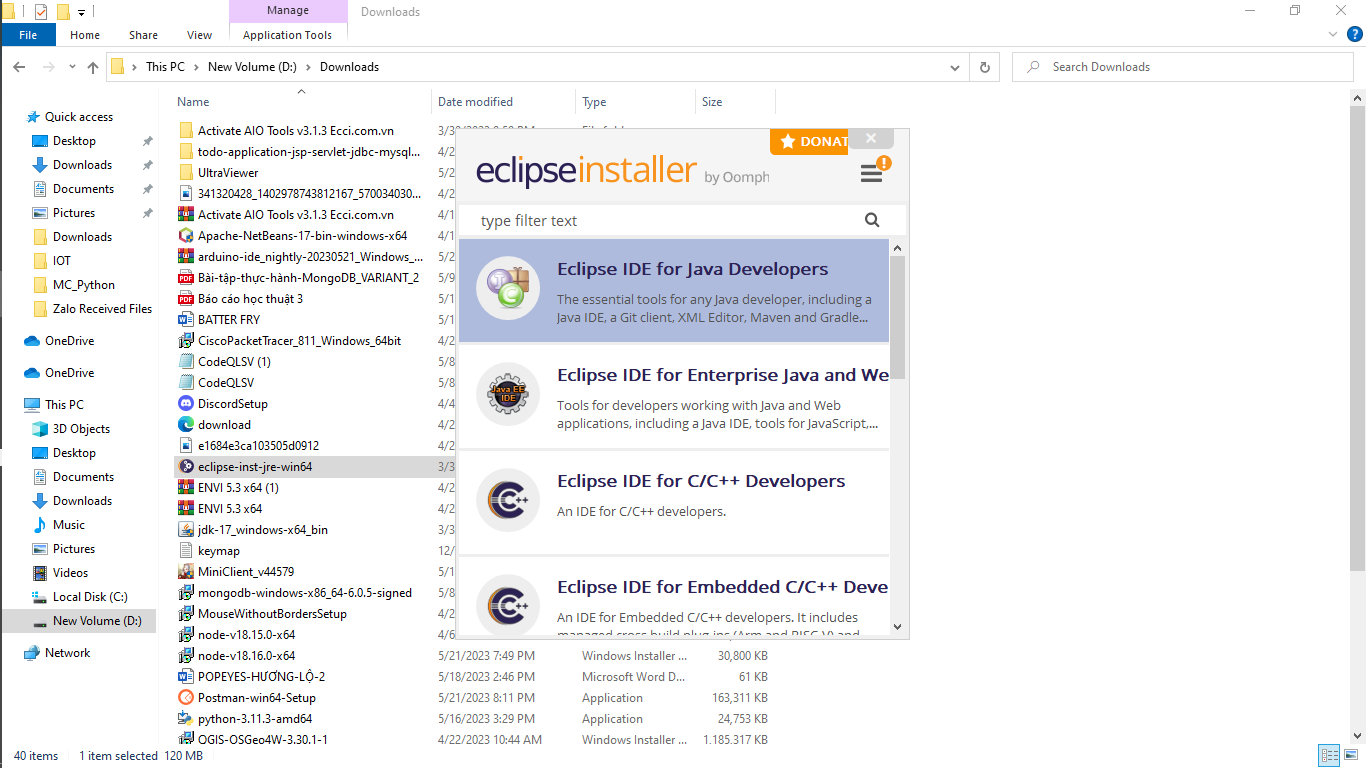
# **Phần 4: Cài đặt**

**4.1 IDE**

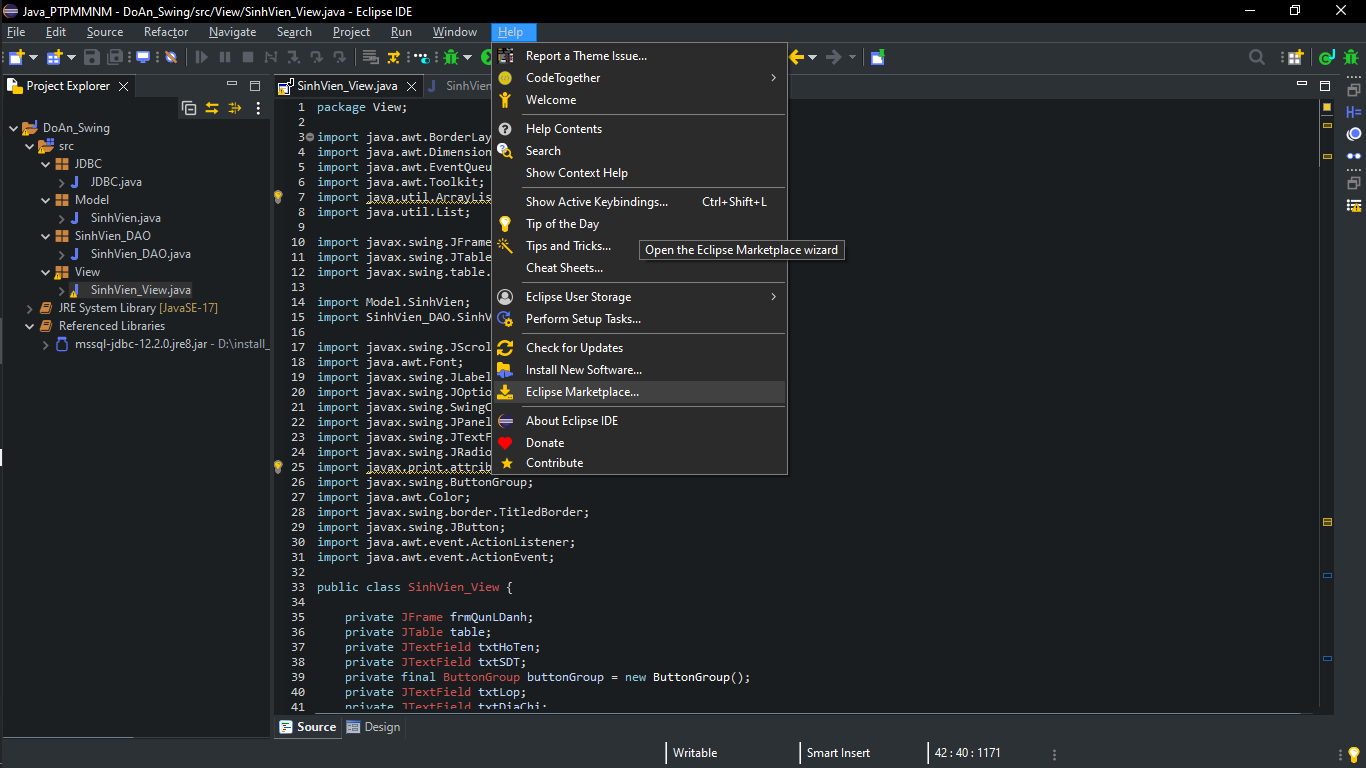
IDE: Eclipse

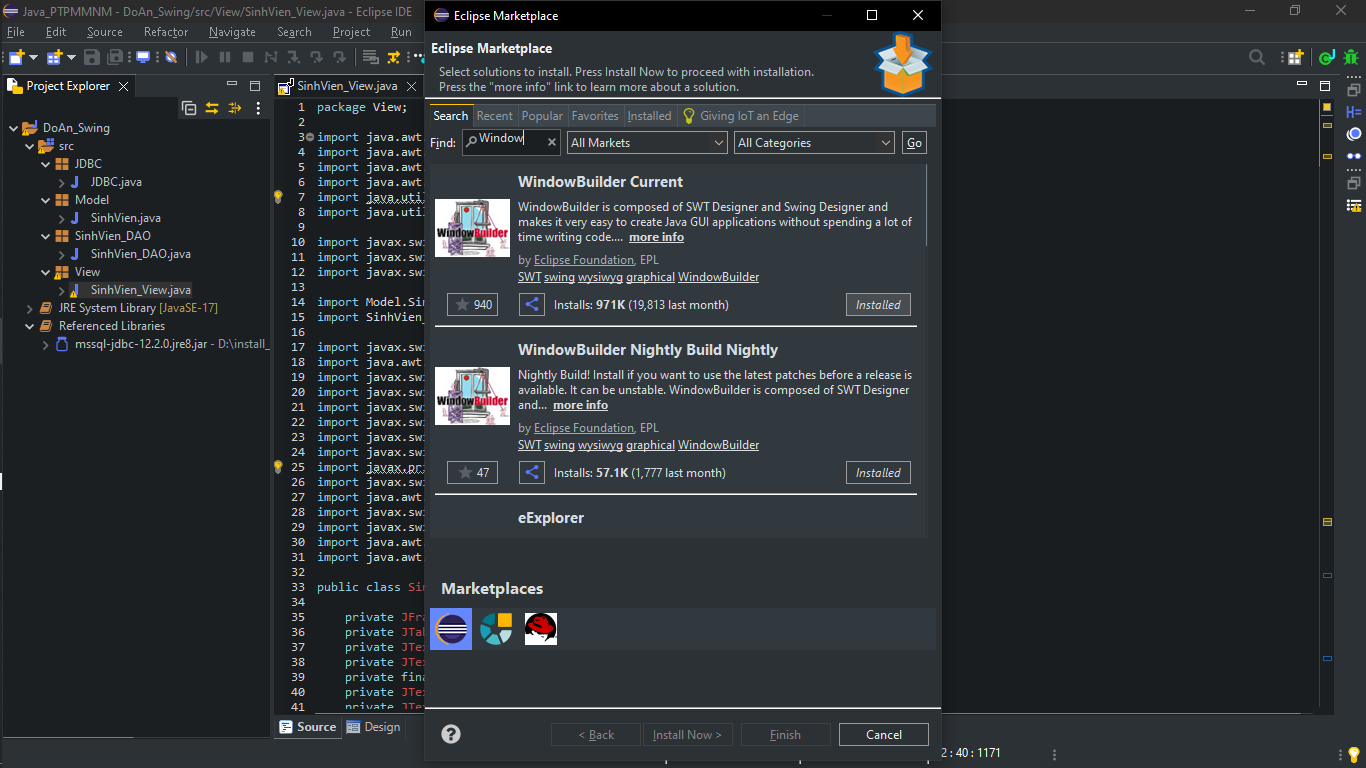
Truy cập đường link: https://www.eclipse.org/downloads/





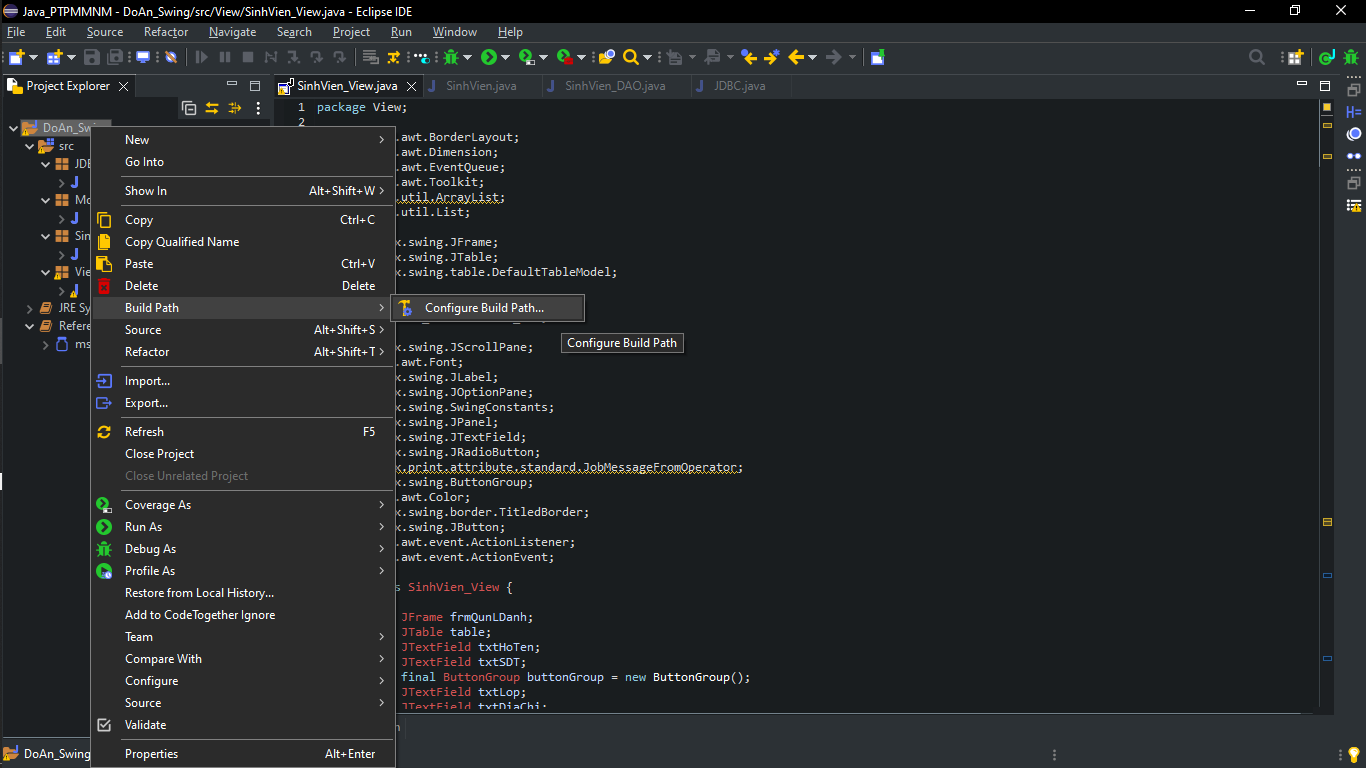
Sau khi hoàn thiện cài đặt IDE thì cài đặt WindowBuilder trong Eclipse

