# LỜI MỞ ĐẦU

Chúng ta đang sống trong một kỷ nguyên mà mọi thành tựu khoa học và công nghệ đều xuất hiện một cách hết sức mau lẹ và cũng được đổi mới một cách cực kì nhanh chóng. Nghành công nghệ thông tin cũng ngày càng chứng tỏ được vị trí của mình bằng sự xuất hiện trong rất nhiều các lĩnh vực, mọi nơi, mọi chốn của đời sống. Có thể nói, nền kinh tế toàn cầu đang chịu ảnh hưởng sâu sắc từ công nghệ thông tin mà trong đó có không ít vai trò của các ngành công nghệ phần mềm. Các phần mềm làm ra được ứng dụng phục vụ cho công viêc, hoạt động của các công ty, tổ chức như : các chương trình quản lý nhân sự, quản lý kho, quản lý bến bãi, quản lý hệ thống bán hàng… ngày càng nhiều .Và chúng đều thể hiện được tầm quan trọng của mình đối với tổ chức.

Tuy nhiên, để làm ra được một phần mềm hoàn hảo và đáp ứng được mọi nhu cầu của người dùng thì không phải lài dễ. Nó cần sự hỗ trợ rất lớn của các chuyên ngành trong lĩnh vực công nghệ thông tin. Trong đó, Công nghệ XML là một công nghệ có đóng góp tương đối lớn trong việc hỗ trợ hoàn thành các sản phầm đó. Vì vậy, mà nhóm chúng em quyết định áp dụng công nghệ XML vào trong việc xây dựng chương trình “Quản lý bán đĩa” trong đề tài bài tập lớn được giao. Qua đây chúng em xin bày tỏ lòng cảm ơn sâu sắc tới thầy Đỗ Ngọc Sơn, người đã nhiệt tình chỉ bảo cho chúng em trong quá trình thực hiện đề tài. Chúng em xin chân thành cảm ơn!

Nhóm sinh viên thực hiện!

# MỤC LỤC

[**LỜI MỞ ĐẦU 1**](#_Toc365462929)

[**MỤC LỤC 2**](#_Toc365462930)

[**Chương 1: Giới thiệu 3**](#_Toc365462931)

[**I. Ích Lợi Khi Sử Dụng XML 3**](#_Toc365462932)

[**II. Mục tiêu 3**](#_Toc365462933)

[**III. Phạm vi đề tài, yêu cầu 3**](#_Toc365462934)

[**Chương 2: Nội dung 4**](#_Toc365462935)

[**I. Khảo Sát Nghiệp Vụ 4**](#_Toc365462936)

[**1. Khảo sát và đánh giá hiện trạng 4**](#_Toc365462937)

[**2. Mô hình hoạt động nghiệp vụ của quản lý bán đĩa phim 4**](#_Toc365462938)

[**II. Phân tích thiết kế hệ thống 6**](#_Toc365462939)

[**1. Mô tả vấn đề 6**](#_Toc365462940)

[**2. Đặc tả yêu cầu 6**](#_Toc365462941)

[**III. Thiết kế cơ sở dữ liệu 8**](#_Toc365462942)

[**1. Mô hình phân cấp chức năng 8**](#_Toc365462943)

[**2. Mô hình luồng dữ liệu mức khung cảnh 8**](#_Toc365462944)

[**3. Mô hình dữ liệu mức đỉnh 9**](#_Toc365462945)

[**4. Xác định các thực thể 9**](#_Toc365462946)

[**5. Mô hình dữ liệu mức quan hệ 11**](#_Toc365462947)

[**6. Các bảng dữ liệu 11**](#_Toc365462948)

[**IV. Thiết kế và cài đặt chương trình 13**](#_Toc365462949)

[**1. Các module chính 13**](#_Toc365462950)

[**2. Giao diện chương trình 22**](#_Toc365462951)

[**Chương 3: Kết luận 26**](#_Toc365462952)

# Chương 1: Giới thiệu

## Ích Lợi Khi Sử Dụng XML

* XML có thể tách rời dữ liệu,sử dụng XML dữ liệu sẽ được lưu trữ trong những tập tin XML riêng biệt.
* XML có thể mô tả thông tin của những đối tượng phức tạp mà CSDL quan hệ không thể giải quyết được
* XML dùng để chia sẻ dữ liệu với những tập tin văn bản dễ hiểu
* XML có thể dùng để chuyển đổi dữ liệu giữa các hệ thống không tương thích
* XML dùng để lưu trữ dữ liệu làm cho dữ liệu của chúng ta hữu ích hơn

## Mục tiêu

* + Tìm hiểu đề tài, khảo sát thực trạng nhu cầu và hoạt động quản lý bán đĩa của một số công ty trong thời điểm hiện tại.
  + Thiết kế phần mềm quản lý bán đĩa với mục tiêu hướng người sử dụng và hiệu quả: dễ sử dụng, quản lý tốt, giao diện bắt mắt, thao tác nhanh, dễ dàng….

