**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

**======\*\*\*======**

****

**BÁO CÁO BÀI TẬP LỚN**

**LẬP TRÌNH WINDOWS**

**ĐỀ TÀI: HỆ THỐNG QUẢN LÝ ĐOÀN VIÊN**

|  |  |
| --- | --- |
| **Giảng viên**: | Thầy Phạm Văn Hà |
| **Nhóm - Lớp**: | Nhóm 25 – KTPM2 – K11 |
| **Thành viên**: | Nguyễn Văn Trường |
|  | Ngô Tiến Lâm |
|  |  |

Hà Nội, năm 2018

**Mục lục:**

[**Phần 1: KHẢO SÁT VÀ PHÂN TÍCH BÀI TOÁN** 4](#_Toc532254247)

[**1.** **Khảo sát bài toán** 4](#_Toc532254248)

[**2.** **Hiện trạng về nghiệp vụ hệ thống** 4](#_Toc532254249)

[**2.1.** **Quản lý đoàn viên về tổ chức** 4](#_Toc532254250)

[**3.** **Đánh giá hiện trạng** 5](#_Toc532254251)

[**Phần 2: PHÂN TÍCH HỆ THỐNG** 6](#_Toc532254252)

[**1.** **Hoạt động của phần mềm quản lý đoàn viên** 6](#_Toc532254253)

[**2.** **Lập bảng dữ liệu** 7](#_Toc532254254)

**[3.](#_Toc532254255)****[Thiết kết giao diện](#_Toc532254255)** [11](#_Toc532254255)

**[Phần 3: CHƯƠNG TRÌNH VÀ KẾT QUẢ ĐẠT ĐƯỢC](#_Toc532254256)** [17](#_Toc532254256)

[**1.** **Một số đoạn mã nguồn quan trọng** 17](#_Toc532254257)

[**1.1.** **Lớp DAL** 17](#_Toc532254258)

[**1.2.** **Lớp DTO** 18](#_Toc532254259)

[**1.3.** **Lớp BUS** 19](#_Toc532254260)

[**1.4.** **Lớp BUS** 22](#_Toc532254261)

[**Phần 4: ĐÁNH GIÁ VÀ KẾT LUẬN** 25](#_Toc532254262)

[**1.** **Đánh giá và kết luận** 25](#_Toc532254263)

[**Phần 5: TÀI LIỆU THAM KHẢO** 25](#_Toc532254264)

***Lời nói đầu***

Ngày nay cuộc sống đang phát triển,mọi thứ cần phải khắc phục và sửa đổi để đáp ứng nhu cầu cuộc sống để ngày một hoàn thiện hơn.

Hiện nay việc sử dụng tin học phục vụ trong nhà trường đã được áp dụng rất rộng rãi trong nước và quốc tế. Hầu hết tất cả các trường đại học, trung học cơ sở và phổ thông trung học đều sử dụng tin học trong công tác quản lý. Công tác quản lý trong môi trường giáo dục bao gồm rất nhiều mặt như quản lý học sinh, sinh viên, đoàn viên, học tập, hoạt động.

Trường Đại học Công nghiệp Hà Nội với số lượng sinh viên tương đối nhiều (khoảng 45.000 sinh viên). Việc quản lý đoàn viên của trường là cực kỳ khó khan nếu không có sự ứng dụng của tin học. Vì vậy để quản lý được ký túc xá sinh viên một cách chặt chẽ hiệu quả cần xây dựng một hệ thống quản lý bằng tin học. Đây là vấn đề bức thiết đối với nhà trường, cũng như BCH Đoàn trường Đại học Công nghiệp Hà Nôi.

Dựa vào mô hình cụ thể, quản lý đoàn viên Đại học Công nghiệp Hà Nội chúng em đã tìm hiểu và đi sâu vào nghiên cứu đề tài: “Phần mềm quản lý đoàn viên” dưới sự hướng dẫn của thầy Phạm Văn Hà phụ trách bộ môn Lập trình Windows.

Trong bài tập chúng em còn nhiều thiếu sót, rất mong Thầy và các bạn chỉ bảo thêm

Nhóm chúng em xin chân thành cảm ơn!

* **Giới thiệu hệ thống:**
* Hệ thống được xây dựng bằng ngôn ngữ C# (Windows Form)
* Cơ sở dữ liệu: Microsoft SQL Server
* Các chức năng chính của hệ thống:
  + - Đăng nhập
    - Quản lý đoàn viên
    - Quản lý hội viên
    - Quản lý đoàn phí
    - Quản lý người dùng
    - Quản lý hoạt động

# Phần 1: KHẢO SÁT VÀ PHÂN TÍCH BÀI TOÁN

1. **Khảo sát bài toán**

Với số lượng đoàn viên và thanh niên mỗi năm ngày càng tăng lên, trong khi đó số lượng cán bộ Đoàn của khoa có hạn và phần lớn phải tham gia các công tác chuyên môn. Do đó việc quản lý Đoàn viên đặt ra nhiều vấn đề cần phải giải quyết.

Trong công tác đoàn vụ đoàn khoa phải theo dõi chặt chẽ để quản lý các thông tin của từng đoàn viên thông qua sổ đoàn. Các hoạt động học tập, tham gia phong trào của đoàn viên cũng phải được quản lý chặt chẽ. Công tác thu đoàn phí phải được quản lý chính xác, mỗi năm phải thống kê các đoàn viên chưa đóng đoàn phí và lên danh sách. Các đoàn viên tham gia các hoạt động càn phải được ghi nhận để khen thưởng và đánh giá xếp loại.

Trong đoàn viên có những đoàn viên thuộc các đối tượng chính sách và miễn giảm, đoàn khoa càn phải nắm được danh sách các đoàn viên này để thuận lợi cho việc theo dõi và giúp đỡ.

Sau mỗi năm học đoàn khoa phải nộp bảng tổng họp các biểu mẫu về đoàn cấp trên và lên danh sách khen thưởng kỷ luật đối với các đoàn viên. Ngoài ra đoàn khoa phải nhận xét đánh giá vào sổ đoàn.