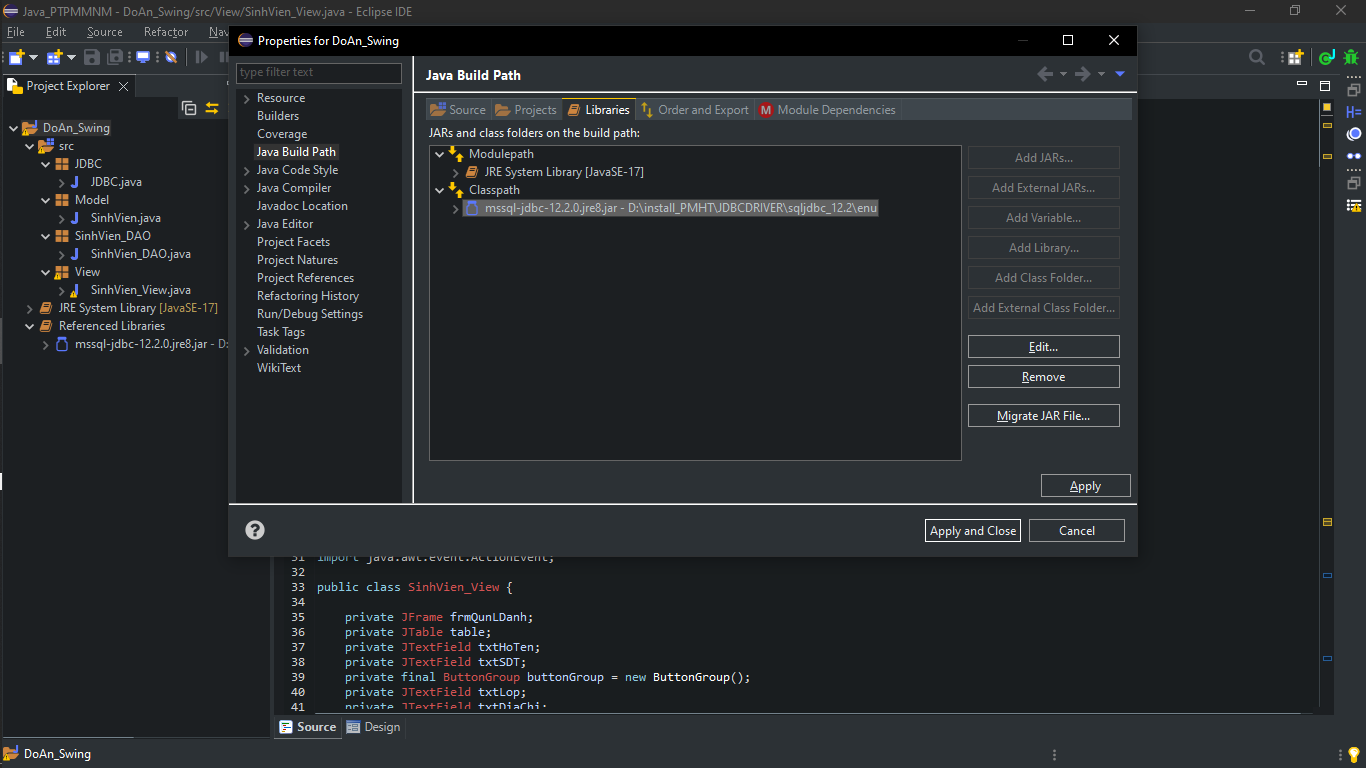




Sau khi hoàn thiện cài đặt. Tạo dự án mới và tiến hành code.

Tải bộ driver jdbc để kết nối java với SQL Sever

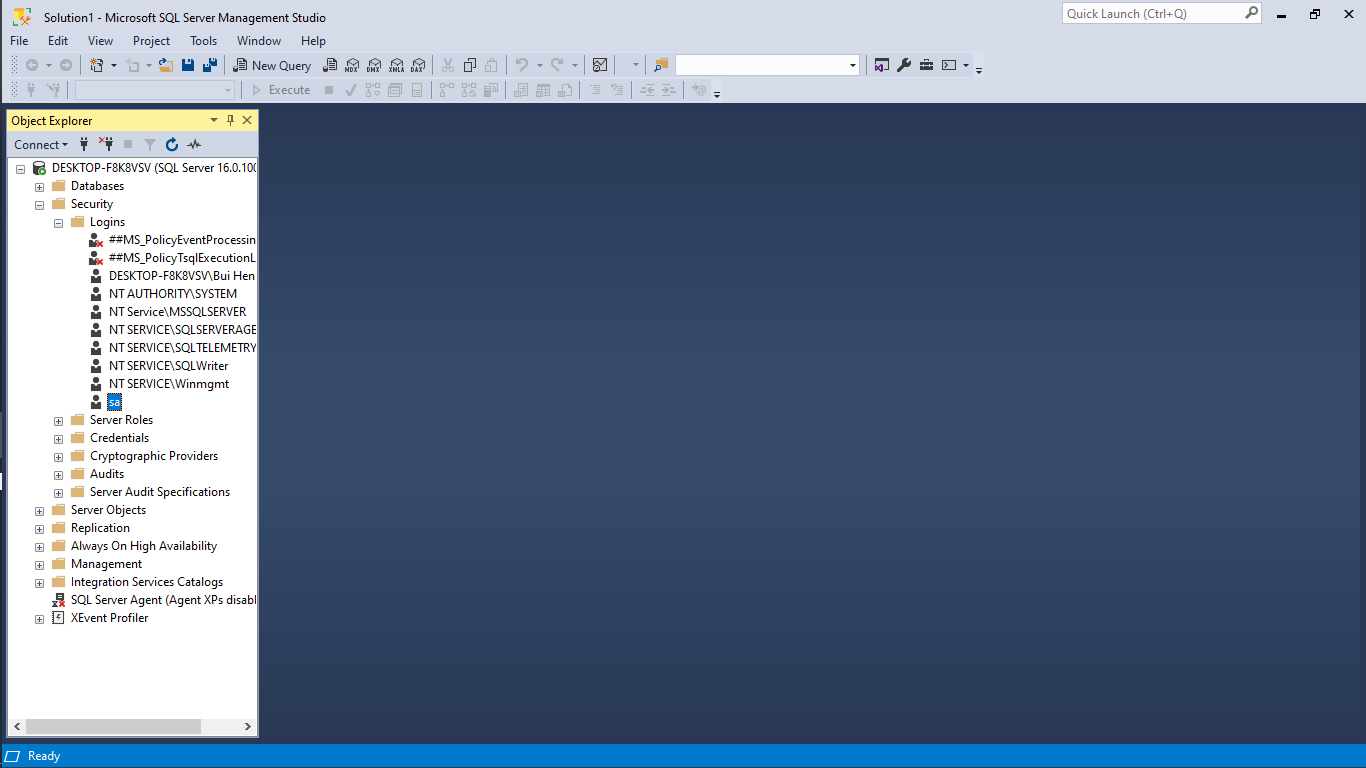




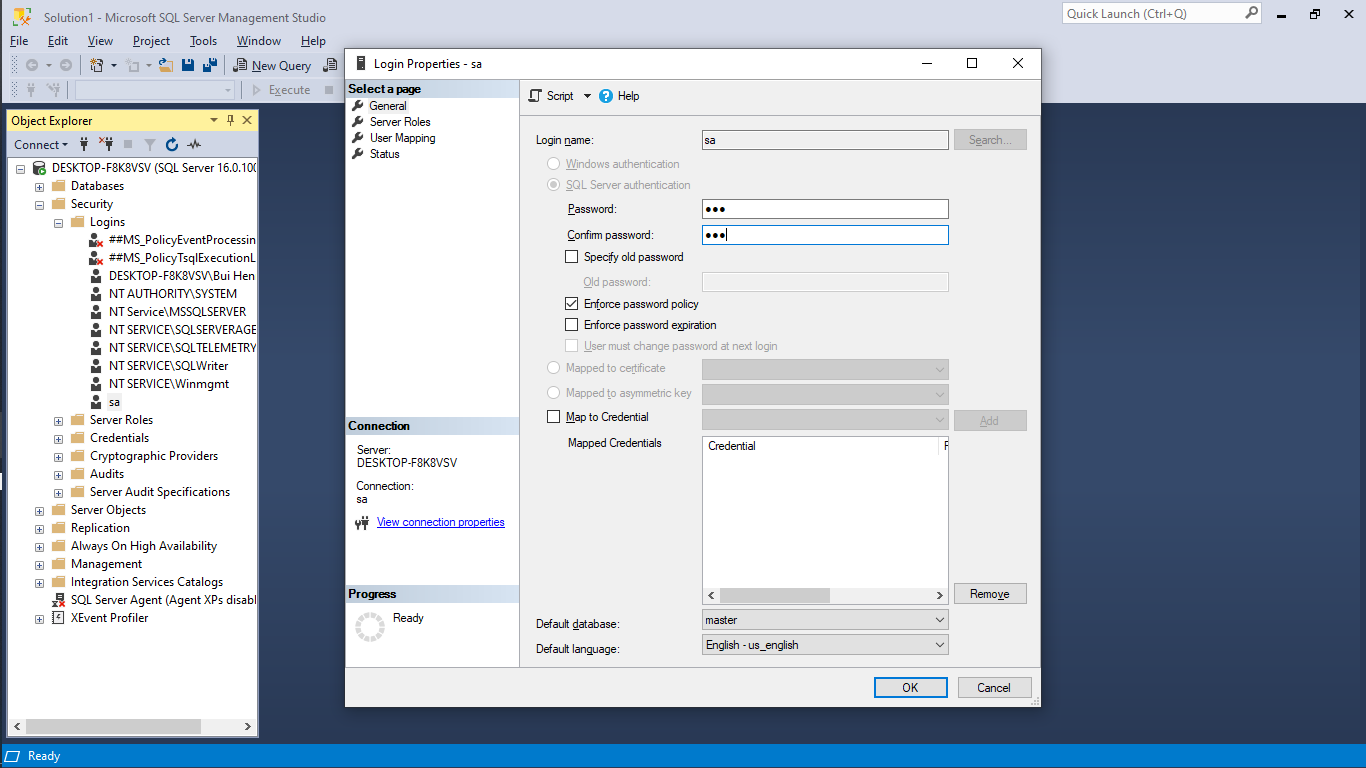
Sau khi add bộ thư viện, ta có thể kế nối Java với SQL Sever

