**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT ĐẠI HỌC ĐÀ NẴNG**

**NGÀNH CÔNG NGHỆ THÔNG TIN ===== =====**

****

ĐỒ ÁN KẾT THÚC HỌC PHẦN

***Môn học*: CÔNG NGHỆ XML**

Đề tài: *Quản lý bán nông sản cho cửa hàng tiện lợi*

**GVHD : NGÔ LÊ QUÂN**

**NHÓM SV : NHÓM 09**

**HỌ TÊN SV : TRẦN ĐẠI**

**TĂNG THỊ THU HÒA**

**LỚP : 302CNX01**

***Đà Nẵng, tháng 8 năm 2021***

**LỜI MỞ ĐẦU**

XML là từ viết tắt của từ *Extensible Markup Language* là ngôn ngữ đánh dấu mở rộng. XML có chức năng truyền dữ liệu và mô tả nhiều loại dữ liệu khác nhau. Tác dụng chính của XML là đơn giản hóa việc chia sẻ dữ liệu giữa các nền tảng và các hệ thống được kết nối thông qua mạng Internet.

XML dùng để cấu trúc, lưu trữ và trong trao đổi dữ liệu giữa các ứng dụng và lưu trữ dữ liệu. Ví dụ khi ta xây dựng một ứng dụng bằng Php và một ứng dụng bằng Java thì hai ngôn ngữ này không thể hiểu nhau, vì vậy ta sẽ sử dụng XML để trao đổi dữ liệu. Chính vì vậy, XML có tác dụng rất lớn trong việc chia sẻ, trao đổi dữ liệu giữa các hệ thống**.**

Ưu điểm lớn nhất của XML là sự độc lập, XML được sử dụng để mô tả dữ liệu dưới dạng text. Nên hầu hết các chương trình bình thường đều có thể đọc được chúng.

XML có thể đọc và phân tích nguồn dữ liệu khá dễ dàng nên nó được sử dụng với mục đích chính là trao đổi dữ liệu giữa các chương trình, các hệ thống khác nhau. XML được sử dụng Remove Procedure Calls nhằm phục vụ cho các dịch vụ của Website.

Tuy nhiên, XML cũng có những sai sót (5% - 7%), con số này tuy không quá cao, nhưng thực tế, người ta vẫn cần cân nhắc trước khi sử dụng chúng.

Trong quá trình học và tiếp cận công nghệ XML lần này, nhóm chúng em sẽ áp dụng kiến thức để trải nghiệm vào ĐỒ ÁN KẾT THÚC HỌC PHẦN – với chủ đề: ***Quản lý bán nông sản cho cửa hàng tiện lợi***. Với việc sử dụng công nghệ XML cho quản lý Database, kết hợp với ngôn ngữ C#. Hơn hết, đó sự đồng hành và giúp đỡ của thầy giáo chủ nhiệm môn này: Ngô Lê Quân. Qua đây, nhóm chúng em xin cảm ơn thầy đã nhiệt tình chỉ bảo và trao kiến thức để chúng em thực hiện đề tài lần này!

*Nhóm thực hiện đề tài*

Mục lục

Contents

[Chương I: GIỚI THIỆU 5](#_Toc79337643)

[I. Ích lợi khi sử dụng công nghệ XML: 5](#_Toc79337644)

[II. Mục tiêu đề tài: 5](#_Toc79337645)

[III. Phạm vi đề tài: 6](#_Toc79337646)

[**Chương 2: Nội dung** 7](#_Toc79337647)

[I. Khảo sát nghiệp vụ: 7](#_Toc79337648)

[1. Khảo sát và đánh giá hiện trạng: 7](#_Toc79337649)

[2. Mô hình hoạt động nghiệp vụ cho quản lý bán nông sản: 7](#_Toc79337650)

[2.1. Các bước thực hiện của nhân viên khhi hoạt động thực tế: 7](#_Toc79337651)

[2.2 Kết luận: 8](#_Toc79337652)

[II. Phân tích thiết kế hệ thống: 8](#_Toc79337653)

[1. Mô tả vấn đề: 8](#_Toc79337654)

[1.1. Mục đích chính của hệ thống: 8](#_Toc79337655)

[1.2. Các chức năng chính: 8](#_Toc79337656)

[1.3. Đối tượng sử dụng và vai trò của mỗi đối tượng: 8](#_Toc79337657)

[2. Đặc tả yêu cầu: 9](#_Toc79337658)

[2.1. Quản lý nhập nông sản: 9](#_Toc79337659)

[2.2. Quản lý nông sản: 9](#_Toc79337660)

[2.3. Quản lý Bán nông sản: 9](#_Toc79337661)

[2.4. Quản lý tài khoản khách hàng: 9](#_Toc79337662)

[III. Thiết kế cơ sở dữ liệu 10](#_Toc79337663)

[1. Mô hình phân cấp chức năng 10](#_Toc79337664)

[2. Xác định các thực thể 10](#_Toc79337665)

[3. Mô hình dữ liệu mức quan hệ 11](#_Toc79337666)

[4.1. Bảng Nhân viên (tableNhanVien) 11](#_Toc79337667)

[4.2. Bảng Khách hàng (tableKhachHang) 11](#_Toc79337668)

[4.3. Bảng Danh mục sản phẩm (tableDanhMucNongSan) 12](#_Toc79337669)

[4.4. Bảng Chi tiết danh mục (tableChiTietNongSan) 12](#_Toc79337670)

[4.5. Bảng Hóa đơn nhập xuất (tableHoaDonNhapXuat) 12](#_Toc79337671)

[4.6. Bảng Chi tiết hóa đơn (tableChiTietHoaDon) 12](#_Toc79337672)

[IV. Thiết kế và cài đặt chương trình 12](#_Toc79337673)