Khi đoàn viên chuyển sinh hoạt đoàn, đoàn khoa phải cấp giấy giới thiệu (nếu đủ điều kiện) cho đoàn viên chuyển sinh hoạt đến cơ sở Đoàn khác

Đoàn khoa là tổ chức xây dựng các chương trình hoạt động cho đoàn viên để hỗ trợ cho công tác học tập và giải trí cho đoàn viên. Do đó đoàn khoa cần phải nắm được các nhu cầu do đoàn viên đề xuất.

1. **Hiện trạng về nghiệp vụ hệ thống**
   1. **Quản lý đoàn viên về tổ chức**
   2. **Đối với đoàn trường**

Khi một đoàn viên chuyển đến, thông tin cá nhân được lưu tại văn phòng đoàn trường theo mẫu.

Hàng ngày Đoàn trường sẽ theo dõi hoạt động của đoàn viên thông qua các tổ chức trực thuộc như Liên chi đoàn các khoa, các chi đoàn trong toàn trường.

Mỗi sinh viên sẽ được phát một sổ đoàn, huy hiệu đoàn và thẻ đoàn:

Hồ sơ đoàn viên: Cuốn “Sổ đoàn viên” (Khổ 13x19 cm) gồm:

Đoàn TNCS Hồ Chí Minh tự giới thiệu

Người xin vào Đoàn tự giới thiệu

Đơn xin vào Đoàn

Nghị quyết về việ chuẩn y kết nạp đoàn viên

Nhân xét ưu, khuyết điểm hàng năm, phần nàn do BCH Liên chi đoàn nhận xét ưu khuyết điểm hàng năm, thành tích được khen thưởng, khuyết điểm bị kỷ luật và công nhận tiến bộ.

Giới thiệu chuyển sinh hoạt Đoàn

* 1. **Đối với chi đoàn**

Ban chấp hành chi đoàn phải có “Sổ chi đoàn” theo mẫu của trung ương Đoàn

Sau mỗi nhiệm kỳ ban chấp hành chi đoàn phải thực hiện đầy đủ các yêu cầu trong “Sổ chi đoàn”

1. **Đánh giá hiện trạng**

Sau khi khảo sát và đánh giá hoạt động về công tác quản lý đoàn viên của Trường Đại học Công nghiệp Hà Nội, có thể nhận định được những thiếu xót của hiện trạng như:

* + Thiếu các công cụ hỗ trợ nghiệp vụ công việc. (Chủ yếu là ghi sổ sách)
  + Không có một công cụ để trao đổi thông tin giữa đoàn viên và cán bộ đoàn Trường.
  + Kém hiệu lực và thiều hiệu quả
  + Mất thời gian làm báo cáo.
  + Các thông tin quản lý số sách nhiều, phức tạp.
  + Khi cần, truy xuất, tìm kiếm thông tin mất nhiều thời gian.
  + Không đạt hiệu quả cao trong công việc.
  + Lãng phí thời gian công sức, tức lãng phí ngày lao động của các nhân viên để thực hiện các nghiệp vụ cộng, tổng hợp số liệu, lập báo cáo…

**Phần 2: PHÂN TÍCH HỆ THỐNG**

1. **Hoạt động của phần mềm quản lý đoàn viên**

**Quản lý đoàn viê**n: Khi sinh viên chuyển tới cơ sở Đoàn, nếu sinh viên đó được Đoàn trường tiếp nhân. Đoàn trường sẽ cập nhật hồ sơ sinh viên đó vào hệ thống. Khi cần thiết có thể xóa hoặc sửa thông tin của sinh viên. Thông tin đoàn viên: Mã sinh viên, họ và tên, địa chỉ, số điện thoại, lớp, ngành học, khoa, khóa học.

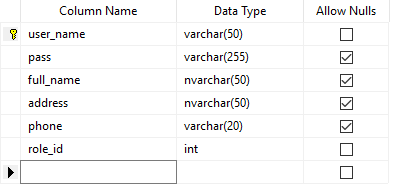
**Quản lý hội viên:** Khi sinh viên được kết nạp vào Hội sinh viên, Hội sinh viên trường sẽ cập nhật hồ sơ sinh viên đó vào hệ thống. Khi cần thiết có thể xóa hoặc sửa thông tin của sinh viên. Thông tin đoàn viên: Mã sinh viên, họ và tên, địa chỉ, số điện thoại, lớp, ngành học, khoa, khóa học.

**Quản lý đoàn phí – hội phí:** Khi Đoàn trường và Hội sinh viên trường có thông báo thu đoàn phí – hội phí. Thông báo sẽ được báo tới tất cả sinh viên toàn trường. Hệ thống sẽ cập nhật tình hình đóng đoàn phí và hội phí của sinh viên đưa ra báo cáo – thống kê.

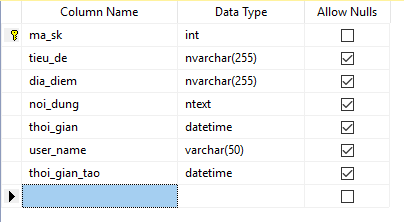
**Quản lý hoạt động:** Khi Đoàn trường – Hội sinh viên trường có các hoạt động dành cho sinh viên. Hoạt động sẽ được thông báo tới các sinh viên. Cũng như đếm ngược tới hoạt động gần nhất.