**4.2 DATABASE**

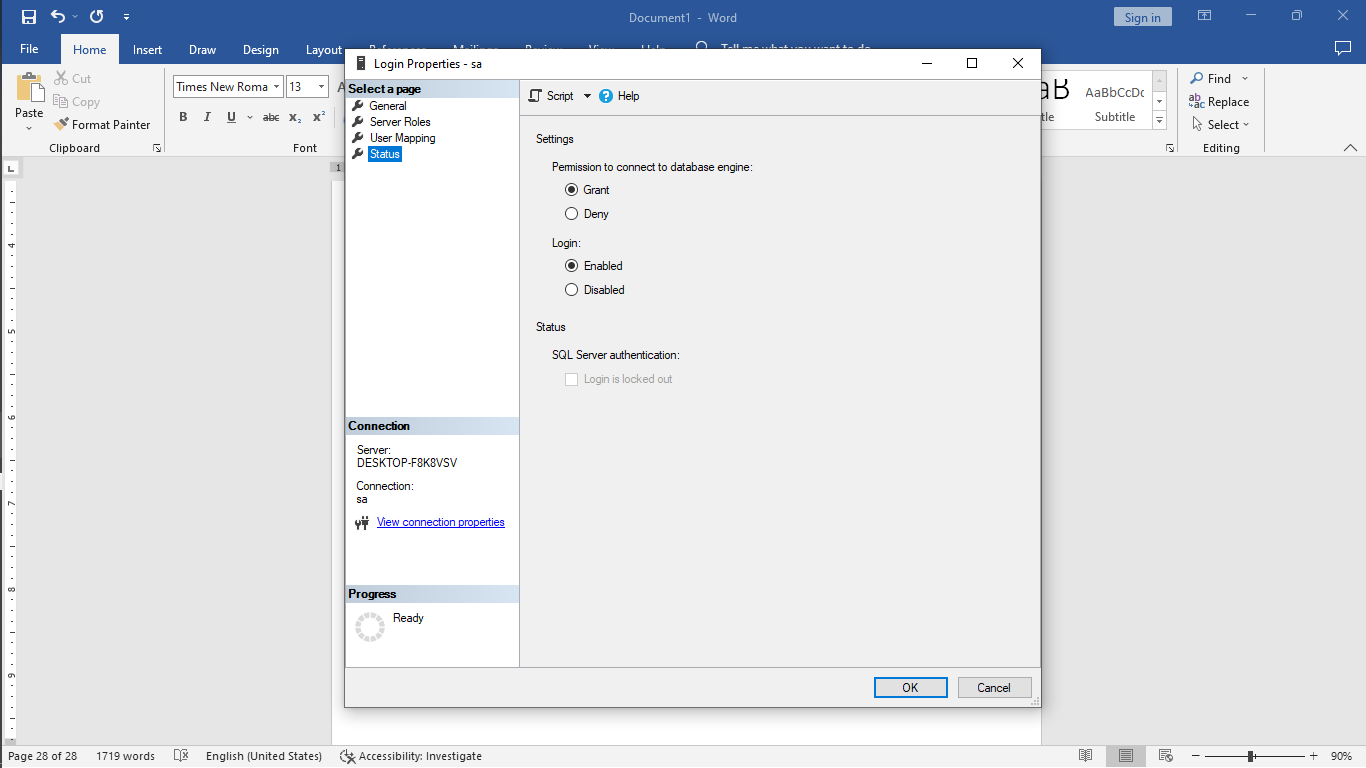
Thiết lập tài khoản sa và cài đặt mật khẩu

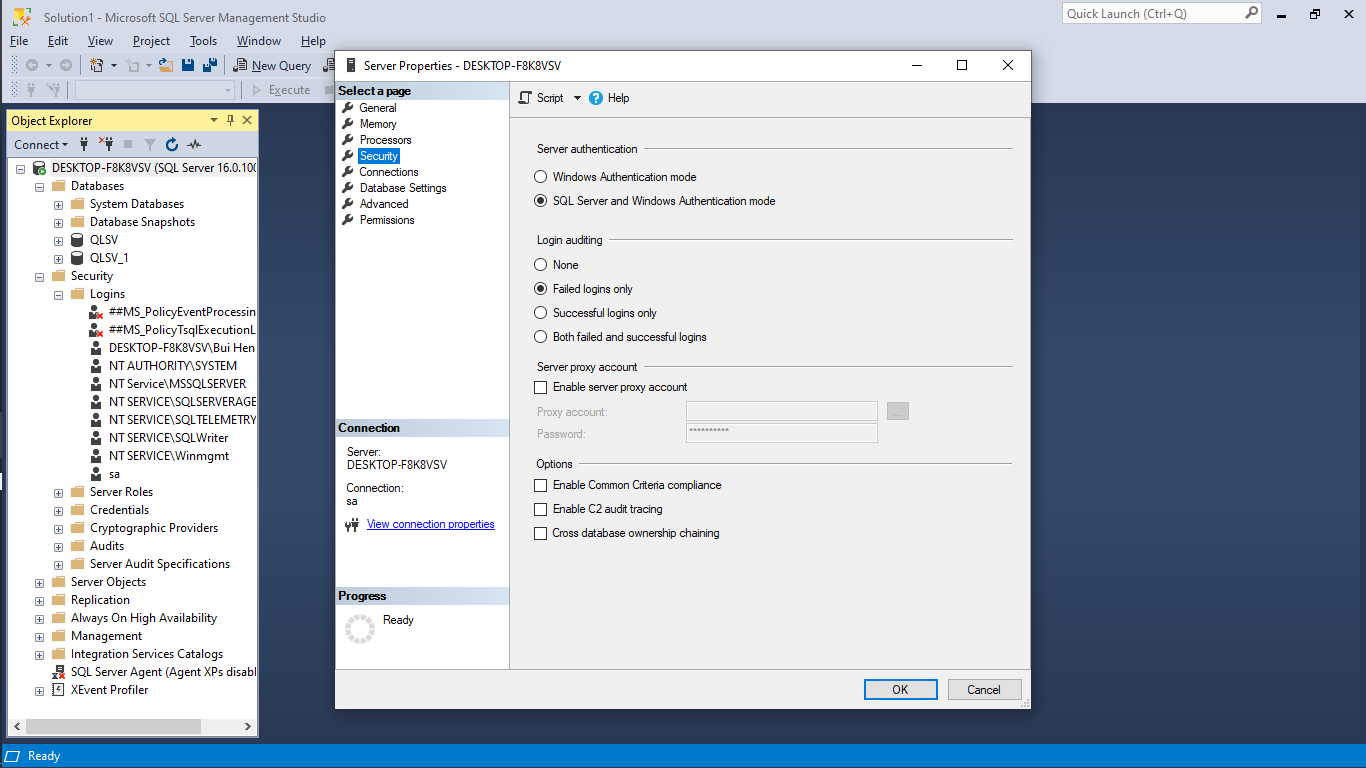
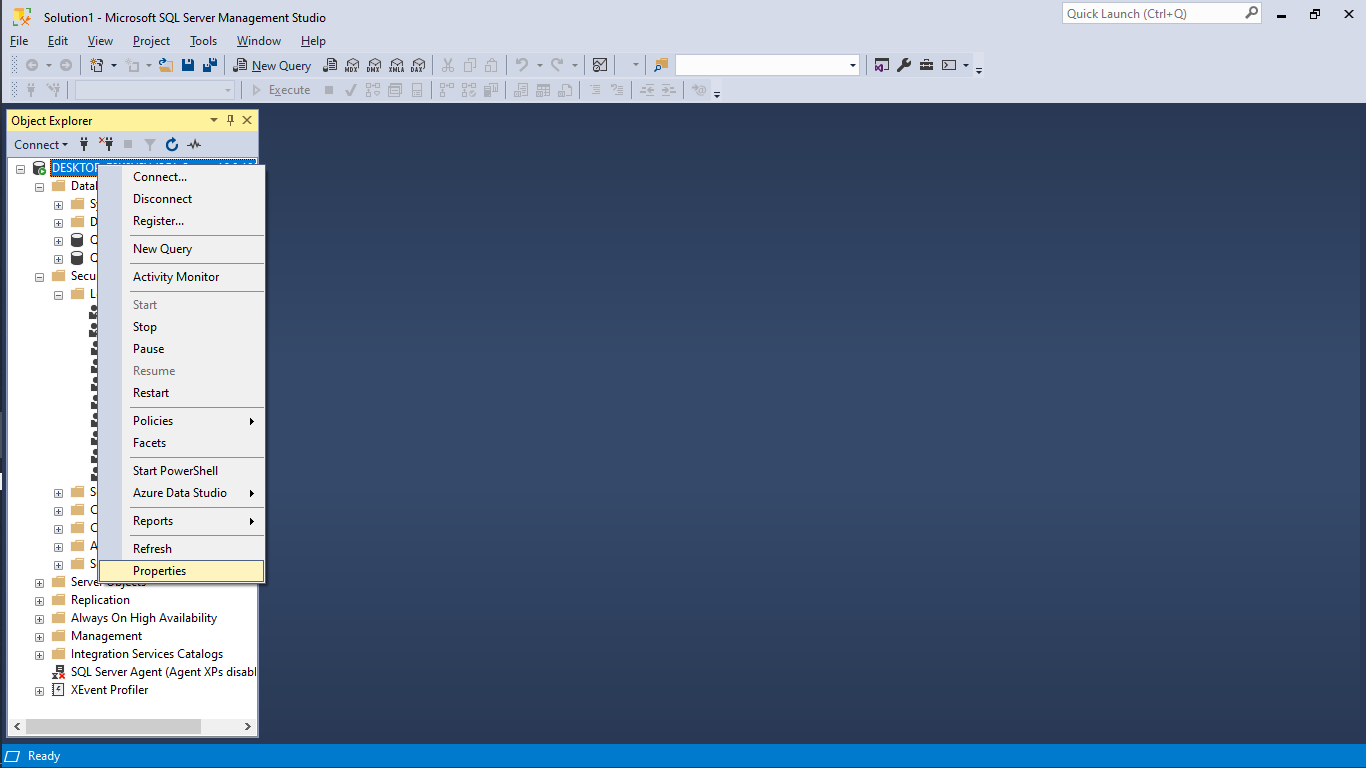