[1. Các Module chính 12](#_Toc79337674)

[1.1. Form Đăng nhập: 12](#_Toc79337675)

[1.2. Form Menu: 13](#_Toc79337676)

[1.3. Form Quản lý nông sản: 14](#_Toc79337677)

[1.4. Form Quản lý bán hàng – Thanh toán: 16](#_Toc79337678)

[1.5. Form Quản lý khách hàng: 19](#_Toc79337679)

[1.6. Form Quản lý – Sao lưu dữ liệu: 21](#_Toc79337680)

[2. Giao diện chương trình: 22](#_Toc79337681)

[2.1. Form Đăng nhập: 22](#_Toc79337682)

[2.2. Form Menu: 22](#_Toc79337683)

[2.3. Form Nhập – Quản lý nông sản: 23](#_Toc79337684)

[2.4. Form Bản nông sản: 24](#_Toc79337685)

[2.5. Form Quản lý khách hàng: 24](#_Toc79337686)

[2.6. Form Quản lý Sao lưu dữ liệu 24](#_Toc79337687)

# **Chương I: GIỚI THIỆU**

# I. Ích lợi khi sử dụng công nghệ XML:

Như đã biết, mục đích chính của việc áp dụng công nghệ XML là chia sẻ tài nguyên dữ liệu giữa các platform trên các hệ thống khác nhau. Đặc biệt, các hệ thống được kết nối mạng, file XML được chia sẻ nhanh chóng hơn. Bởi vậy, XML thường được dùng vào việc thay đổi, chia sẻ giữa các hệ thống với nhau.

Khi có sự trao đổi dữ liệu giữa các hệ thống khác nhau, dữ liệu đó được tổ chức dưới dạng XML. Ví dụ cụ thể: Nếu một nhà hàng muốn lấy thông tin khách du lịch từ hệ thống công ty lữ hành thì giữa hệ thống cần đảm bảo thực hiện các thao tác công việc sau:

Các hệ thống cần có sự thống nhất với nhau về file cấu trúc XML

Công ty lữ hành cần có trích xuất dữ liệu từ hệ thống của mình, đóng gói dữ liệu dưới dạng file XML theo cấu trúc đã thỏa thuận trước đó.

Vì vậy, ưu điểm của công nghệ XML là:

* XML là dữ liệu độc lập, đây cũng là ưu điểm lớn nhất của XML. Nó được sử dụng và mô tả dữ liệu dưới dạng text. Vì vậy, hầu hết các phần mềm hay chương trình bình thường đều đọc được nó.
* File XML có thể dễ dàng đọc và phân tích dữ liệu. Do đó, nó được dùng để trao đổi dữ liệu giữa các chương trình, hệ thống với nhau.
* File XML được tạo một cách dễ dàng chủ với vài thao tác đơn giản.
* File XML được sử dụng dành cho Remote Procedure Calls với mục đích phục vụ các dịch vụ trong thiết kế Website.

# II. Mục tiêu đề tài:

Tìm hiểu và phân tích đề tài đã chọn, khảo sát thực tế và nhu cầu hoạt động của việc quản lý bán hàng nông sản hiện nay của các của hàng với quy mô lớn. Việc quản lý xuất nhập kho và bán ra là cực kì bấc lợi khi hóa đơn phải nhập bằng tay và viết giấy.

Vậy nên, thiết kế nên một phần mềm quản lý bán Nông sản hướng người sử dụng đỡ phải quản lý bằng các loại giấy tờ, thay vào đó phần mềm sẽ điều khiển và lưu trữ thông tin, tính tiền cho khách, in hóa đơn.

Với việc thiết kế phần mềm của nhóm chúng em, yêu cầu nên cần cho người sử dụng như sau: dễ sử dụng, quản lý tốt, giao diện bắt mắt, thao tác nhanh, dễ sử dụng,…

# III. Phạm vi đề tài:

Phần mềm được xây dựng trên phạm vi môn học “CÔNG NGHỆ XML”, sử dụng phần mềm code Visual Studio 2010 (Ngôn ngữ C#), lưu trữ cơ sở dữ liệu bằng file XML.

|  |  |
| --- | --- |
| STT | Yêu cầu |
| 1 | Cơ sở dữ liệu (chuyển đổi giữa XML và mySQL) |
| 2 | XSD (lược đồ) |
| 3 | XSLT |
| 4 | Form Database |
| 5 | Thêm, Sửa, Xóa |
| 6 | Hiển thị trình duyệt |
| 7 | Cấp nhật dữ liệu |
| 8 | Kiểm tra hợp lệ |

Yêu cầu cần có trong phần mềm:

# **Chương 2: Nội dung**

# I. Khảo sát nghiệp vụ:

# 1. Khảo sát và đánh giá hiện trạng:

Như chúng ta đã biết, mặt hàng nông sản từ lâu không còn xa lạ với chúng ta. Xã hội hiện đại hơn, những mặt hàng này không chỉ có mặt ngoài chợ, mà thay vào đó, những shop bán hàng với những nông sản được gắn mát chất lượng, sạch sẽ, an toàn và có nguồn gốc xuất xứ rõ ràng. Rất hợp với yêu cầu người sử dụng.

Ngày càng các của hàng tiện lợi chuyên nhập và bán các nông sản sạch mọc lên. Tuy nhiên, việc quá tải do yeu cầu người sử dụng đã dẫn đến một số khó khăn về quản lý nông sản của cửa hàng. Lý do thực tế: nhân viên quản lý một lần với một lương lớn nông sản, rất nhiều loại danh mục, thông tin từng sản phẩm, số lượng, … Vì thế mà chúng ta nên có giải pháp tốt hơn để quản lý khối nông sản này cho nhân viên.