1. **Lập bảng dữ liệu**

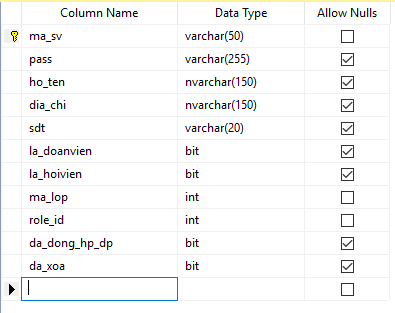
***Bảng User***



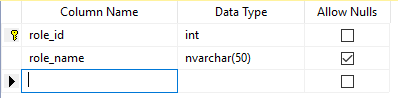
***Bảng hoạt động***



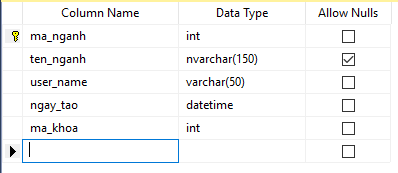
***Bảng sinh viên***



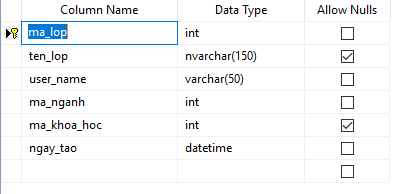
***Bảng phần quyền***



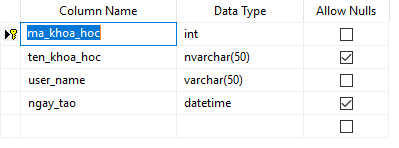
***Bảng ngành học***



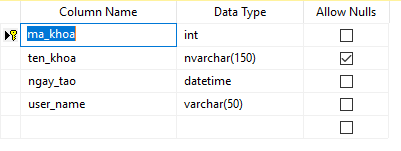
***Bảng lớp***

******

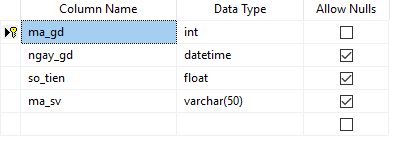
***Bảng khóa học***



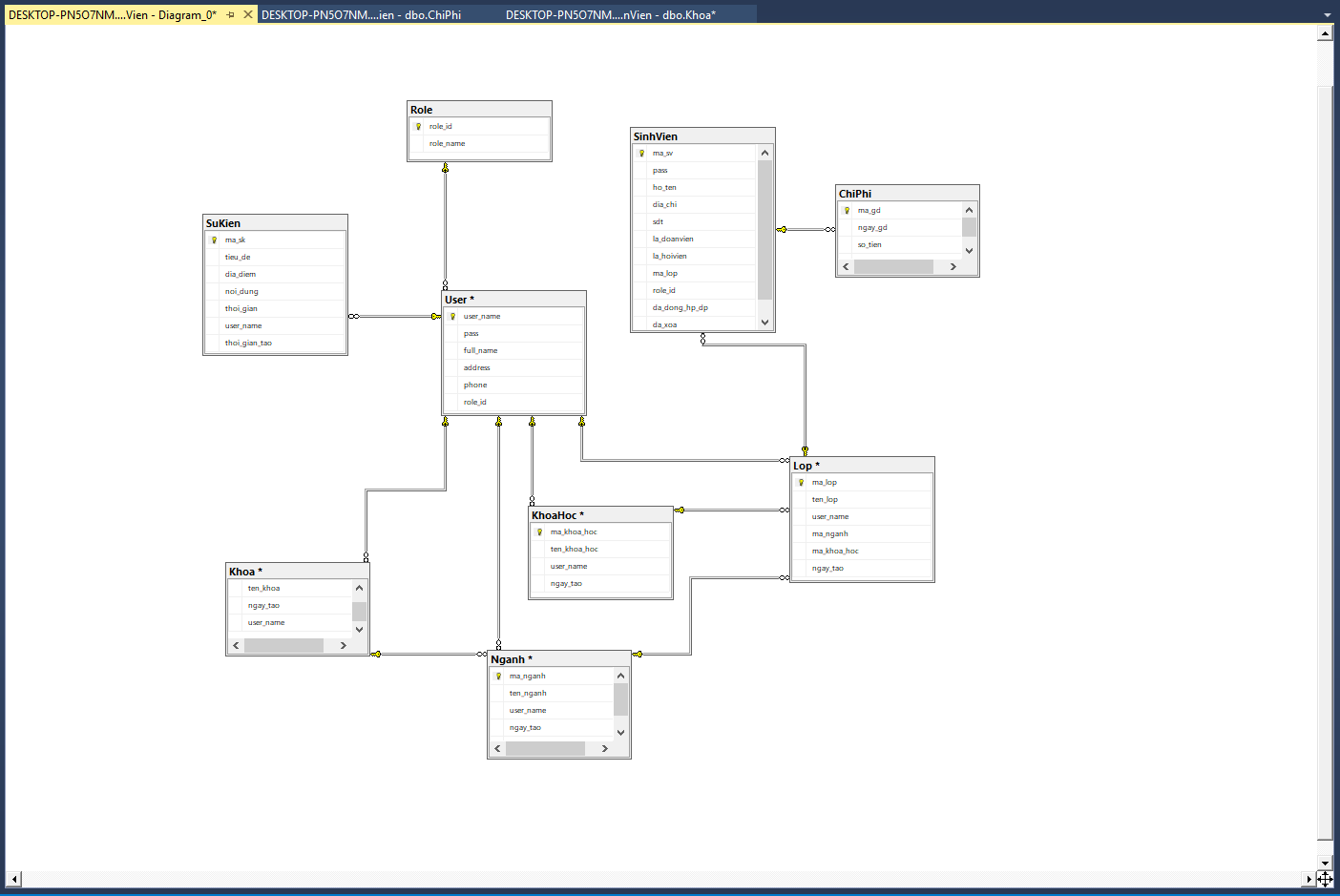
***Bảng khoa***



***Bảng chi phí***

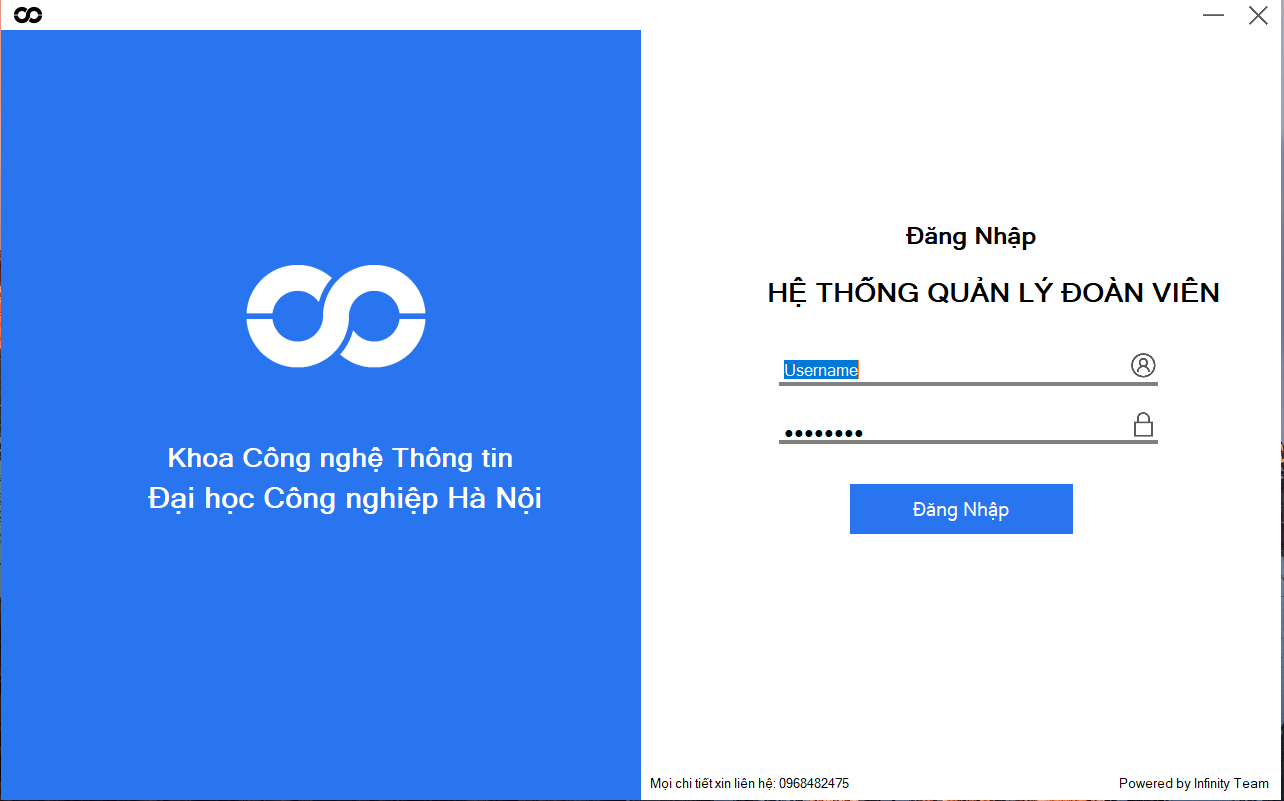


***Mô hình dữ liệu vật lý***

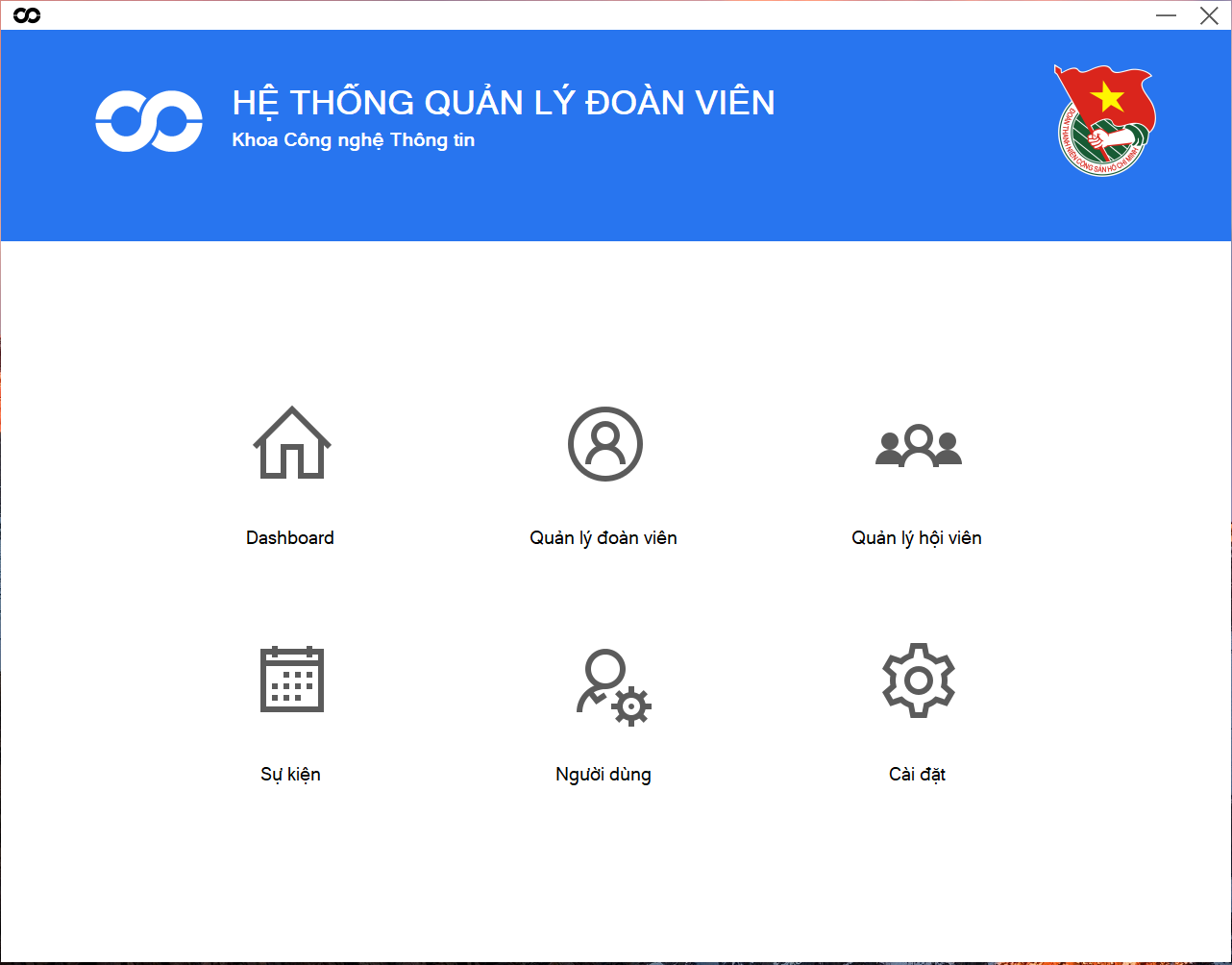


1. **Thiết kết giao diện**

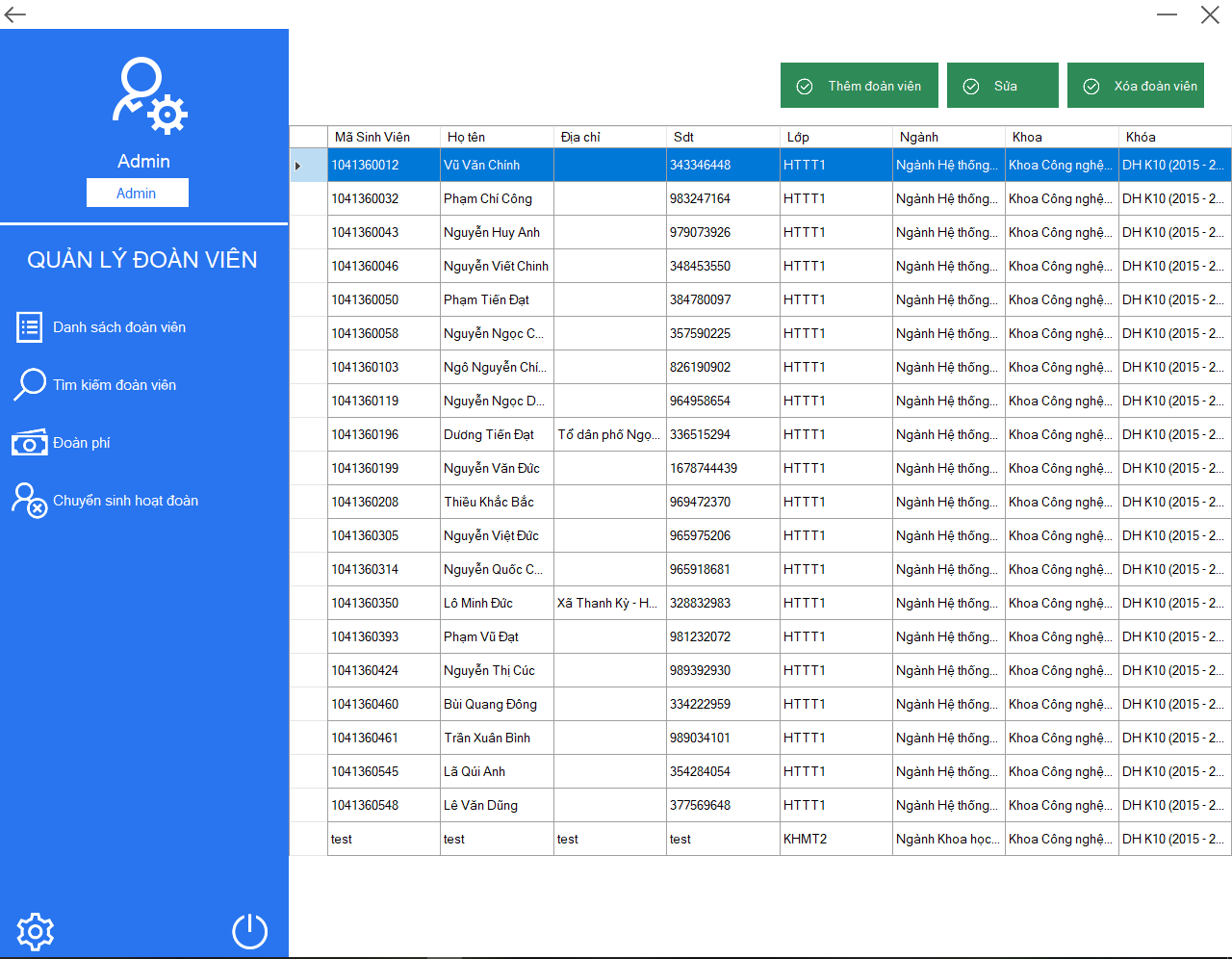
***Giao diện đăng nhập***



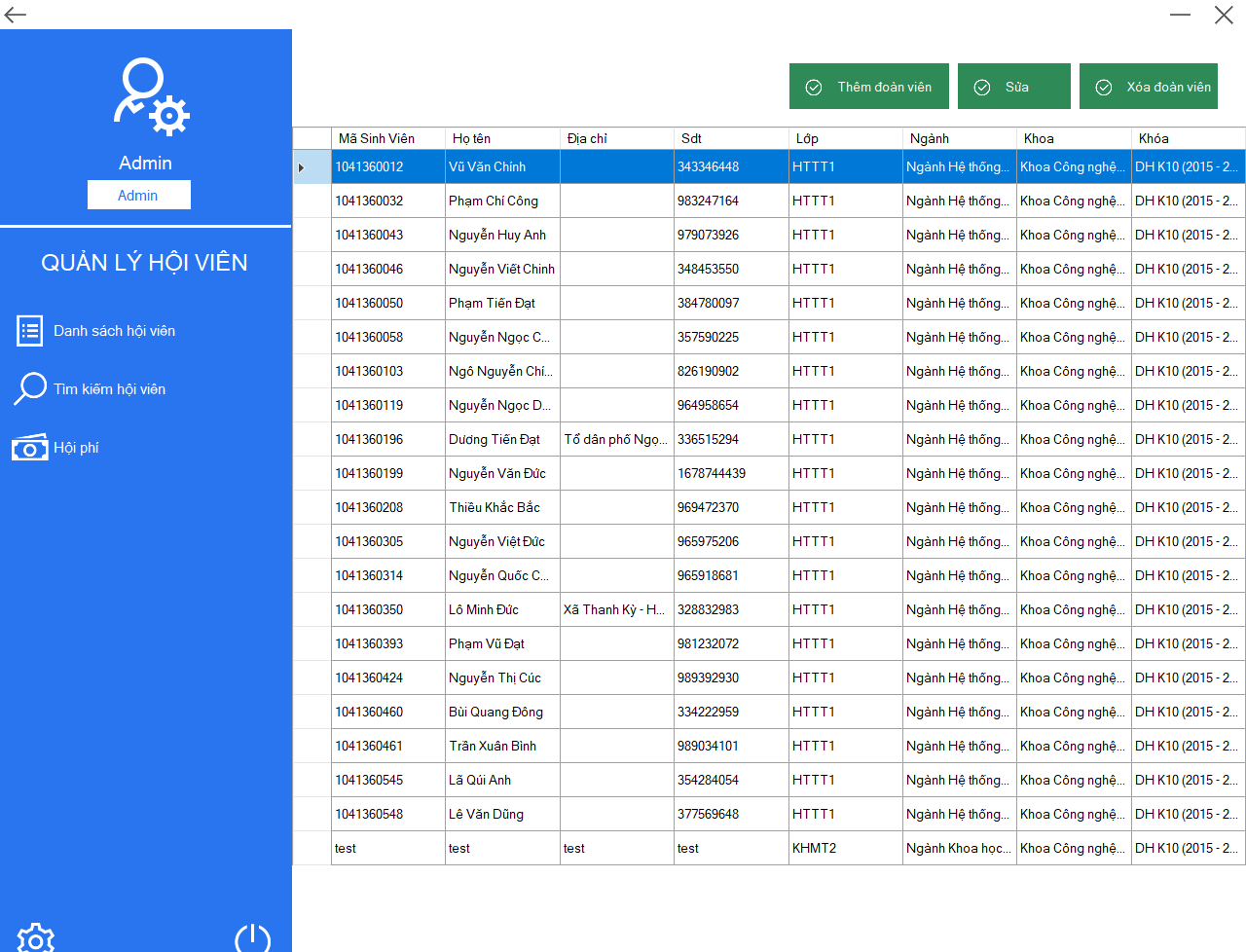
***Giao diện bảng điều khiển***

******

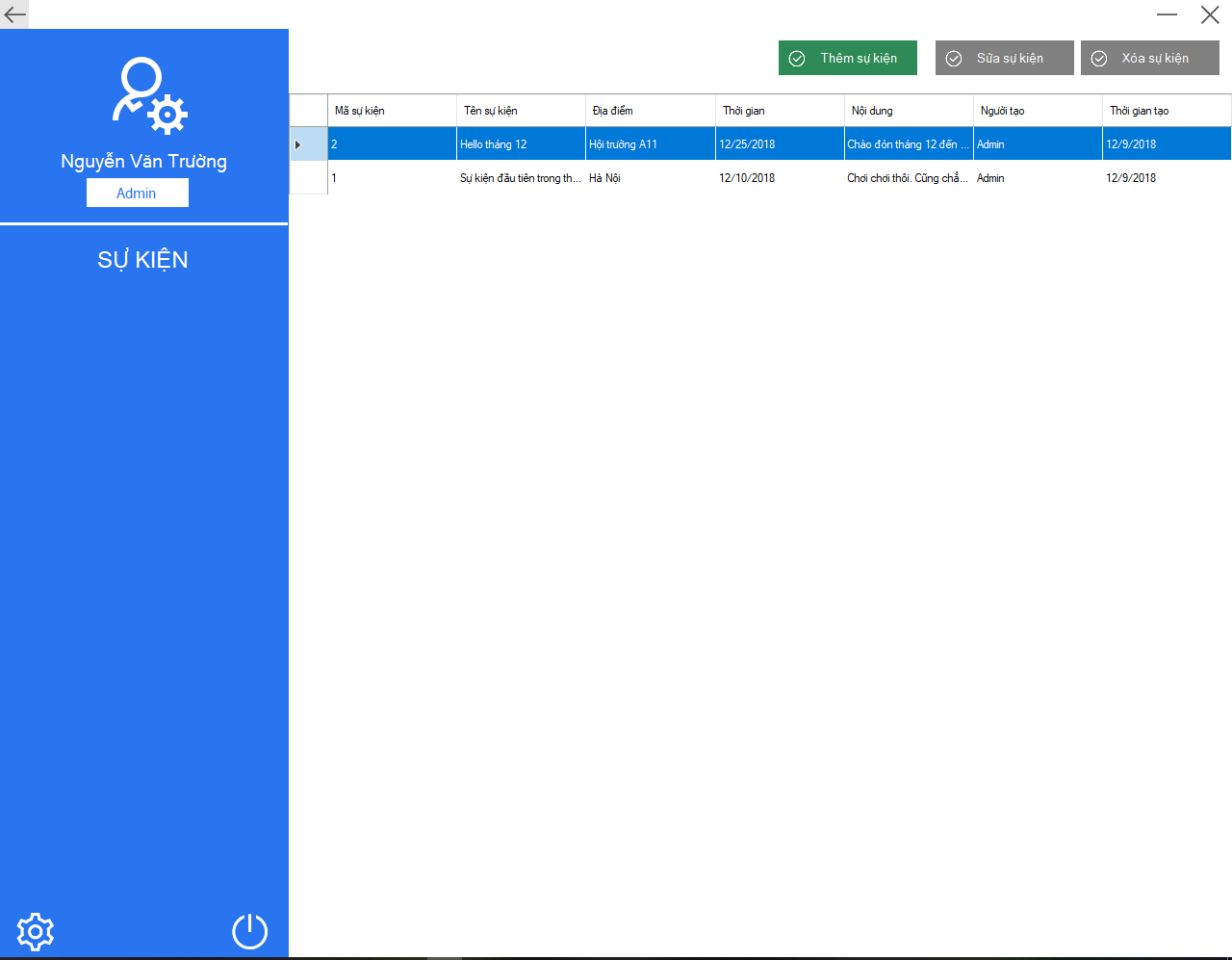