Chuột phải chọn Properties

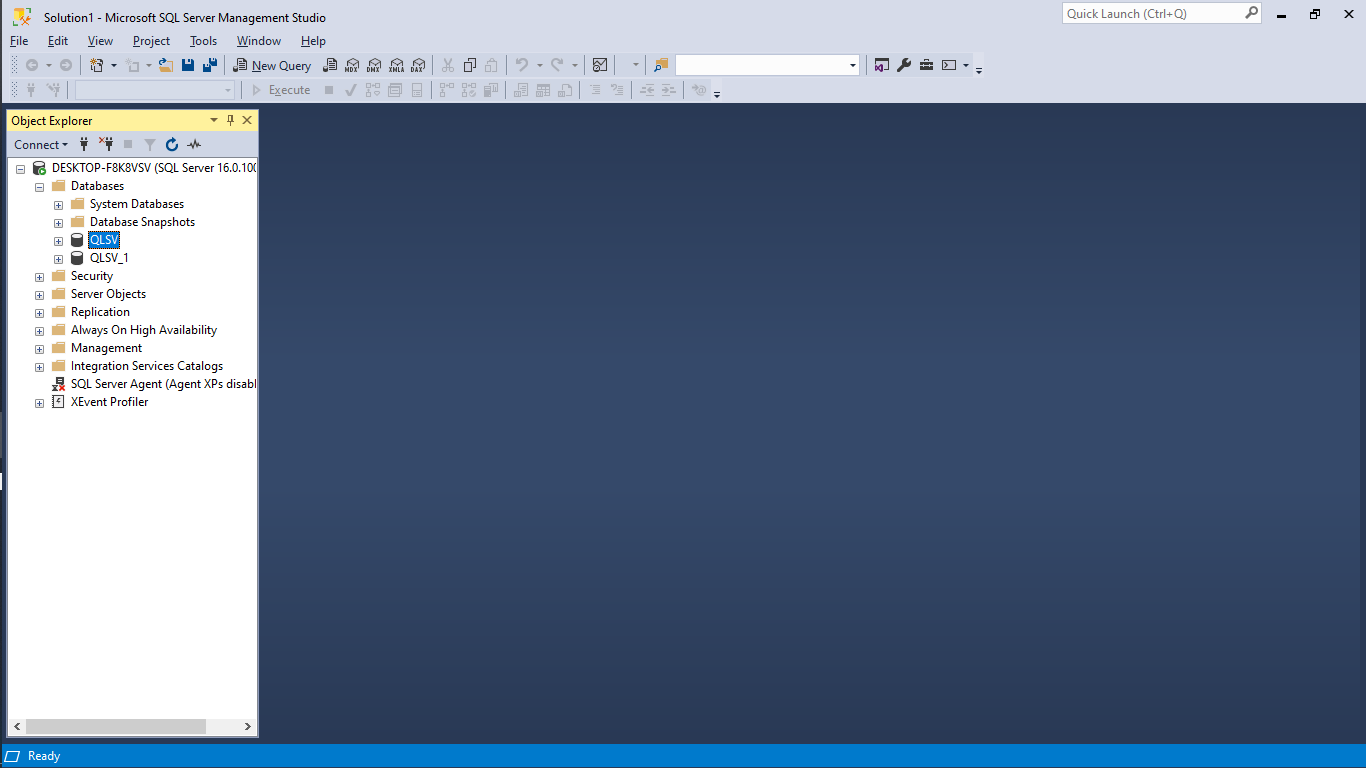
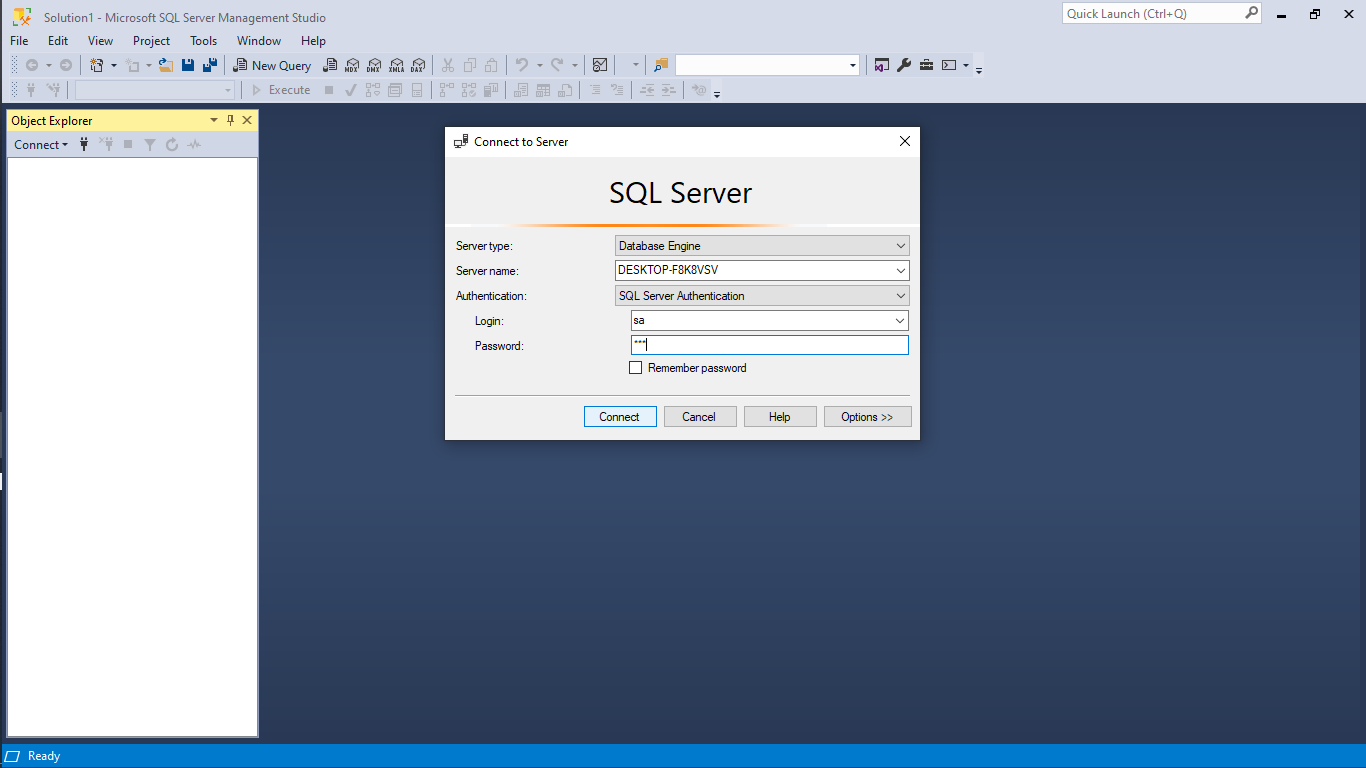


Cài đặt mật khẩu bạn muốn cài

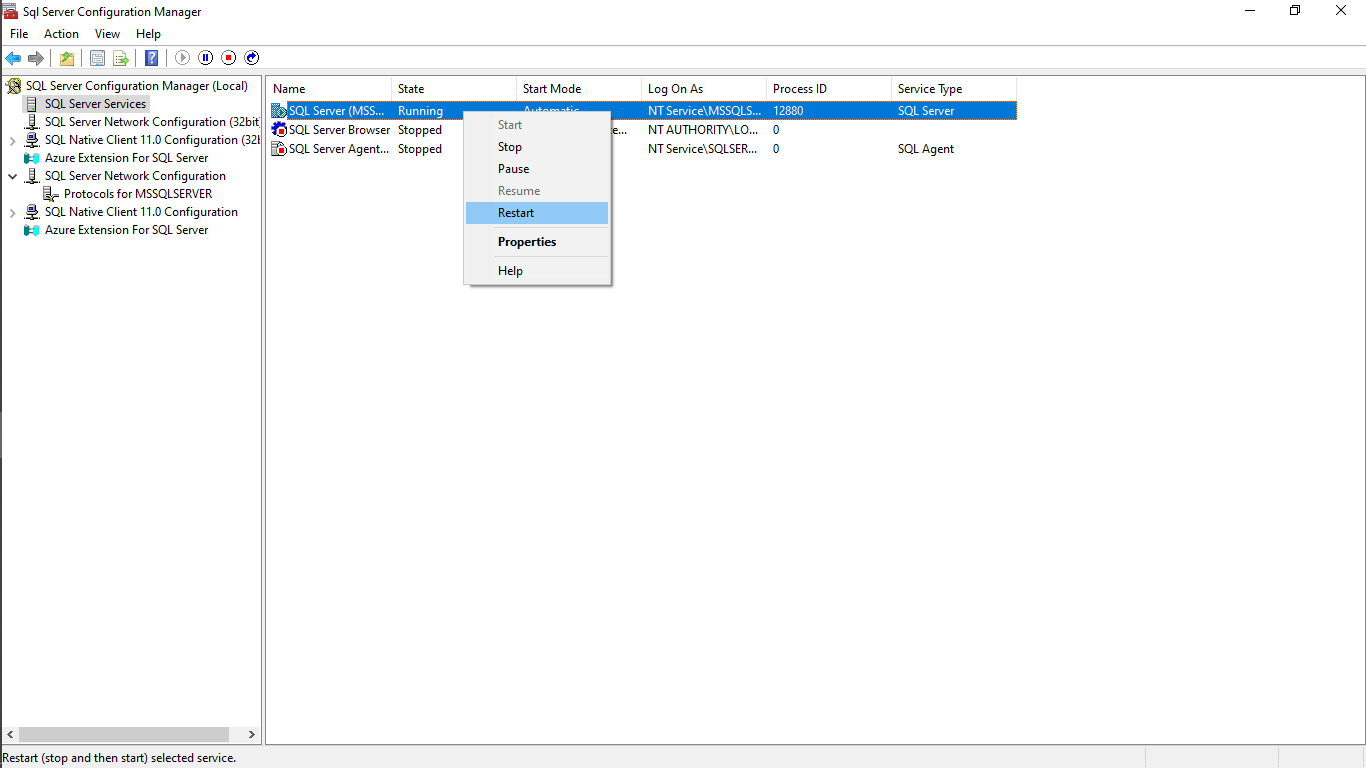




Sau khi cài đặt quá trình này xong, ta hủy kết nối và kết nối lại với Database

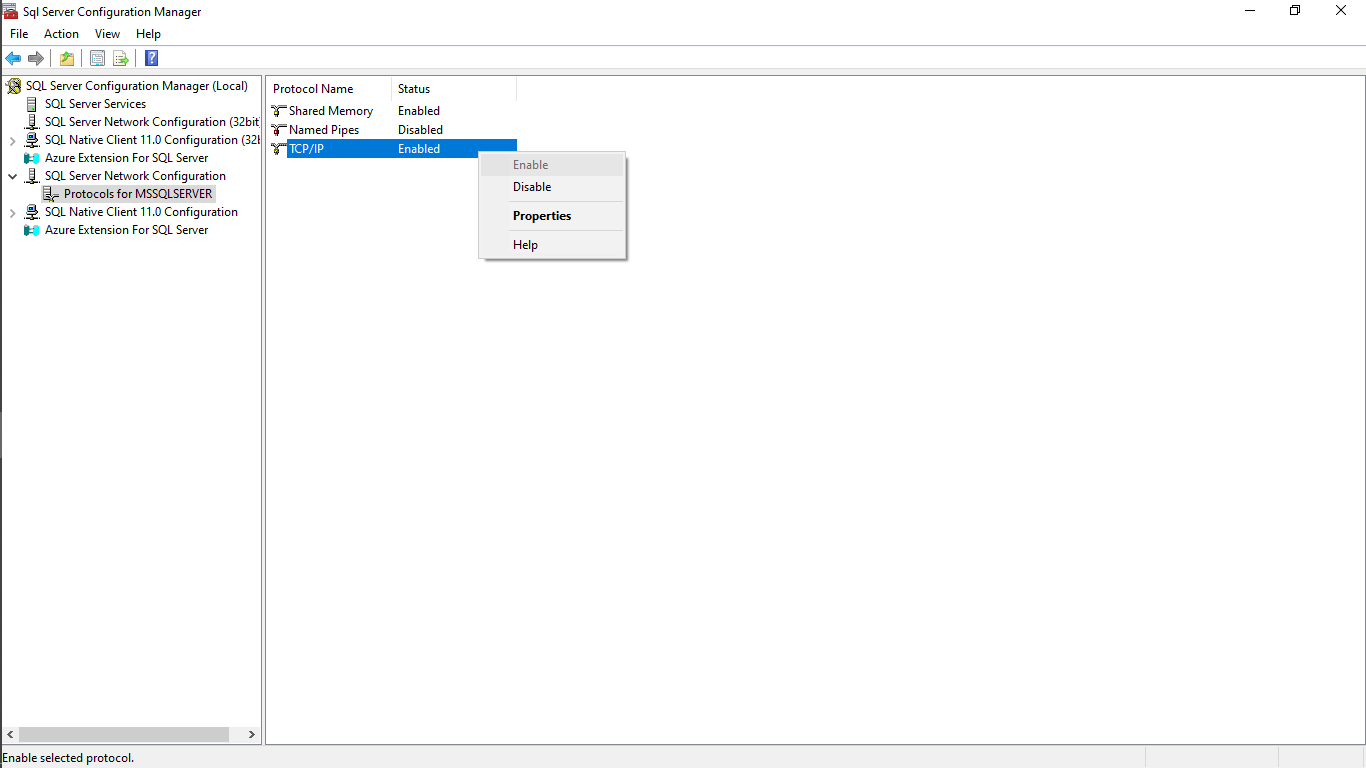


Trường hợp có lỗi thì khởi động lại SQL Sever bằng SQL Server 2022 Configuration Manager

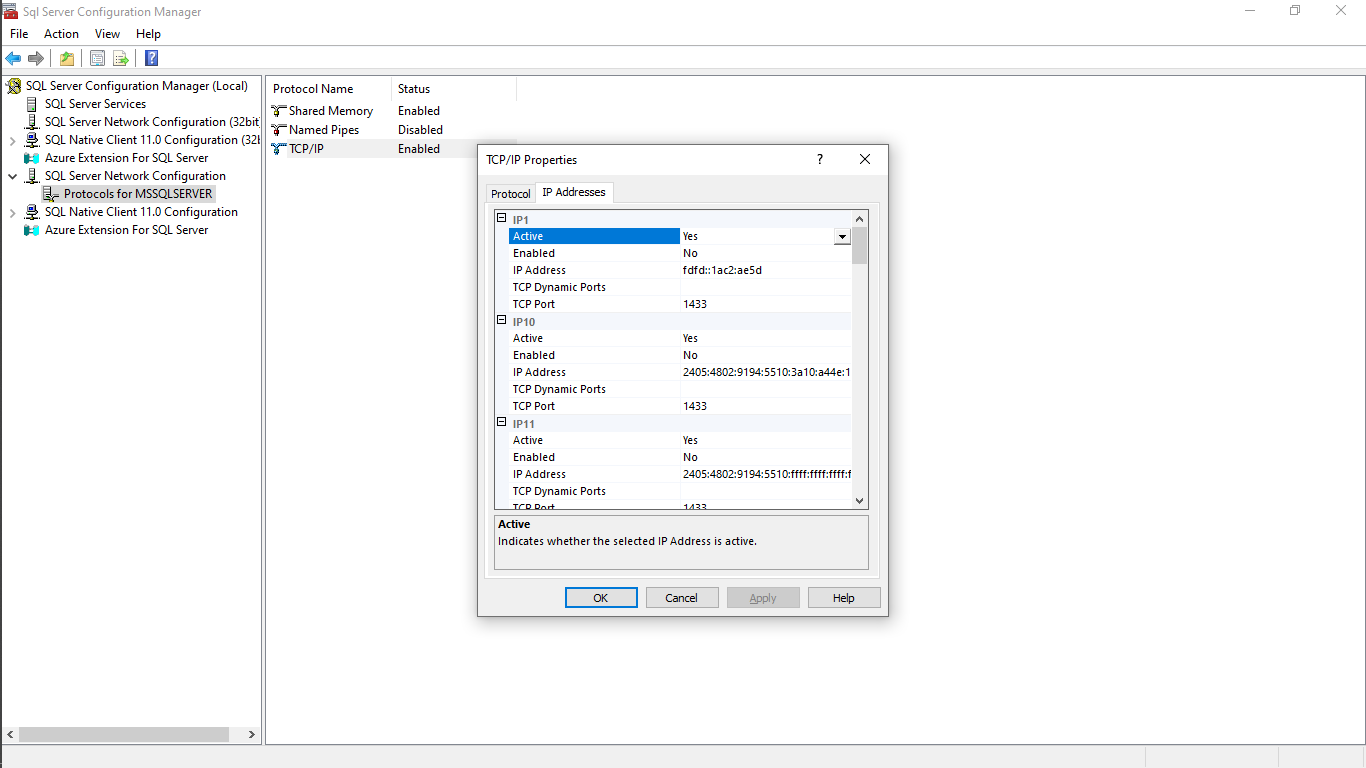


Click nút restart

Lúc này Java có thể chưa kết nối được với SQL Sever do chưa nhận được port 1433 từ SQL Sever. Ta phải kích hoạt port bằng cách:



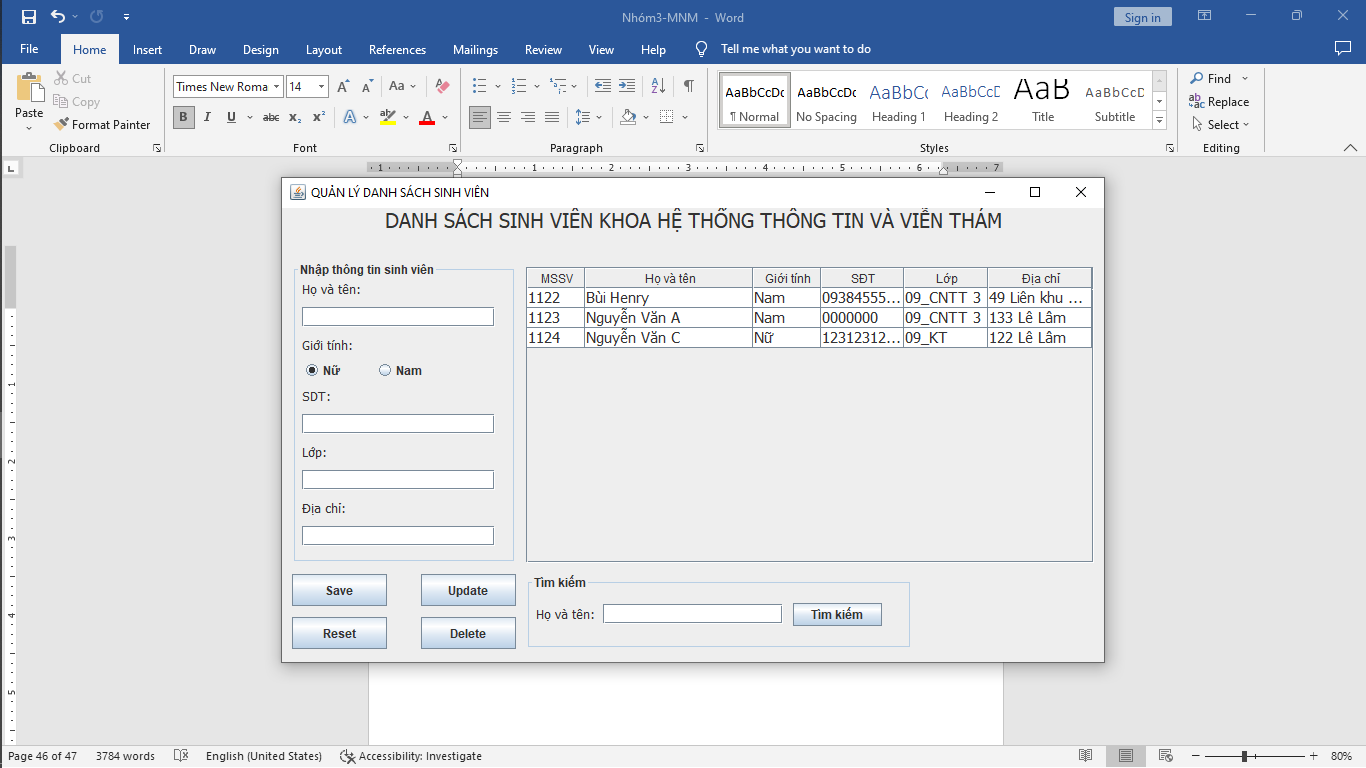
Nhập các trường TCP Port là 1433



=> Hoàn thành Java kết nối với SQL Sever

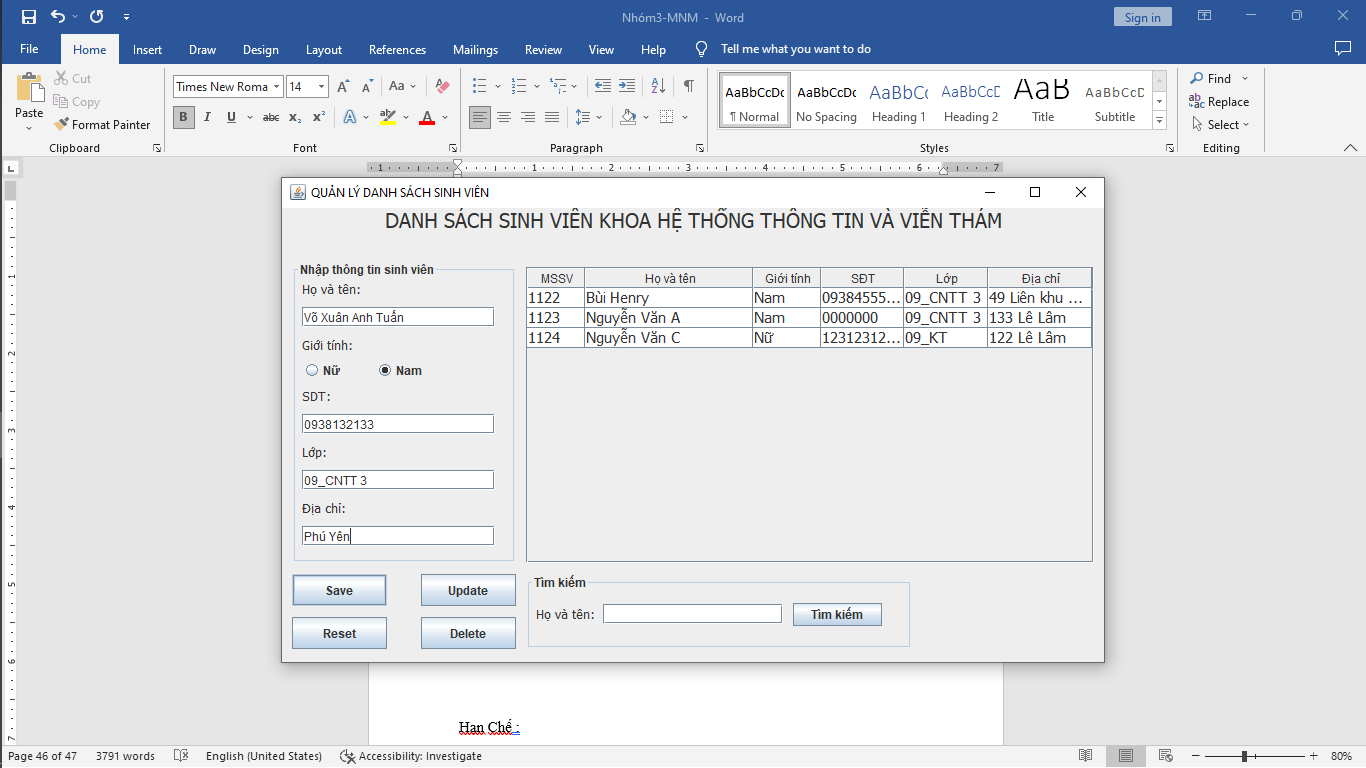
# **Phần 5. Kết quả đạt được**

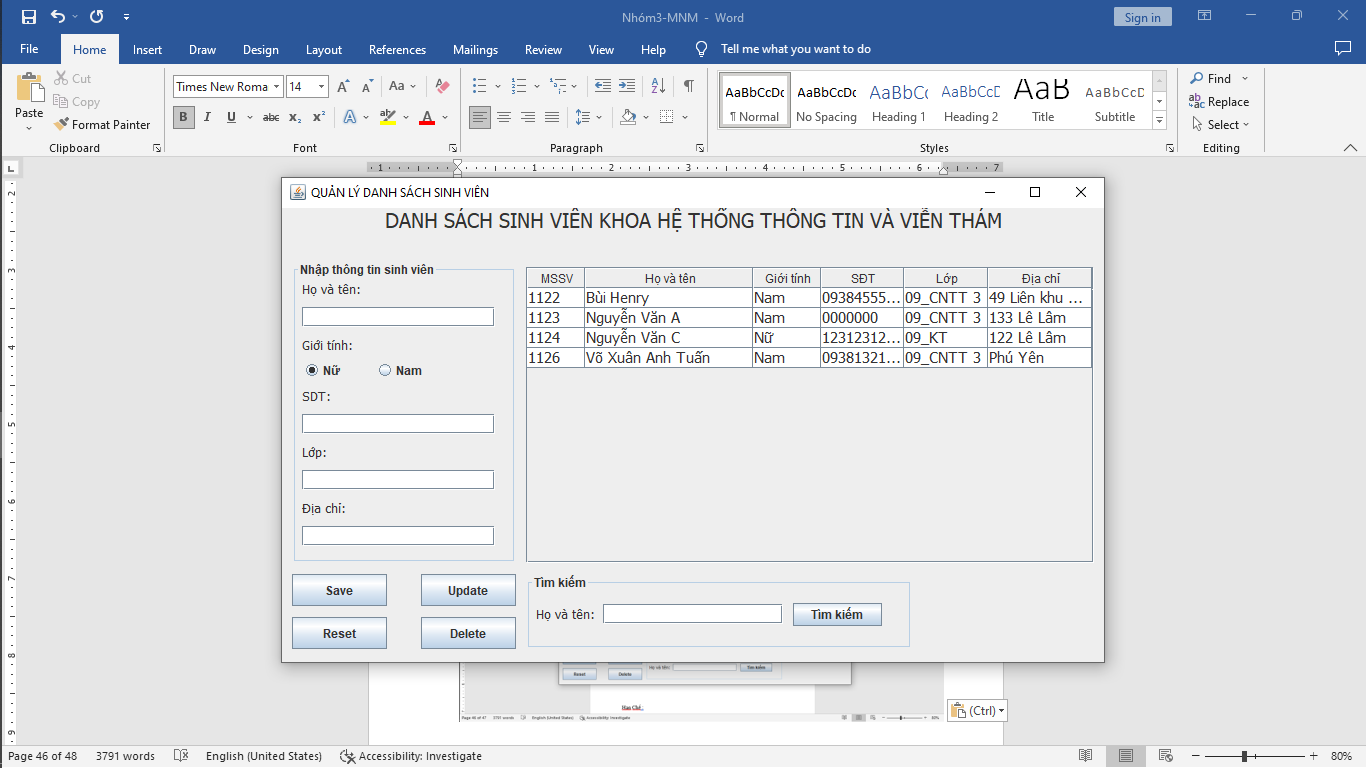
**5.1 Giao diện ứng dụng**

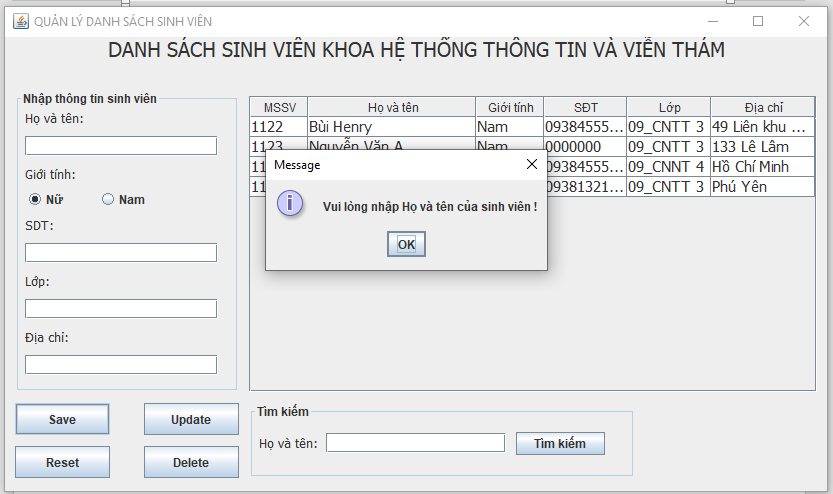


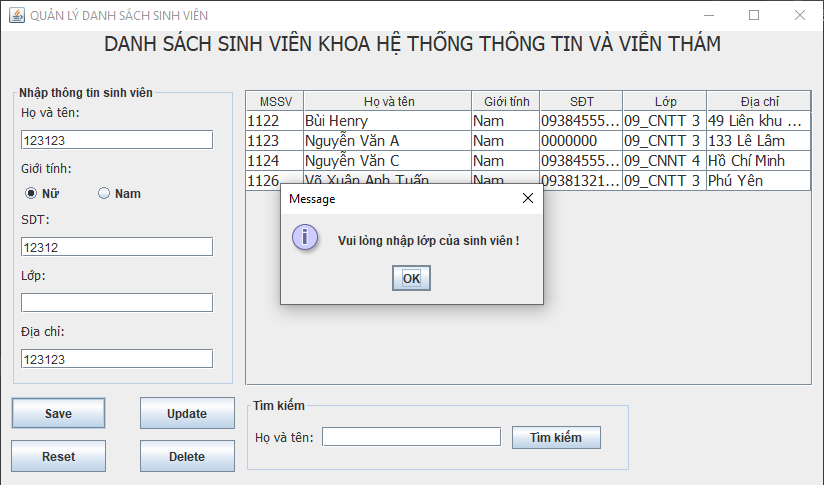
**5.2 Các phím chức năng**

**5.2.1 Save**

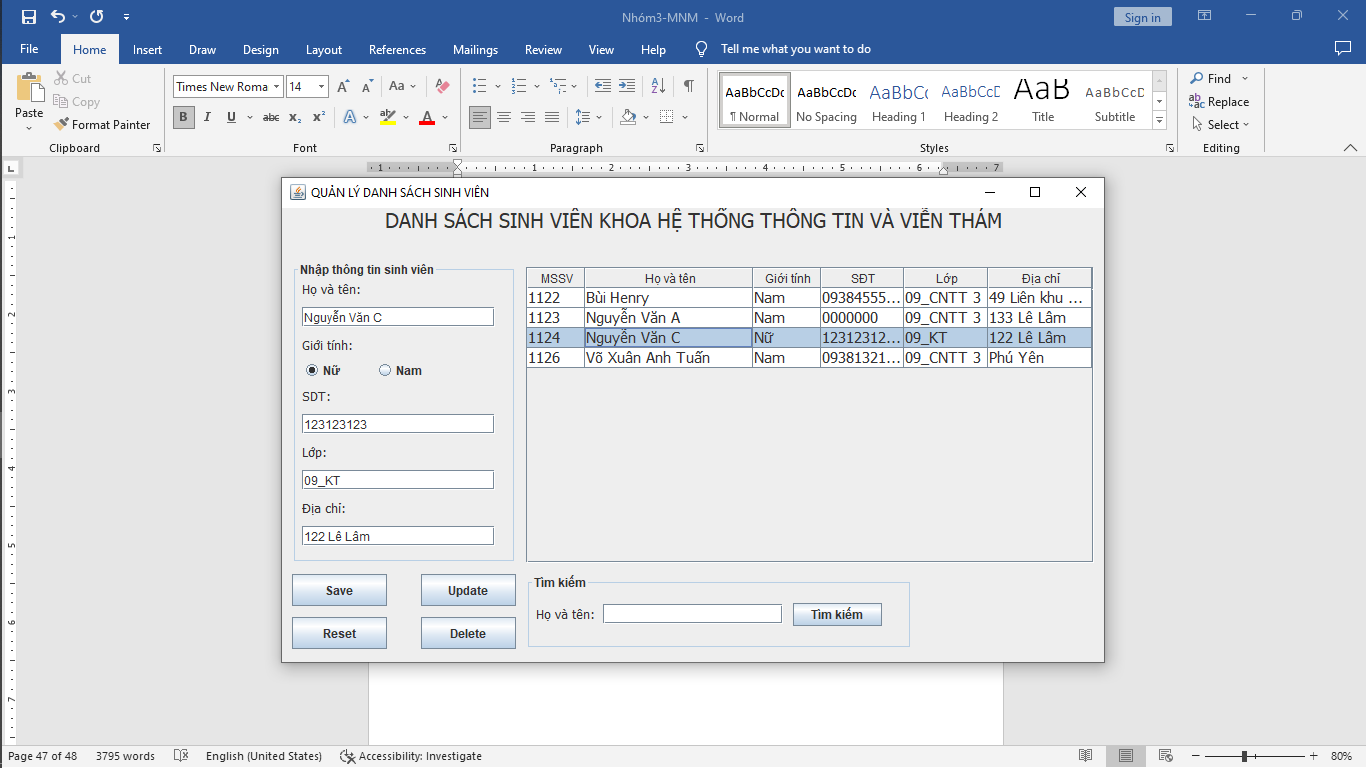


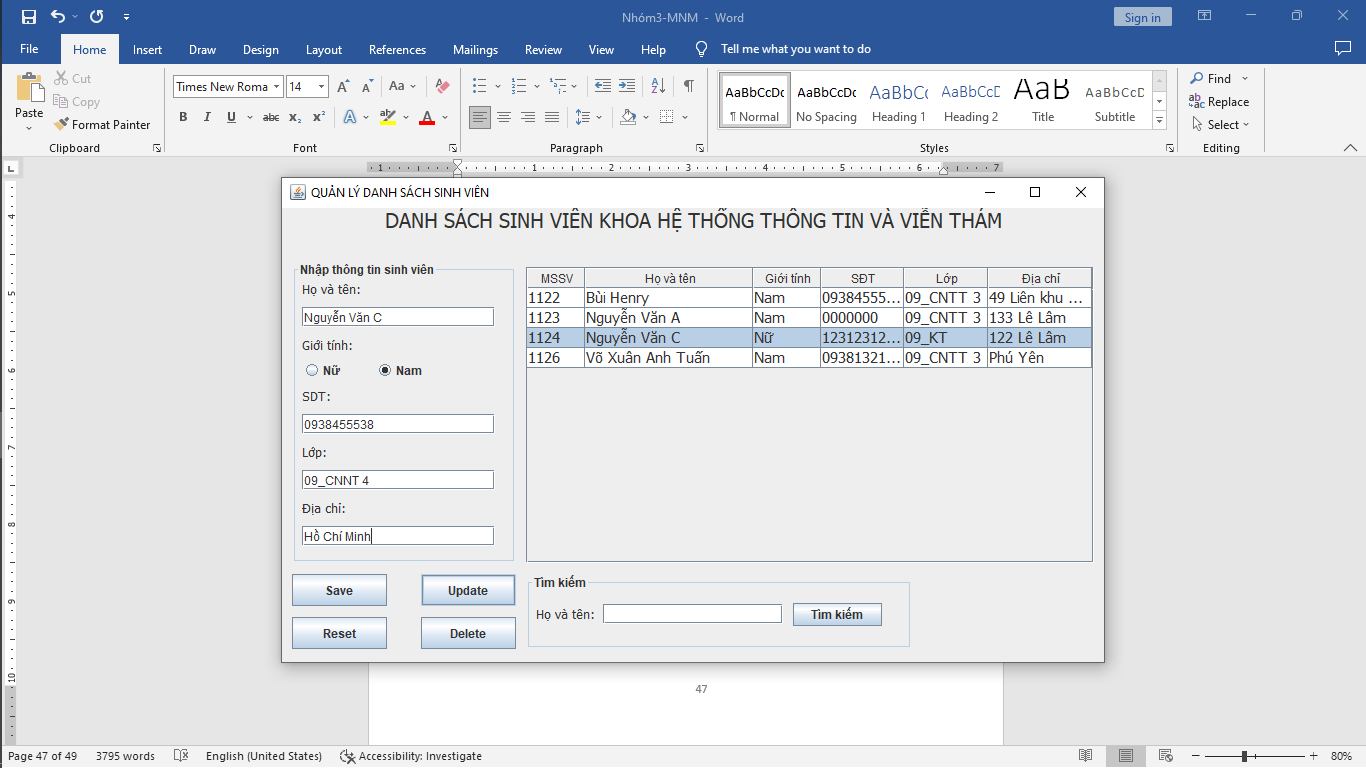


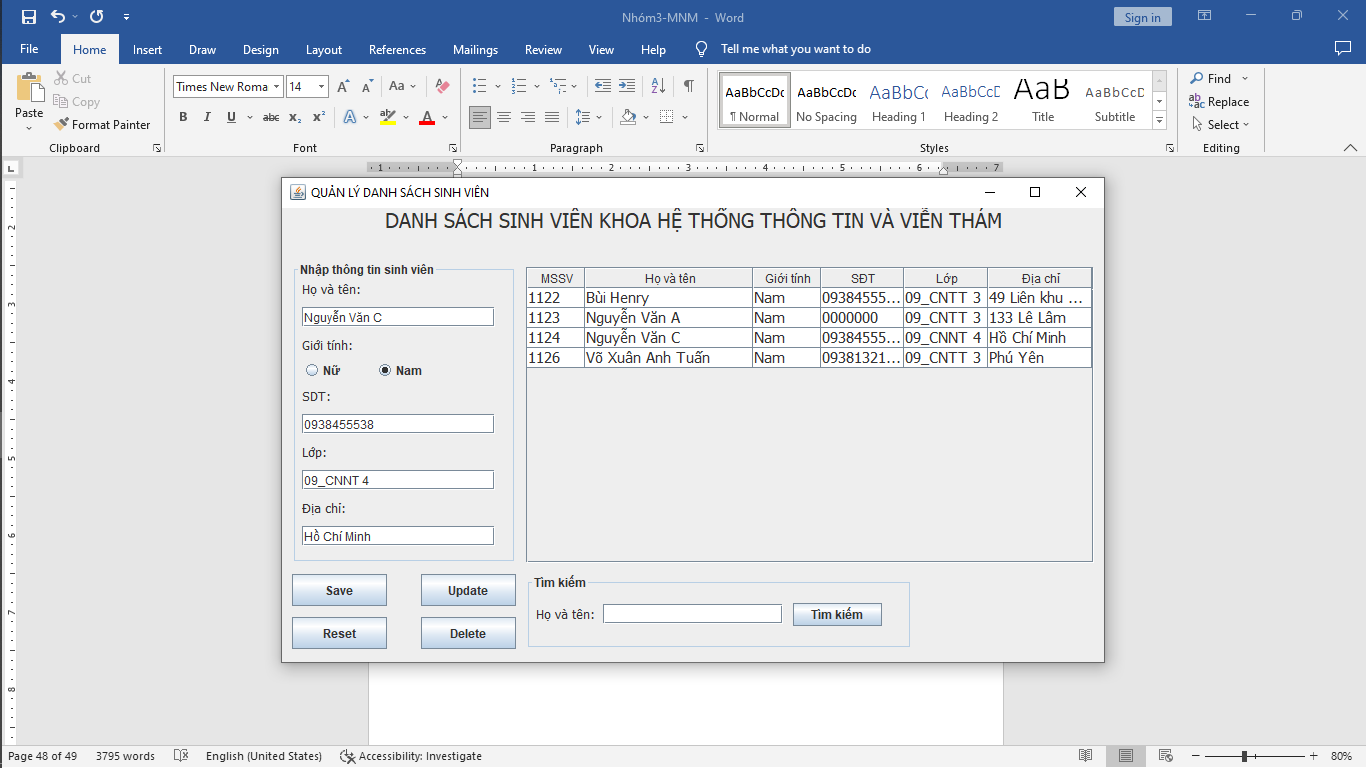
****

****

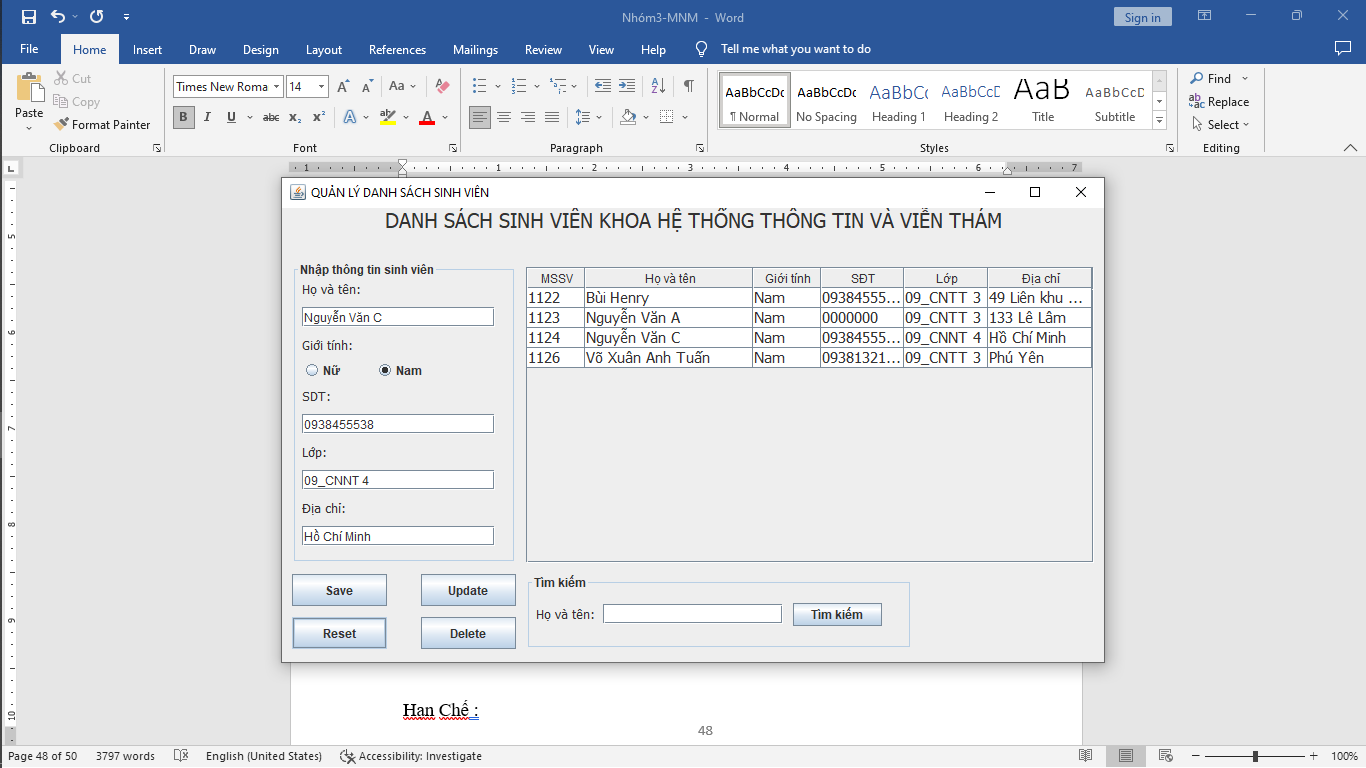