Bài toán mà nhóm chúng em đặt ra ở đây gồm có:

* + Quản lý công việc, hóa đơn nhập hàng từ các nhà sản xuất
  + Quản lý danh mục, số lượng nông sản hiện có
  + Quản lý khách hàng đăng kí khách thành viên với cửa hàng để nhận khuyến mãi

|  |  |
| --- | --- |
| Yêu cầu phần mềm | Yêu cầu dữ liệu |
| Phần mềm dễ sử dụng, dễ cài đặt, tạo điều kiện thuận lợi cho người thao tác. | Dữ liệu phải được thiết kế chính xác, phù hợp với yêu cầu khách hàng. |
| Phần mềm có giao diện dễ sử dụng, tiện nghi, thân thiện và thích ứng với ngôn ngữ máy tính. | Xử lý được chức năng của hệ thống. |
| Phần mềm có phân quyền cho nhân viên và quản trị viên, để hệ thống và kiểm tra, củng cố cho người sử dụng. |  |

# 2. Mô hình hoạt động nghiệp vụ cho quản lý bán nông sản:

# 2.1. Các bước thực hiện của nhân viên khhi hoạt động thực tế:

Phân loại các nhóm danh mục nông sản theo thể loại, thành các nhóm riêng. Cập nhật danh sách các nhóm nông sản riêng, phân vùng và trưng bày để khách hàng dễ tìm kiếm và lựa chọn.

Nhân viên phải thường xuyên thiết kế và trung bày các bảng mục lục danh sách các nông sản, nội dung, giúp khách hàng đến tra cứu và tìm kiếm sản phẩm muốn mua.

Nhân viên thực hiên các công việc bán hàng, ghi chép sổ sách và nhật ký bán hàng. Hỗ trợ khách hàng tìm kiếm, tra cứu thông tin về nông sản.

Trên cơ sở dữ liệu sổ sách đề ra danh sách các đĩa phim cần phải bổ sung và nhập vào.

Kiểm tra định kỳ danh sách các nông sản nhập về, số lượng bán được, số lượng tồn đọng.

# 2.2 Kết luận:

Qua phân tích trên, ta thấy rằng việc sắp xếp, phân loại và quản lý rất phức tạp,… có thể gây nhầm lần cho người kinh doanh. Việc quản lý nông sản cũng khó xử lý, khó tra cứu.

Vì vậy, việc tạo ra sản phẩm phần mềm “Quản lý bán nông sản” cần đặt ra đáp ứng được yêu cầu nghiệp vụ trên.

# II. Phân tích thiết kế hệ thống:

# 1. Mô tả vấn đề:

# 1.1. Mục đích chính của hệ thống:

Tin học hóa về dịch vụ bán và quản lý nông sản của cửa hàng

Giúp người dùng có thể tiết kiệm chi phí thuê nhân viên đồng thời tiết kiệm thời gian quản lý số lượng hàng hóa.

# 1.2. Các chức năng chính:

* + Quản lý Nhập nông sản (số lượng nhập vào, bán ra)
  + Quản lý Bán nông sản
  + Quản lý Nông sản (số lượng hiện có trong cửa hàng)
  + Quản lý Khách hàng
  + Cập nhật – Sao lưu dữ liệu

# 1.3. Đối tượng sử dụng và vai trò của mỗi đối tượng:

* + Chủ cửa hàng: Quản lý nhân viên và nhân báo cáo từ nhân viên.
  + Nhân viên cửa hàng: Quản lý hóa đơn, bán hàng, nhập nông sản.

# 2. Đặc tả yêu cầu:

# 2.1. Quản lý nhập nông sản:

Nhân viên nhận hàng từ nơi sản xuất, nhập và ghi thông tin của hàng hóa. Các thông tin nhân viên gồm có: Danh mục sản phẩm, Tên sản phẩm, Số lượng, Đơn giá,…

Thao tác gồm có: Thêm, xóa, sửa và in hóa đơn.

# 2.2. Quản lý nông sản:

Nông sản sẽ được thêm vào cửa hàng nhập hàng mới về. Đồng thời sẽ xóa đi, hoặc sửa theo ý muốn của nhân viên.

Thông tin về nông sản sẽ bao gồm: Danh mục (phân loại), tên nông sản, giá, số lượng, nhà sản xuất,…

# 2.3. Quản lý Bán nông sản:

Nông sản khi được nhập vào kho sẽ được đưa vào menu khách hàng mua và chọn.

Số lượng bán sẽ được in hóa đơn cho khách hàng.

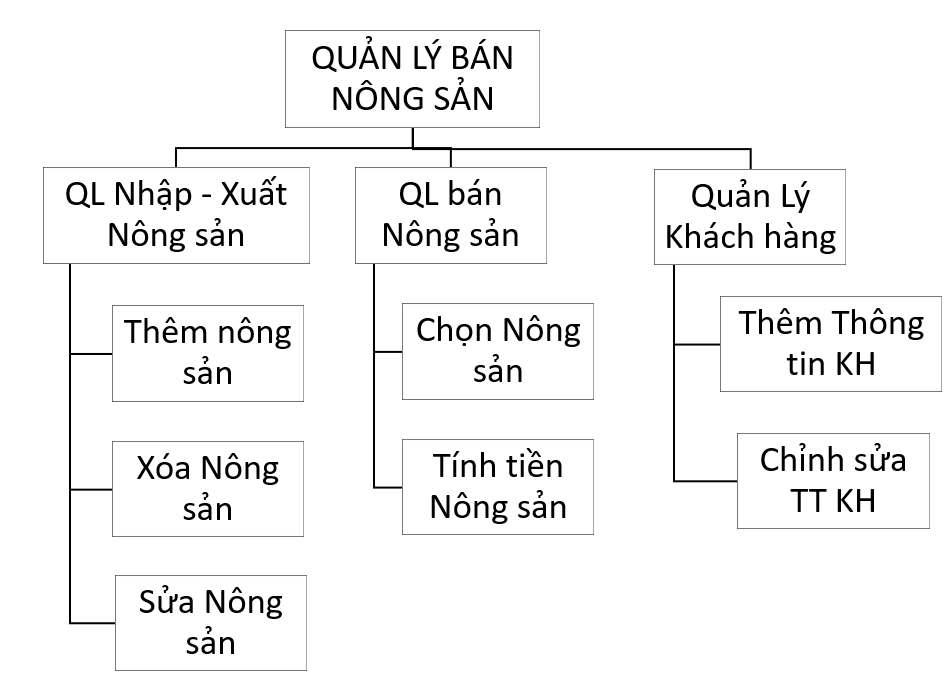
# 2.4. Quản lý tài khoản khách hàng:

Khách hàng sau khi được đăng kí và tạo khách thành viên sẽ được ghi tên vào Cơ sở dữ liệu của cửa hàng.

Mỗi tài khoản sẽ gồm: Tên khách thành viên, ngày sinh, địa chỉ, ….

# III. Thiết kế cơ sở dữ liệu

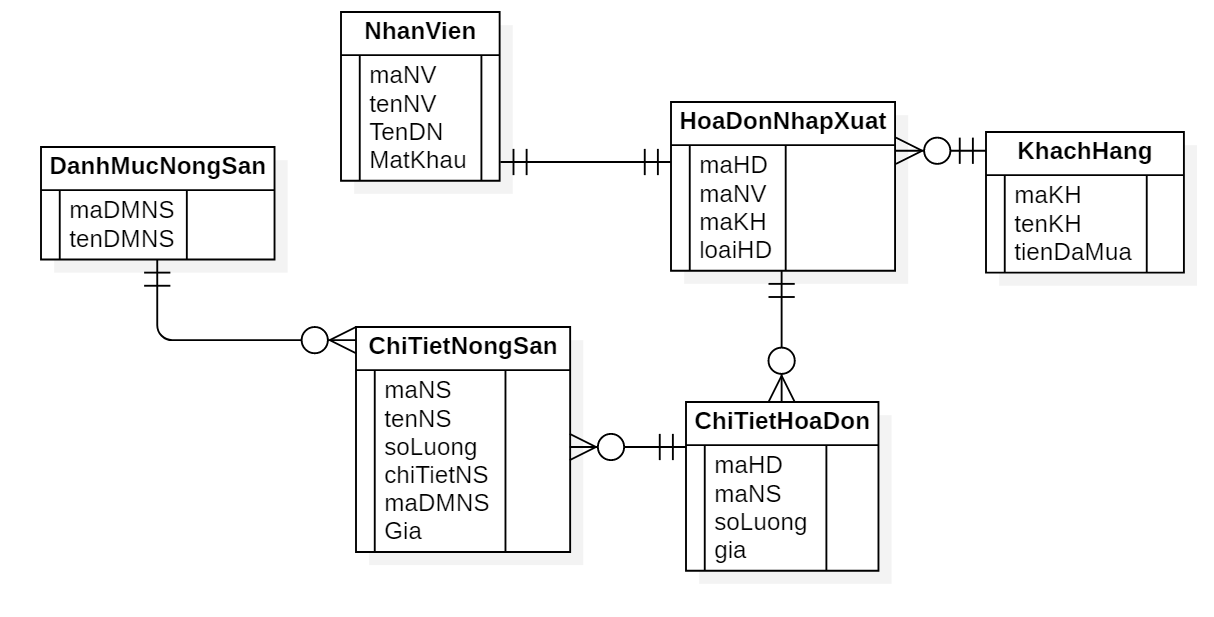
# 1. Mô hình phân cấp chức năng



# 2. Xác định các thực thể

|  |  |
| --- | --- |
| Tên thực thể | Thuộc tính |
| Nhân viên | * Mã nhân viên * Họ tên * Tên đăng nhập * Mật khẩu |
| Khách hàng | * Mã khách hàng * Tên khách hàng * Số tiền đã dùng * Năm sinh * Địa chỉ |
| Danh mục nông sản | * Mã danh mục * Tên danh mục |
| Chi tiết nông sản | * Mã nông sản * Tên nông sản * Số lượng * Chi tiết * Mã danh mục * Giá nông sản |
| Hóa đơn Nhập xuất | * Mã hóa đơn * Mã nhân viên * Mã khách hàng * Loại hóa đơn |
| Chi tiết hóa đơn | * Mã hóa đơn * Mã nông sản * Số lượng * Đơn giá |

# 3. Mô hình dữ liệu mức quan hệ

4. Các bảng dữ liệu

# 4.1. Bảng Nhân viên (tableNhanVien)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Ràng buộc | Khóa chính/ phụ | Mô tả |
| 1 | maNV | Varchar(7) | Not null | Primary key | Mã nhân viên |
| 2 | tenNV | Nvarchar(50) | Not null |  | Tên nhân viên |
| 3 | tenDN | Varchar(25) | Not null |  | Tên đảng nhập |
| 4 | matKhau | Varchar(25) | Not null |  | Mật khẩu |

# 4.2. Bảng Khách hàng (tableKhachHang)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Ràng buộc | Khóa chính/ phụ | Mô tả |
| 1 | maKH | Varchar(7) | Not null | Primary key | Mã khách hàng |
| 2 | tenKH | Nvarchar(50) | Not null |  | Tên khách hàng |
| 3 | soTienDaDung | int | null | Default 0 | Số tiền đã dùng |
| 4 | namSinh | int | Null |  | Năm sinh |
| 5 | diaChi | Nvarchar(50) | Null |  | Địa chỉ |