***Giao diện quản lý đoàn viên***



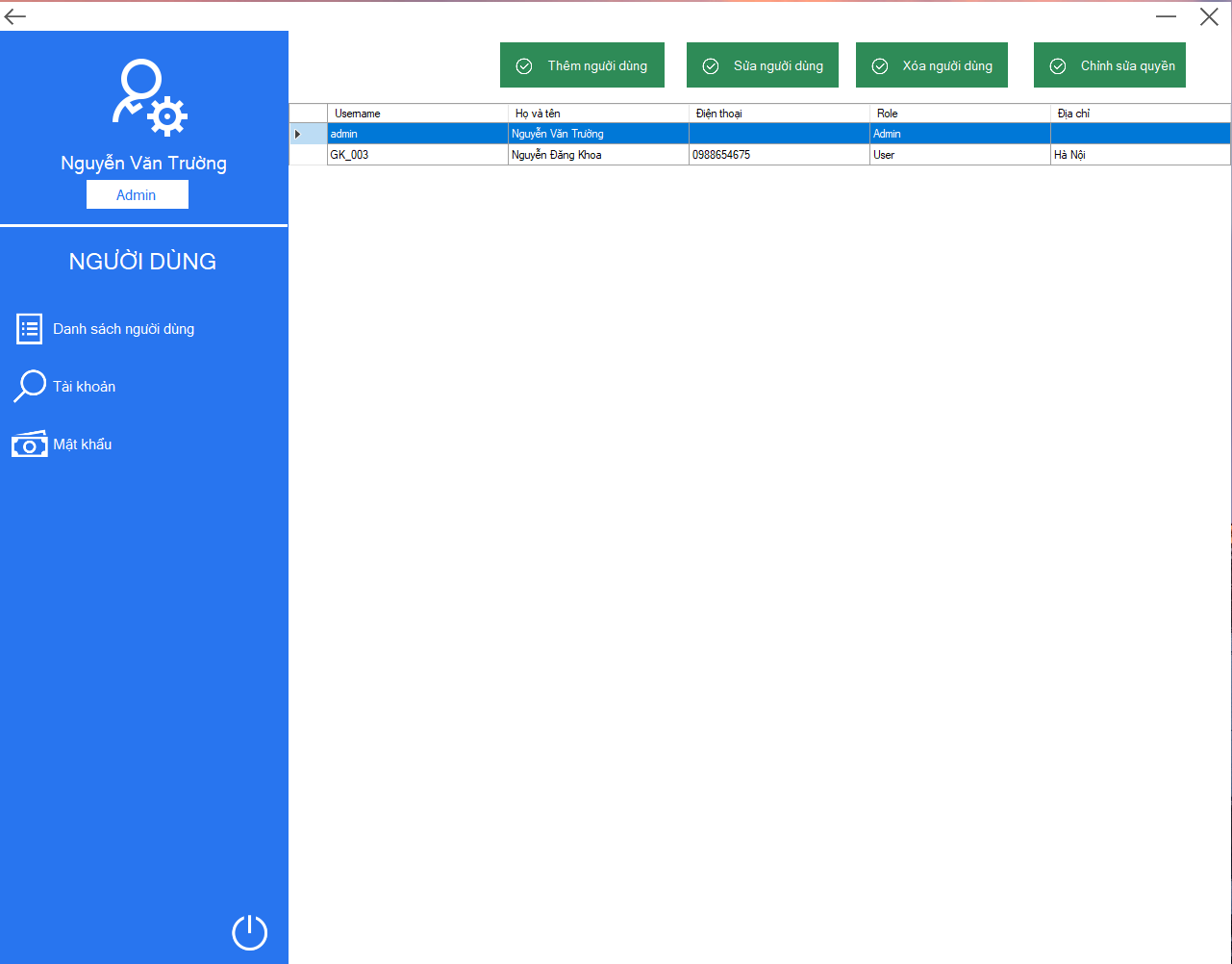
***Giao diện quản lý hội viên***



***Giao diện quản lý sự kiện***



***Giao diện quản lý người dung***

******

**Phần 3: CHƯƠNG TRÌNH VÀ KẾT QUẢ ĐẠT ĐƯỢC**

1. **Một số đoạn mã nguồn quan trọng**
   1. **Lớp DAL**

namespace DAL

{

public class DataProvider

{

private static DataProvider instance;

public static DataProvider Instance

{

get { if (instance == null) instance = new DataProvider(); return DataProvider.instance; }

private set { instance = value; }

}

private string stringConnection = @"Data Source=DESKTOP-PN5O7NM;Initial Catalog=QLDoanVien;Integrated Security=True";

public DataTable ExcuteQuery(string query, object[] parameter = null)

{

DataTable da = new DataTable();

using (SqlConnection connection = new SqlConnection(stringConnection))