**5.2.2 Chức năng Update**

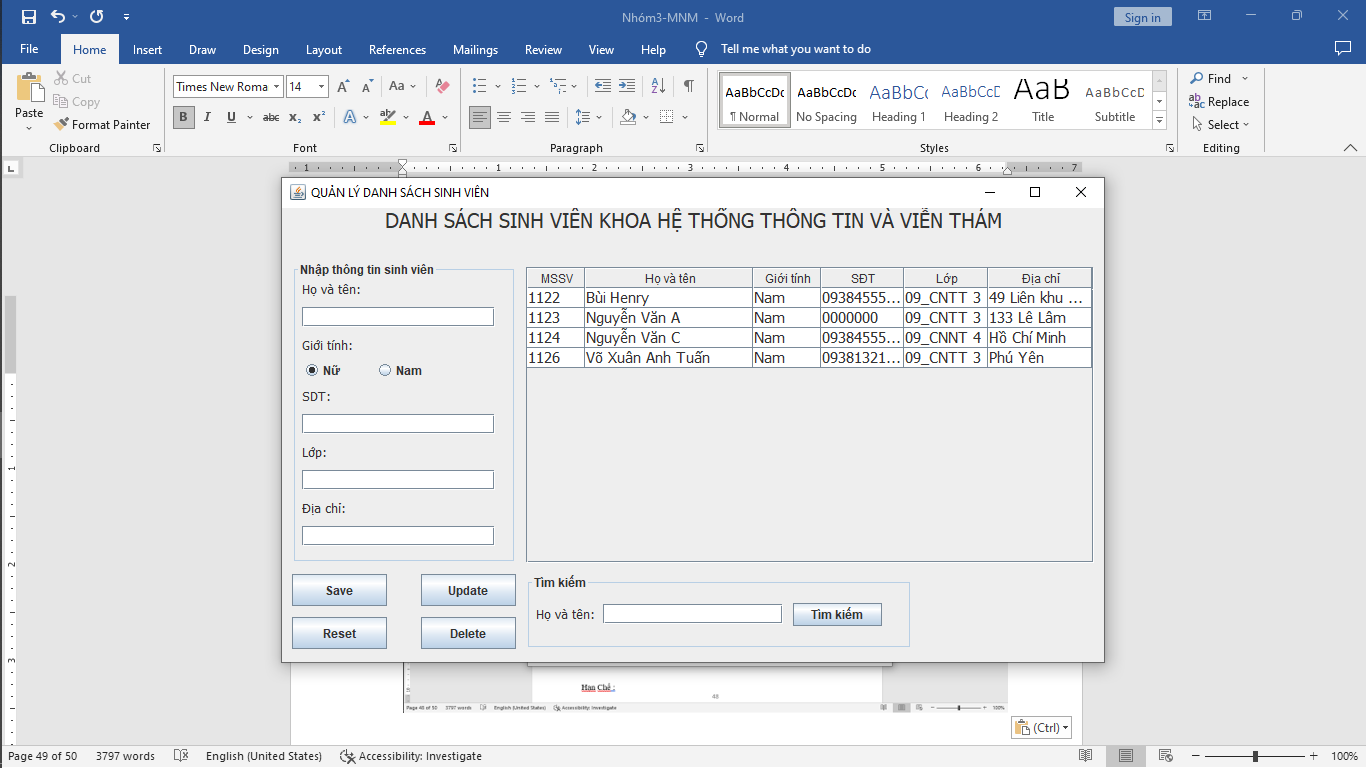




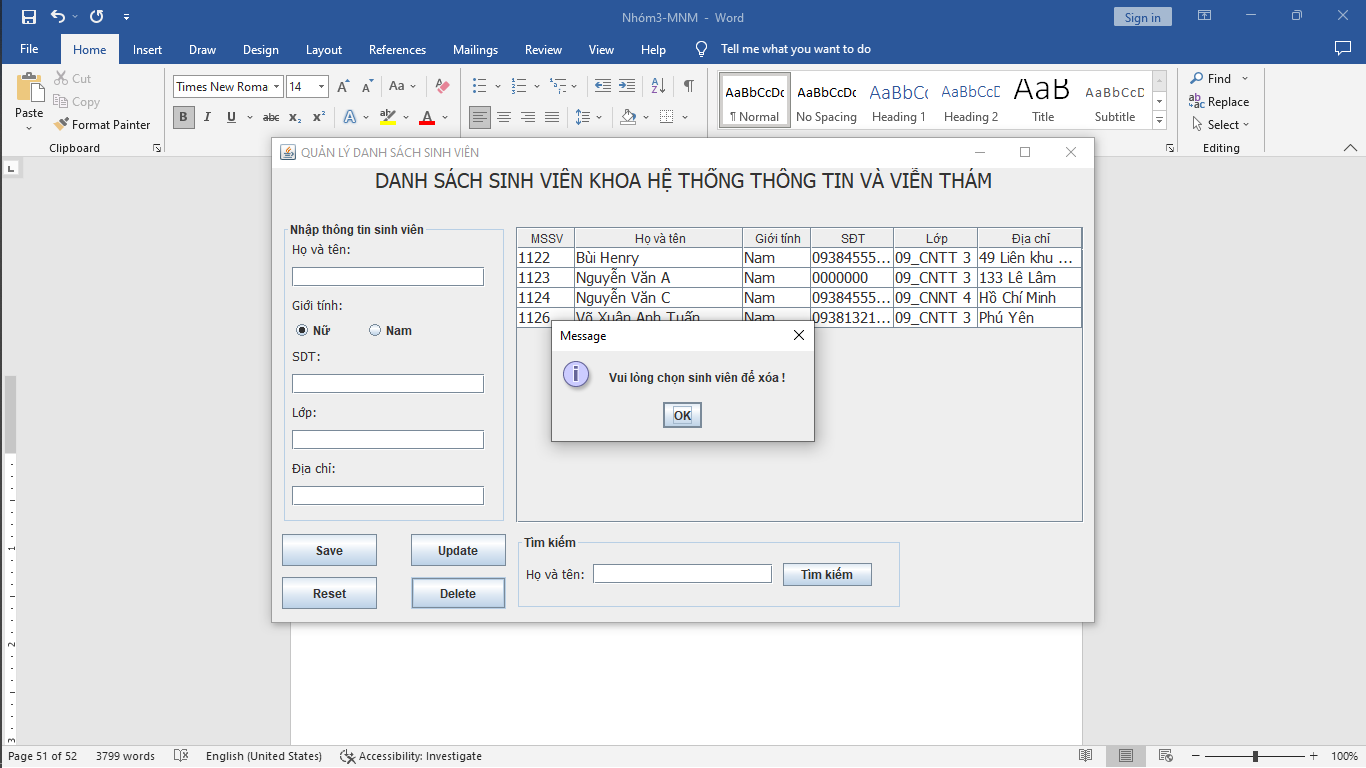


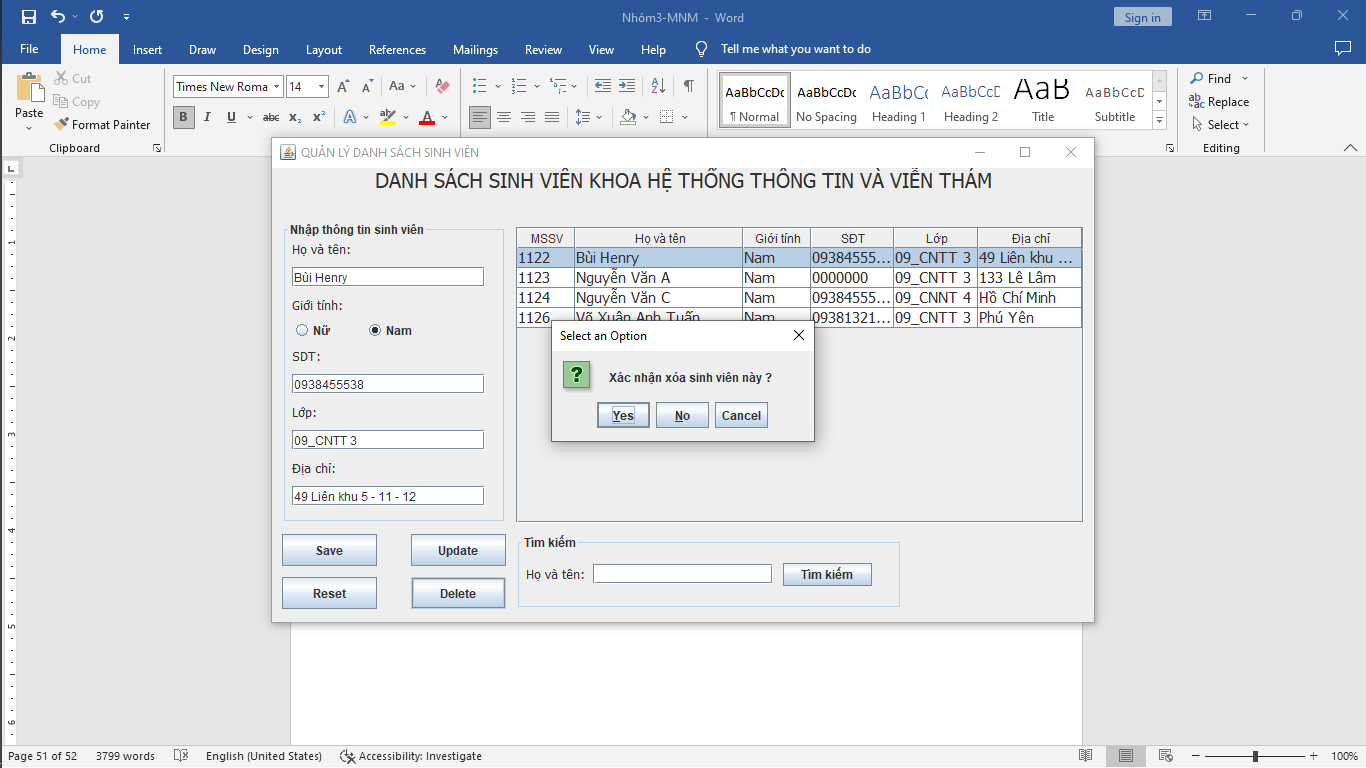
**5.2.3 Reset**

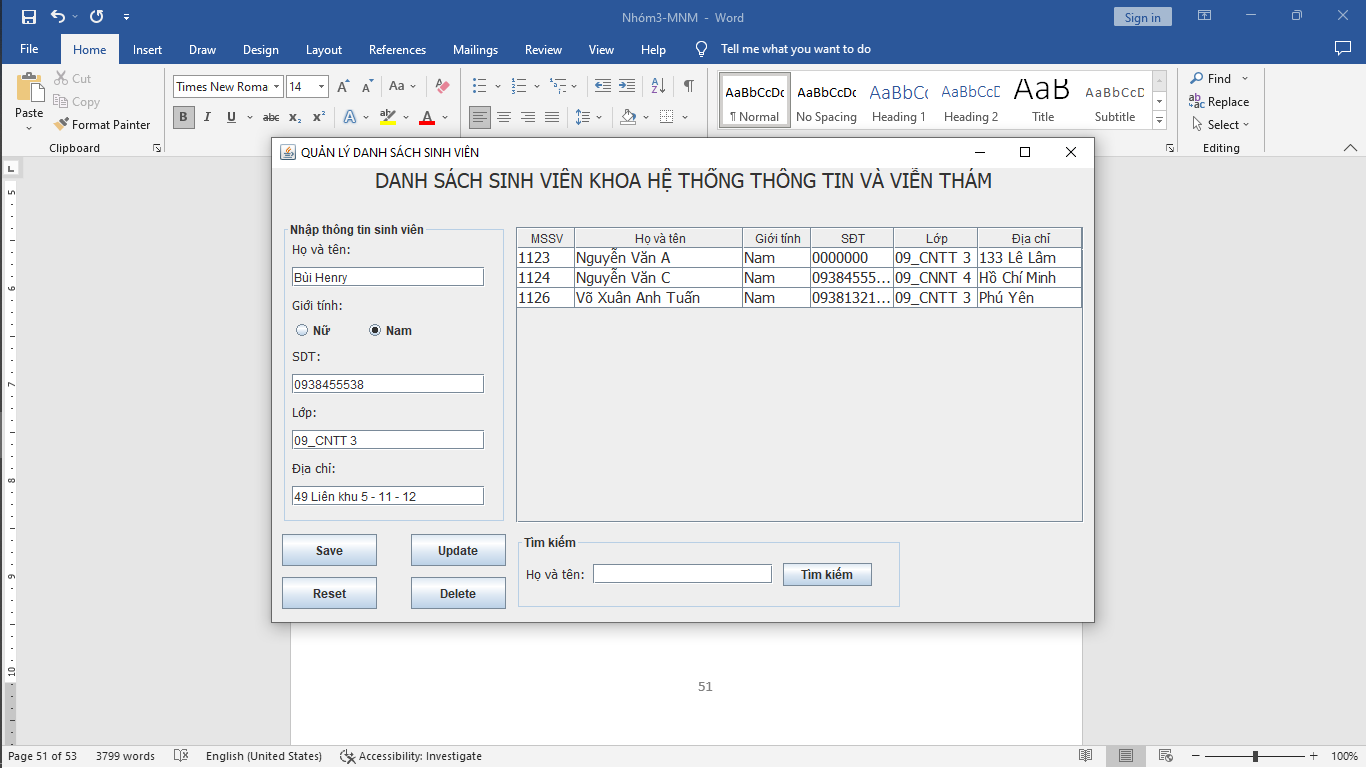




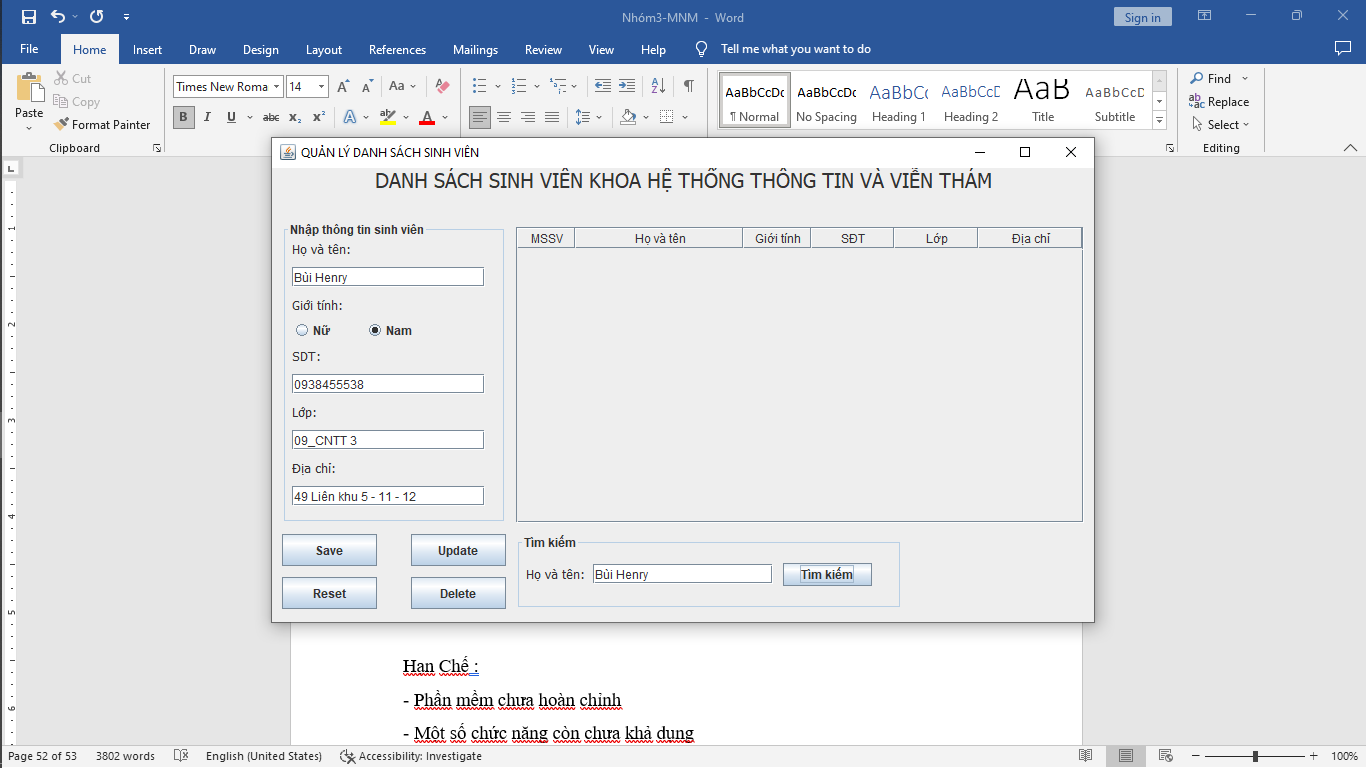
**5.2.4 Delete**

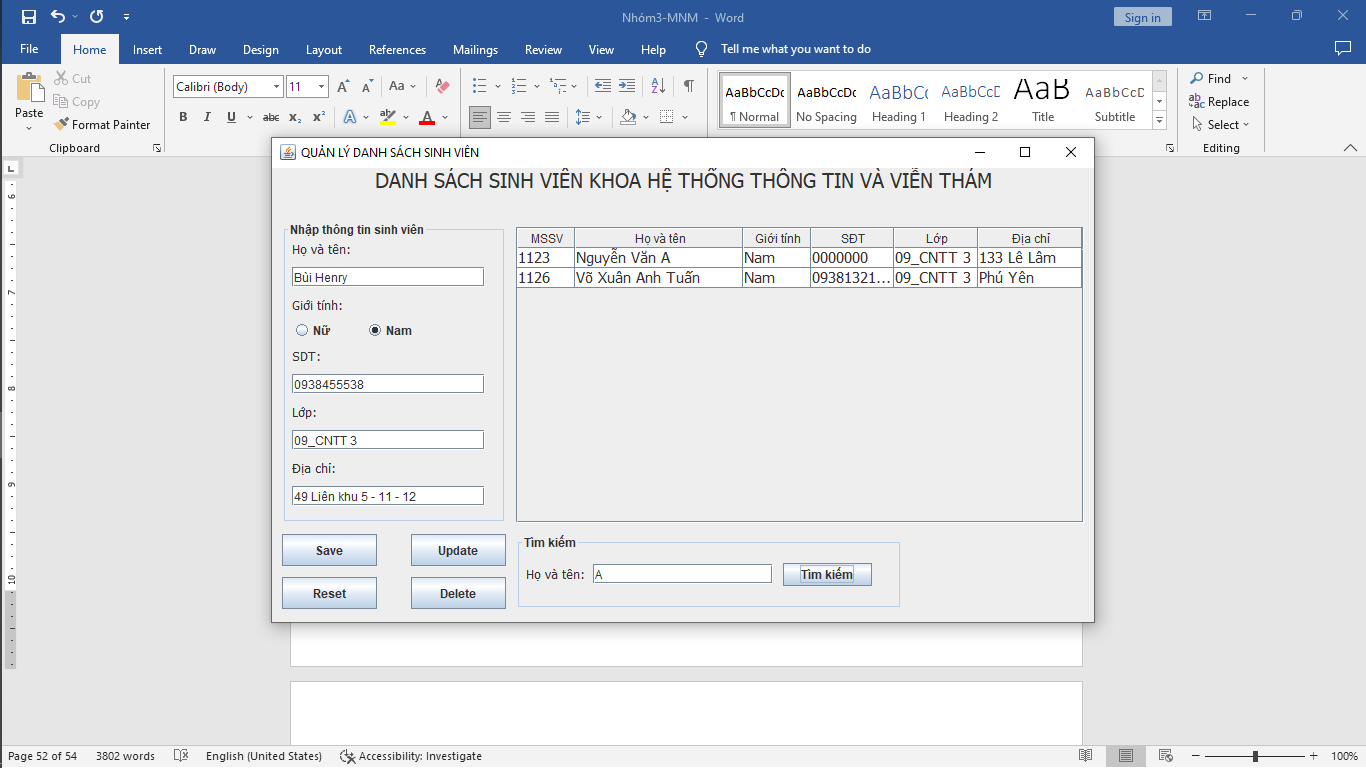






**5.2.4 Tìm kiếm**





Hạn Chế :

- Ít tính năng

- Không có tính bảo mật vì không có cấp quyền truy cập ứng dụng

- Không thể cập nhật điểm số

- Chỉ có tìm kiếm theo tên, nếu trùng tên sẽ khó tìm kiếm

- Không có tính năng sắp xếp dữ liệu