# 4.3. Bảng Danh mục sản phẩm (tableDanhMucNongSan)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Ràng buộc | Khóa chính/ phụ | Mô tả |
| 1 | maDM | Varchar(7) | Not null | Primary key | Mã danh mục |
| 2 | tenDM | Nvarchar(50) | Not null |  | Tên danh mục |

# 4.4. Bảng Chi tiết danh mục (tableChiTietNongSan)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Ràng buộc | Khóa chính/ phụ | Mô tả |
| 1 | maNS | Varchar(7) | Not null | Primary key | Mã CT nông sản |
| 2 | tenNS | Nvarchar(50) | Not null |  | Tên CT nông sản |
| 3 | soLuong | int | null | Default 0 | Số lượng |
| 4 | chiTiet | Nvarchar(50) | Null |  | Chi tiết |
| 5 | maDMNS | Varchar(7) | Not null |  | Mã danh mục nông sản |
| 6 | gia | int | null |  | Giá tiền |

# 4.5. Bảng Hóa đơn nhập xuất (tableHoaDonNhapXuat)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Ràng buộc | Khóa chính/ phụ | Mô tả |
| 1 | maHD | Varchar(7) | Not null | Primary key | Mã hóa đơn |
| 2 | maNV | varchar(7) | Not null |  | Mã nhân viên |
| 3 | maKH | varchar(7) | null | Default 0 | Mã khách hàng |
| 4 | loaiHD | varchar(1) | Null |  | Loại hóa đơn |

# 4.6. Bảng Chi tiết hóa đơn (tableChiTietHoaDon)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Ràng buộc | Khóa chính/ phụ | Mô tả |
| 1 | maHD | Varchar(7) | Not null | Primary key | Mã hóa đơn |
| 2 | maNS | varchar(7) | Not null |  | Mã nông sản |
| 3 | soLuong | int | null | Default 0 | Số lượng |
| 4 | donGia | int | Null | Default 0 | Đơn giá |

# IV. Thiết kế và cài đặt chương trình

# 1. Các Module chính

# 1.1. Form Đăng nhập:

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

using QuanLyNongSan.Model;

namespace QuanLyNongSan

{

public partial class Form1 : Form

{

static public Form2 form2;

public Form1()

{

InitializeComponent();

}

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

{

DangNhap DN = new DangNhap();

String maNV = DN.Login(textBoxTenDN.Text, textBoxMatKhau.Text);

if (!maNV.Equals(""))

{

form2 = new Form2();

form2.maNV = maNV;

form2.Show();

}

else

{

MessageBox.Show("Đăng Nhập Thất Bại", "Thông Báo");

}

Console.WriteLine(DN.Login(textBoxTenDN.Text, textBoxMatKhau.Text));

}

private void Form1\_Load(object sender, EventArgs e)

{

}

private void textBoxTenDN\_TextChanged(object sender, EventArgs e)

{

}

}

}

# 1.2. Form Menu:

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

namespace QuanLyNongSan

{

public partial class Form2 : Form

{

static public Form3 form3;

static public Form4 form4;

public static String nameEmployee;

public String maNV = "";

public Form2()

{

InitializeComponent();

}

private void Form2\_Load(object sender, EventArgs e)

{

}

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

{

form3 = new Form3();

form3.Show();

}

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

{

form4 = new Form4();

form4.Show();

}

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

{

Form5 form5 = new Form5();

form5.Show();

}

}

}

# 1.3. Form Quản lý nông sản:

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

using System.Xml;

using QuanLyNongSan.Model;

namespace QuanLyNongSan

{

public partial class Form3 : Form

{

XMLFile XmlFile = new XMLFile();

XmlNodeList nodeListDM;

XmlNodeList nodeListCTNS;

int stt = 0;

public Form3()

{

InitializeComponent();

}

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

{

stt = 0;

HoaDon hd = new HoaDon();

hd.ThemHoaDon(dataGridView1, textBoxNhaCungCap.Text,"N");

MessageBox.Show("Thêm Nông Sản Thành Công", "Thông Báo");

dataGridView1.Rows.Clear();

}

private void Form3\_Load(object sender, EventArgs e)

{

DanhMucNongSan DMNS = new DanhMucNongSan();

nodeListDM = DMNS.getListMD();

foreach (XmlNode x in nodeListDM)

{

comboBoxLoaiNS.Items.Add(x.ChildNodes[1].InnerText);

}

}

private void comboBoxTenNongSan\_SelectedIndexChanged(object sender, EventArgs e)

{

}

private void comboBoxLoaiNS\_SelectedIndexChanged(object sender, EventArgs e)

{

try

{

ChiTietNongSan CTNS = new ChiTietNongSan();

nodeListCTNS = CTNS.getListName();

String maNS = "";

comboBoxTenNongSan.Items.Clear();

foreach (XmlNode x in nodeListDM)

{

if (x.ChildNodes[1].InnerText.Equals(comboBoxLoaiNS.SelectedItem.ToString()))

maNS = x.ChildNodes[0].InnerText;

}

foreach (XmlNode x in nodeListCTNS)

{

if (x.ChildNodes[4].InnerText.Equals(maNS))

comboBoxTenNongSan.Items.Add(x.ChildNodes[1].InnerText+" "+x.ChildNodes[0].InnerText);

}

}

catch { }

}

private void dataGridView1\_MouseCaptureChanged(object sender, EventArgs e)

{

comboBoxLoaiNS.SelectedIndex = comboBoxLoaiNS.Items.IndexOf(dataGridView1.CurrentRow.Cells[2].FormattedValue.ToString());

comboBoxTenNongSan.SelectedIndex = comboBoxTenNongSan.Items.IndexOf(dataGridView1.CurrentRow.Cells[1].FormattedValue.ToString() + dataGridView1.CurrentRow.Cells[6].FormattedValue.ToString());

textBoxDonGia.Text = dataGridView1.CurrentRow.Cells[3].FormattedValue.ToString();

textBoxDonGia.Text = dataGridView1.CurrentRow.Cells[4].FormattedValue.ToString();