{

SqlCommand command = new SqlCommand(query, connection);

SqlDataAdapter adapter = new SqlDataAdapter(command);

if (parameter != null)

{

string[] listPar = query.Split(' ');

int index = 0;

foreach (string item in listPar)

{

if (item.Contains("@"))

{

command.Parameters.AddWithValue(item, parameter[index]);

index++;

}

}

}

adapter.Fill(da);

connection.Close();

}

return da;

}

public int ExcuteNonQuery(string query, object[] parameter = null)

{

int result = -1;

using (SqlConnection connection = new SqlConnection(stringConnection))

{

connection.Open();

SqlCommand command = new SqlCommand(query, connection);

if (parameter != null)

{

string[] listPar = query.Split(' ');

int index = 0;

foreach (string item in listPar)

{

if (item.Contains("@"))

{

command.Parameters.AddWithValue(item, parameter[index]);

index++;

}

}

}

result = command.ExecuteNonQuery();

}

return result;

}

}

}

* 1. **Lớp DTO**

namespace DTO

{

public class Events

{

public int ma\_sk { get; set; }

public string tieu\_de { get; set; }

public string noi\_dung { get; set; }

public string dia\_diem { get; set; }

public DateTime thoi\_gian { get; set; }

public DateTime thoi\_gian\_tao { get; set; }

public string user\_name { get; set; }

}

}

* 1. **Lớp BUS**

namespace BUS

{

public class EventsBus

{

private static EventsBus instance;

public static EventsBus Instance

{

get { if (instance == null) instance = new EventsBus(); return EventsBus.instance; }

private set { instance = value; }

}

public List<SuKienModel> DanhSachSuKien()

{

List<SuKienModel> events = new List<SuKienModel>();

string query = @"SELECT [ma\_sk]

,[tieu\_de]

,[noi\_dung]

,[dia\_diem]

,[thoi\_gian]

,[thoi\_gian\_tao]

,[User].[full\_name]

FROM [dbo].[SuKien]

INNER JOIN [dbo].[User] ON [SuKien].[user\_name] = [User].[user\_name] ";

DataTable data = DataProvider.Instance.ExcuteQuery(query);

if(data.Rows.Count>0)

{

foreach(DataRow row in data.Rows)

{

events.Add(GetEvent(row));

}

return events.OrderByDescending(x=>x.thoi\_gian).ToList();

}

else

{

return null;

}

}

private SuKienModel GetEvent(DataRow row)

{

SuKienModel temp = new SuKienModel();

temp.ma\_sk = int.Parse(row["ma\_sk"].ToString());

temp.tieu\_de = row["tieu\_de"].ToString();

temp.dia\_diem = row["dia\_diem"].ToString();

temp.noi\_dung = row["noi\_dung"].ToString();

temp.thoi\_gian = DateTime.Parse(row["thoi\_gian"].ToString());

temp.thoi\_gian\_tao = DateTime.Parse(row["thoi\_gian\_tao"].ToString());

temp.full\_name = row["full\_name"].ToString();

return temp;