- Kiến thức nghiệp vụ chưa được sâu rộng

- Kiến thức về ngôn ngữ còn nhiều hạn chế

# **Phần 6. Kết luận**

Trải qua 60 tiết học lý thuyết và thực hành đã giúp chúng em có kiến thức nền về môn học "Phát triển phần mềm mã nguồn mở" giúp bản thân mỗi người trong nhóm em cải thiện bản thân và hiệu suất làm việc , biết thêm nhiều về công nghệ , công cụ mới . Đây là hành trang thực sự có ích với em trong sự nghiệp của một Kỹ sư phần mềm sau này.

Về phần project , tuy có một số thời điểm , tiến trình công việc bị deplay vì khó khăn trong vấn đề cân bằng giữa việc học của từng môn nhưng giảng viên bộ môn đã hỗ trợ em rất nhiệt tình để em có thể hoàn thành project .

Xin thay mặt các thành viên trong nhóm chân thành cảm ơn sự giảng giải và giúp dỡ của Giảng viên bộ môn Thầy Trần Văn Định đã cung cấp kiến thức một cách cô động và đầy đủ cũng nhưng là đưa ra ý kiến về phần chọn phần mềm để bọn em có thể hoàn thành project . Thầy đã luôn theo dõi và hỏi han về phần bài làm để có thể giúp chúng em giải quyết những khúc mắc , khó khăn trong việc hoàn thành dự án nhỏ . Một lần nữa em xin gửi lời cảm ơn chân thành đến Thầy.