}

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

{

try

{

if (textBoxNhaCungCap.Text.Equals(""))

{

int a = int.Parse(textBoxNhaCungCap.Text);

}

int soLuong = int.Parse(textBoxSoLuong.Text);

int donGia = int.Parse(textBoxDonGia.Text);

int tong = soLuong \* donGia;

dataGridView1.Rows.Add(++stt, comboBoxTenNongSan.SelectedItem.ToString().Substring(0, comboBoxTenNongSan.SelectedItem.ToString().Length - 7), comboBoxLoaiNS.SelectedItem.ToString(), soLuong, donGia, tong,

comboBoxTenNongSan.SelectedItem.ToString().Substring(comboBoxTenNongSan.SelectedItem.ToString().Length-7));

textBoxSoLuong.Text = "";

textBoxDonGia.Text = "";

}

catch {

MessageBox.Show("Thêm Nông Sản Thất Bại", "Thông Báo");

}

}

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

{

dataGridView1.Rows.RemoveAt(dataGridView1.CurrentRow.Index);

}

private void groupBox1\_Enter(object sender, EventArgs e)

{

}

}

}

# 1.4. Form Quản lý bán hàng – Thanh toán:

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

using QuanLyNongSan.Model;

using System.Xml;

namespace QuanLyNongSan

{

public partial class Form4 : Form

{

XMLFile XmlFile = new XMLFile();

XmlDocument XDoc;

XmlDocument XDocKhachHang;

int sTT = 0;

public Form4()

{

InitializeComponent();

}

void loadTable() {

dataGridView1.Rows.Clear();

XDoc = XmlFile.getXmlDocument("ChiTietNongSans.xml");

XmlNodeList nodeList = XDoc.SelectNodes("/ChiTietNongSans/ChiTietNongSan");

foreach (XmlNode x in nodeList)

{

dataGridView1.Rows.Add(x.ChildNodes[0].InnerText, x.ChildNodes[1].InnerText, x.ChildNodes[3].InnerText, x.ChildNodes[2].InnerText, x.ChildNodes[5].InnerText);

}

}

private void Form4\_Load(object sender, EventArgs e)

{

textBoxMaKhachHang.Text = "";

loadTable();

}

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

{

try

{

int soLuongMua = int.Parse(textBoxSoLuong.Text);

if ( soLuongMua > 0 && int.Parse(textBoxSoLuong.Text) <= int.Parse(dataGridView1.CurrentRow.Cells[3].FormattedValue.ToString()))

{

dataGridView2.Rows.Add(

++sTT,

dataGridView1.CurrentRow.Cells[0].FormattedValue.ToString(),

dataGridView1.CurrentRow.Cells[1].FormattedValue.ToString(),

soLuongMua,

dataGridView1.CurrentRow.Cells[4].FormattedValue.ToString(),

(soLuongMua \* int.Parse(dataGridView1.CurrentRow.Cells[4].FormattedValue.ToString()))

);

}

else

MessageBox.Show("Đầu Vào Sao","Thông Báo");

}

catch { }

capNhatTongTien();

}

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

{

try { dataGridView2.Rows.RemoveAt(dataGridView2.CurrentRow.Index); }

catch{}

capNhatTongTien();

}

void capNhatTongTien() {

int tongTien = 0;

for (int i = 0; i < dataGridView2.Rows.Count - 1; i++) {

tongTien += int.Parse(dataGridView2.Rows[i].Cells[5].Value.ToString());

}

labelTongTien.Text = tongTien.ToString();

}

private void dataGridView1\_MouseCaptureChanged(object sender, EventArgs e)

{

Image image = Image.FromFile("imgs/"+dataGridView1.CurrentRow.Cells[0].Value.ToString()+".jpg");

pictureBox1.Image = image;

}

private void textBoxMaKhachHang\_TextChanged(object sender, EventArgs e)

{

try

{

XDocKhachHang = XmlFile.getXmlDocument("KhachHangs.xml");

XmlNodeList nodeList = XDocKhachHang.SelectNodes("/KhachHangs/KhachHang[maKH ='" + textBoxMaKhachHang.Text + "']");

if (nodeList.Count != 0)

{

labelDanhSachNS.Text = nodeList[0].ChildNodes[1].InnerText;

}

}

catch { }

}

private void dataGridView2\_CellValueChanged(object sender, DataGridViewCellEventArgs e)

{

try {

dataGridView2.CurrentRow.Cells[5].Value = (int.Parse(dataGridView2.CurrentRow.Cells[3].Value.ToString()) \* int.Parse(dataGridView2.CurrentRow.Cells[4].Value.ToString())).ToString();

capNhatTongTien();

}

catch { }

}

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

{

List<XmlNode> nodeList = new List<XmlNode>();

XmlDocument XDoc = XmlFile.getXmlDocument("ChiTietHoaDons.xml");

for (int i = 0; i < dataGridView2.Rows.Count - 1; i++)

{

XmlElement node = XDoc.CreateElement("ChiTietHoaDon");

XmlElement maNS = XDoc.CreateElement("maNS");

Console.WriteLine(i);

maNS.InnerText = dataGridView2.Rows[i].Cells[1].Value.ToString();

XmlElement soLuong = XDoc.CreateElement("soLuong");

soLuong.InnerText = dataGridView2.Rows[i].Cells[3].Value.ToString();

XmlElement donGia = XDoc.CreateElement("DonGia");

donGia.InnerText = dataGridView2.Rows[i].Cells[4].Value.ToString(); ;

node.AppendChild(maNS);

node.AppendChild(soLuong);

node.AppendChild(donGia);

nodeList.Add(node);