}

public int ThemSuKien(Events events)

{

string query = @"INSERT INTO [dbo].[SuKien]

( [tieu\_de]

,[dia\_diem]

,[noi\_dung]

,[thoi\_gian]

,[user\_name]

,[thoi\_gian\_tao] )

VALUES

( @tieu\_de , @dia\_diem , @noi\_dung , @thoi\_gian , @user\_name , @thoi\_gian\_tao )";

return DataProvider.Instance.ExcuteNonQuery(query, new object[] { events.tieu\_de, events.dia\_diem, events.noi\_dung, events.thoi\_gian, events.user\_name, events.thoi\_gian\_tao });

}

public int SuaSuKien(Events events)

{

string query = @"UPDATE [dbo].[SuKien] SET tieu\_de = @tieu\_de , dia\_diem = @dia\_diem , noi\_dung = @noi\_dung , thoi\_gian = @thoi\_gian , user\_name = @user\_name WHERE ma\_sk = @ma\_sk ";

return DataProvider.Instance.ExcuteNonQuery(query, new object[] { events.tieu\_de, events.dia\_diem, events.noi\_dung, events.thoi\_gian, events.user\_name, events.ma\_sk });

}

public int XoaSuKien(Events events)

{

string query = @"DELETE FROM [dbo].[SuKien] WHERE ma\_sk = @ma\_sk";

return DataProvider.Instance.ExcuteNonQuery(query, new object[] { events.ma\_sk });

}

public Events GetEvents(int ma\_sk)

{

string query = @"SELECT \* FROM [QLDoanVien].[dbo].[SuKien] Where ma\_sk = @ma\_sk ";

DataTable data = new DataTable();

data = DataProvider.Instance.ExcuteQuery(query, new object[] { ma\_sk });

if(data.Rows.Count > 0)

{

return GetEvents(data.Rows[0]);

}

return null;

}

private Events GetEvents(DataRow row)

{

Events temp = new Events();

temp.ma\_sk = int.Parse(row["ma\_sk"].ToString());

temp.tieu\_de = row["tieu\_de"].ToString();

temp.dia\_diem = row["dia\_diem"].ToString();

temp.noi\_dung = row["noi\_dung"].ToString();

temp.thoi\_gian = DateTime.Parse(row["thoi\_gian"].ToString());

temp.thoi\_gian\_tao = DateTime.Parse(row["thoi\_gian\_tao"].ToString());

temp.user\_name = row["user\_name"].ToString();

return temp;

}

}

}

* 1. **Lớp BUS**

namespace ADO.UC.DV

{

public partial class DanhSachDoanVienUC : UserControl

{

private List<SinhVienModel> list;

private SinhVien sv;

private User user;

public DanhSachDoanVienUC(User user)

{

InitializeComponent();

dataGridView1.DataSource = SinhVienBus.Instance.GetSinhVienModels();

this.user = user;

if(user.role\_id != 1)

{

btnSuaDV.Visible = false;

btnThemDV.Visible = false;

btnXoaDV.Visible = false;

}

}

public DanhSachDoanVienUC(List<SinhVienModel> list, User user)

{

InitializeComponent();

this.list = list;

dataGridView1.DataSource = this.list;

this.user = user;

if (user.role\_id != 1)

{

btnSuaDV.Visible = false;

btnThemDV.Visible = false;

btnXoaDV.Visible = false;

}

}

private void LoadData()

{

dataGridView1.DataSource = SinhVienBus.Instance.GetSinhVienModels();

dataGridView1.Refresh();

}

private void btnThemDV\_Click(object sender, EventArgs e)

{

DoanVienDialog dv = new DoanVienDialog("Thêm mới sinh viên");

if(dv.ShowDialog() == DialogResult.OK)

{

LoadData();

}

}

private void dataGridView1\_CellClick(object sender, DataGridViewCellEventArgs e)

{

try

{

DataGridViewRow row = dataGridView1.Rows[e.RowIndex];

string masv = row.Cells[0].Value.ToString();

sv = SinhVienBus.Instance.GetSinhVienDV(masv);

}

catch { }

}

private void btnSuaDV\_Click(object sender, EventArgs e)

{

if(sv != null)

{

DoanVienDialog dv = new DoanVienDialog("Sửa thông tin sinh viên", sv, false);

if (dv.ShowDialog() == DialogResult.OK)

{

LoadData();

}

}

else

{

MessageBox.Show("Bạn chưa chọn sinh viên");

}

}

private void btnXoaDV\_Click(object sender, EventArgs e)

{

if (sv != null)

{

ConfirmDialog confirmDialog = new ConfirmDialog(Extention.Confirm.IS\_DOANVIEN, sv);

confirmDialog.deleteSuccess += ConfirmDialog\_deleteSuccess;

confirmDialog.ShowDialog();

}

else

{

MessageBox.Show("Bạn phải chọn một sinh viên");

}

}

private void ConfirmDialog\_deleteSuccess()

{

LoadData();

}

}

}

**Phần 4: ĐÁNH GIÁ VÀ KẾT LUẬN**

1. **Đánh giá và kết luận**

Trong quá trình làm bài tập lớn các thành viên trong nhóm đã cố gắng làm tốt phần việc được giao của nhóm trưởng, thông qua bài tập lớn các thành viên trong trong nhóm đã đoàn kết với nhau hơn, tinh thần làm việc tập thể được nâng lên cao hơn. Nhưng do trình độ hiểu biết còn hạn chế nên bài làm không tránh khỏi những khuyết điểm. Vì vậy chung em rất mong nhận được sự đóng góp ý kiến của thầy giáo và các bạn sinh viên để bài tập lớn của chúng em được hoàn thiện hơn. Chúng em xin chân thành cảm ơn thầy giáo Phạm Văn Hà đã nhiệt tình hướng dẫn chúng em làm bài tập lớn này trong suốt quá trình qua, được sự nhiệt tình hướng dẫn của thầy chúng em đã hoàn thành được bài tập lớn và hiểu hơn về môn học.

**Phần 5: TÀI LIỆU THAM KHẢO**

[1]. **Bài giảng của thầy giáo Phạm Văn Hà**

[2]. **Giáo trình lập trình Windows – Đỗ Ngọc Sơn (Chủ biên)**