}

HoaDon hoaDon = new HoaDon();

String maKH = textBoxMaKhachHang.Text;

if (maKH.Equals(""))

maKH = "KH00000";

Console.WriteLine(maKH);

XmlNodeList n = XDocKhachHang.SelectNodes("/KhachHangs/KhachHang[maKH = '" + maKH + "']");

n[0].ChildNodes[2].InnerText = (int.Parse(n[0].ChildNodes[2].InnerText) + int.Parse(labelTongTien.Text)).ToString();

XDocKhachHang.Save("KhachHangs.xml");

hoaDon.add(XDoc, nodeList, maKH, "X");

loadTable();

}

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

{

}

}

# 1.5. Form Quản lý khách hàng:

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

using QuanLyNongSan.Model;

using System.Xml;

namespace QuanLyNongSan

{

public partial class Form5 : Form

{

XMLFile XmlFile = new XMLFile();

int stt = 0;

public Form5()

{

InitializeComponent();

}

void capNhatBang() {

stt = 0;

dataGridView1.Rows.Clear();

XmlDocument XDoc = XmlFile.getXmlDocument("KhachHangs.xml");

XmlNodeList nodeList = XDoc.SelectNodes("/KhachHangs/KhachHang");

foreach (XmlNode x in nodeList)

{

dataGridView1.Rows.Add(++stt,

x.ChildNodes[1].InnerText,

x.ChildNodes[0].InnerText,

x.ChildNodes[2].InnerText,

x.ChildNodes[3].InnerText,

x.ChildNodes[4].InnerText);

}

}

private void Form5\_Load(object sender, EventArgs e)

{

capNhatBang();

}

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

{

try {

if (textBoxTenKH.Text.Equals("") && textBoxNamSinh.Text.Equals("") && textBoxDiaChi.Text.Equals(""))

{

MessageBox.Show("Nhập THiếu Thông Tin \nVui Lòng Nhập Đày Đủ");

}

else {

int namSinh = int.Parse(textBoxNamSinh.Text);

KhachHang kh = new KhachHang();

if (kh.themKhachHang(textBoxTenKH.Text, namSinh, textBoxDiaChi.Text))

MessageBox.Show("Thêm Khách Hàng Thành Công","Thông Báo");

capNhatBang();

}

}

catch{

MessageBox.Show("Có gì đó sai sai");

}

}

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

{

KhachHang kh = new KhachHang();

if(kh.suaThongTin(dataGridView1.CurrentRow.Cells[2].Value.ToString(),

textBoxTenKH.Text,int.Parse(textBoxNamSinh.Text),textBoxDiaChi.Text))

MessageBox.Show("Đã Sửa Thông Tin Thành Công","Thông Báo");

else

MessageBox.Show("Sửa Thông Tin ĐÃ Thất Bại","Thông Báo");

capNhatBang();

}

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

{

DialogResult dialogResult = MessageBox.Show("Bạn Có Chắc Chắc Về Việc Này", "Thông Báo", MessageBoxButtons.YesNo);

if (dialogResult == DialogResult.Yes) {

try

{

KhachHang kh = new KhachHang();

if(kh.xoaThongTin(dataGridView1.CurrentRow.Cells[2].Value.ToString()))

capNhatBang();

}

catch { }

}

}

private void dataGridView1\_MouseCaptureChanged(object sender, EventArgs e)

{

try

{

textBoxTenKH.Text = dataGridView1.CurrentRow.Cells[1].Value.ToString();

textBoxNamSinh.Text = dataGridView1.CurrentRow.Cells[4].Value.ToString();

textBoxDiaChi.Text = dataGridView1.CurrentRow.Cells[5].Value.ToString();

}

catch { }

}

private void textBoxTenKH\_TextChanged(object sender, EventArgs e)

{

}

}

}

# 1.6. Form Quản lý – Sao lưu dữ liệu:

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

namespace QuanLyNongSan

{

public partial class FormSaoLuuData : Form

{

public FormSaoLuuData()

{

InitializeComponent();

}

private void buttonXML\_SQL\_Click(object sender, EventArgs e)

{

BackUp bk = new BackUp();

bk.BackUpData();

}

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

{

Restore rs = new Restore();

rs.RestoreData();

}

}

}

# 2. Giao diện chương trình:

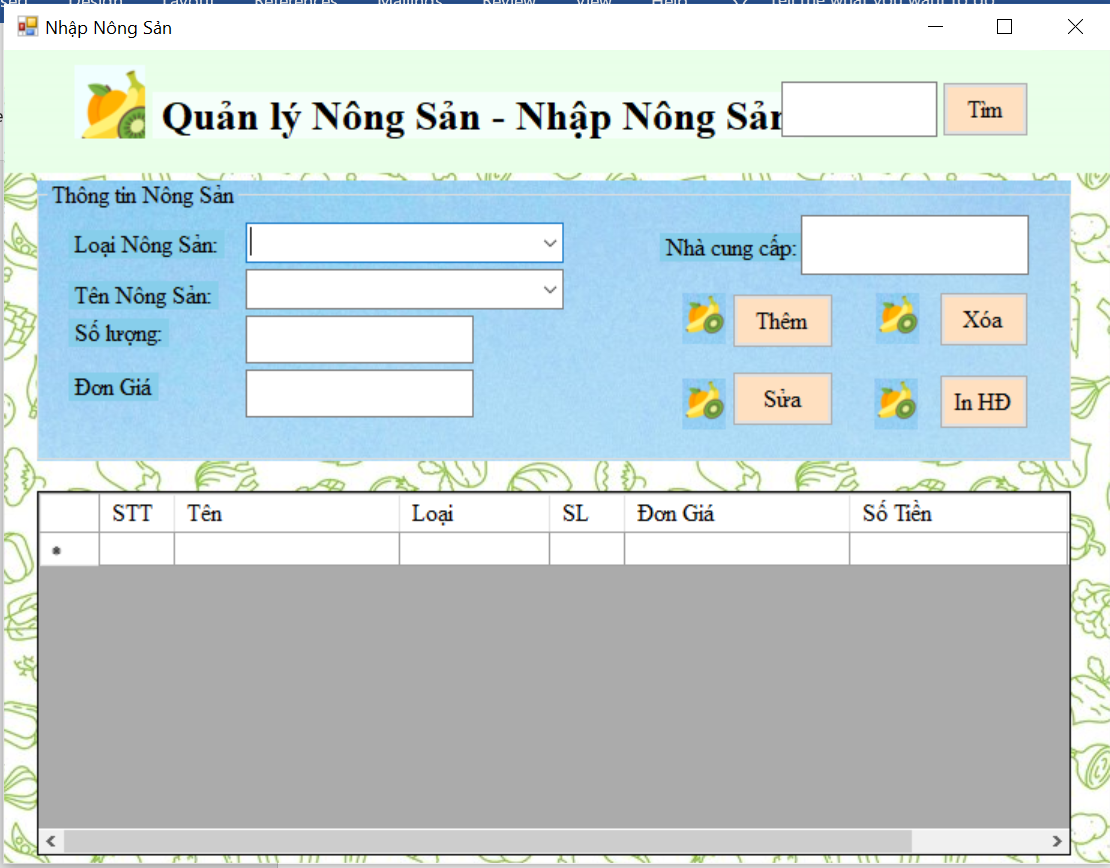
# 2.1. Form Đăng nhập:



# 2.2. Form Menu:



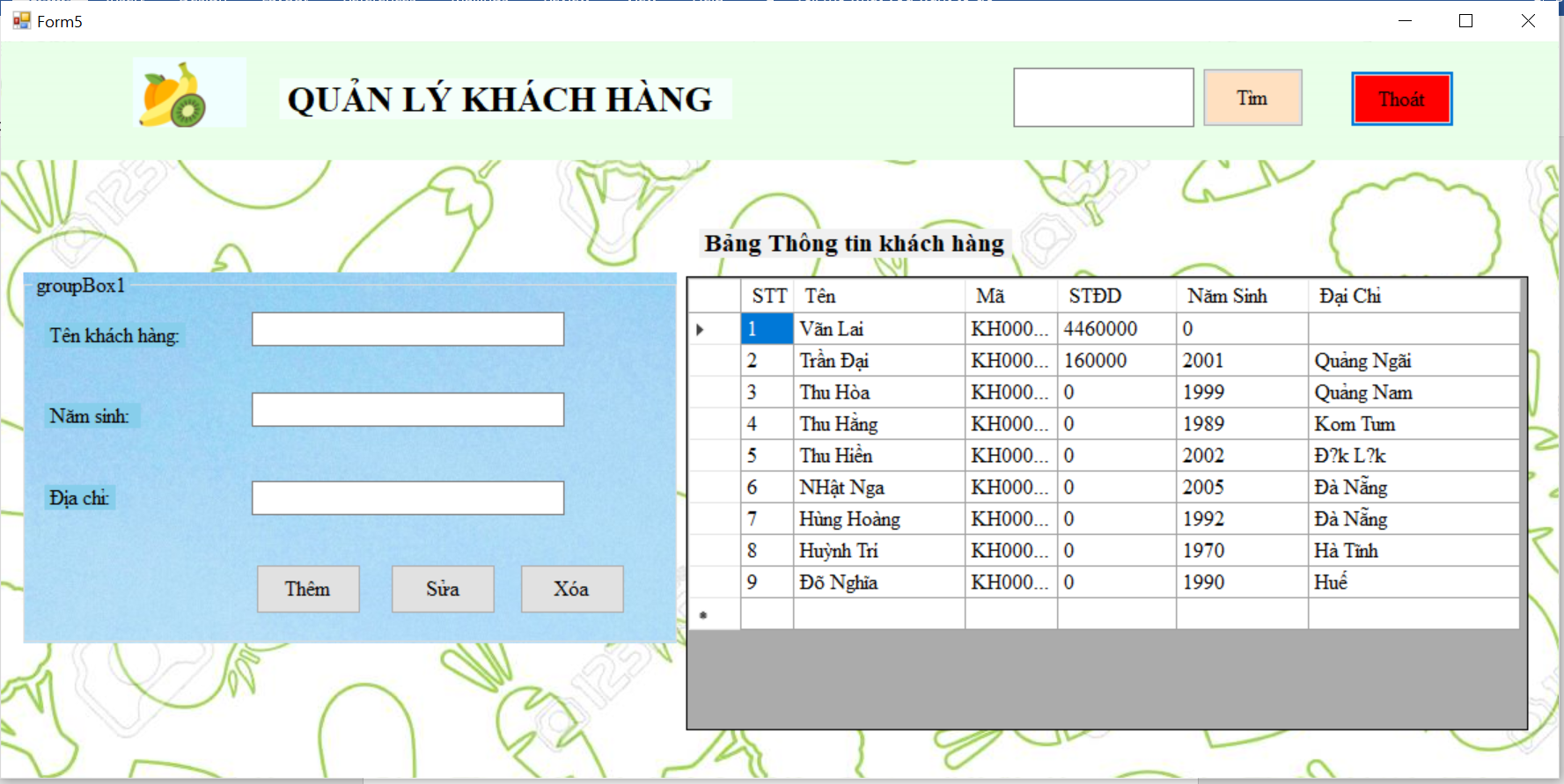
# 2.3. Form Nhập – Quản lý nông sản:



# 2.4. Form Bản nông sản:



# 2.5. Form Quản lý khách hàng:



# 2.6. Form Quản lý Sao lưu dữ liệu