## Phạm vi đề tài, yêu cầu

Phần mềm được xây dựng trên phạm vi môn học “ Công Nghệ XML”

Với editor là Visual Studio 2010 và cơ sở dữ liệu dạng file XML.

+ Yêu cầu:

* Cơ sở dữ liệu
* XSD (lược đồ)
* XML
* XSLT
* Form DataBase > xml
* Thêm
* Sửa xóa
* Hiển thị trình duyệt
* Cập nhật dữ liệu database
* Kiểm tra hợp lệ

# Chương 2: Nội dung

1. **Khảo Sát Nghiệp Vụ**
2. **Khảo sát và đánh giá hiện trạng**

Mặt hàng đĩa phim là một trong những sản phẩm đặc biệt xuất hiện trong các mặt hàng kinh doanh tại nhiều nơi nhằm đáp ứng nhu cầu sử dụng của mọi tầng lớp trong mọi lĩnh vực, nó bao gồm các loại đĩa như: Đĩa phim hoạt hình, đĩa phim tình cảm tâm lý, đĩa phim hành động…

Vì vậy bài toán quản lý bán đĩa phim là bài toán xử lý kết hợp giữa quản lý danh mục với quản lý xất nhâp. Do sản phẩm sử dụng cho đơ vị kinh doanh nhỏ, mặt khác, mặt hàng kinh doanh chỉ duy nhất một loại băng đĩa là đĩa phim, nên bài toán được xác định nhanh gọn trong việc quản lý danh mục đĩa, và hệ thống bán đĩa chứ không đề cập đến vấn đề đơn vị cung cấp, khách hàng trong chương trình.

Bài toán tập chung vào các vấn đề sau:

* Quản lý nhân viên trong cửa hàng
* Quản lý các hóa đơn bán hàng
* Quản lý các đĩa phim
* Quản lý tài khoản người dùng phần mềm

Yêu cầu chương trình phần mềm:

* Phần mềm phải dễ sử dụng, dễ cài đặt, tạo điều kiện thuận lợi cho người dùng trong việc thao tác trên phần mềm khi sử dụng.
* Phần mềm phải giao tiếp dễ dang với người sử dụng, giao diện thân thiện và thích ứng với các phần cứng của máy tính.
* Phần mềm có phân quyền cho nhân viên và người quản lý trong hệ thống để quản lý có thể toàn quyền kiểm soát.

Yêu cầu dữ liệu:

* Dữ liệu phải được thiết kế chính xác, không được thiếu cho việc quản lý.
* Xử lý được các chức năng trong hệ thống.

1. **Mô hình hoạt động nghiệp vụ của quản lý bán đĩa phim**

* Cơ sở bán băng đĩa cần thực hiện phân nhóm các băng đĩa theo thể loại, thành các nhóm riêng. Cập nhật danh sách các nhóm đĩa vào máy tính, phân vùng các đĩa và trưng bày tại nơi mà khách hàng dễ tìm kiếm để thuận lợi cho việc tra cứu và in cho khách hàng.
* Người kinh doanh phải thường xuyên thiết kế và trưng bày các bảng mục lục danh sách các đĩa, nội dung các đĩa gồm những gì, trợ giúp cho khách hàng đến tra cứu và tìm kiếm đĩa phim muốn mua.
* Người kinh doanh thực hiện các công việc bán hàng, ghi chép sổ sách và nhật ký bán hàng. Hỗ trợ khách hàng trong công việc tra cứu thông tin về đĩa.
* Trên cơ sở dữ liệu sổ sách và thực tế đề ra danh sách các đĩa phim cần phải bổ sung và nhập vào
* Kiểm tra định kỳ danh sách các đĩa về số lượng, số lượng bán được, số lượng nhập và số lương tồn đọng.
* Qua đó, nhân thấy việc sắp xếp, phân loại băng đĩa khá phức tạp, nhiều loại dễ gây nhầm lẫn cho người kinh doanh. Việc quán lý bán đĩa cũng khó xử lý, khó tra cứu. Vì vậy, việc tạo ra phần mềm Quản lý bán đĩa đặt ra cần đáp ứng được yêu cầu nghiệp vụ trên.

1. **Phân tích thiết kế hệ thống**
2. **Mô tả vấn đề**

**-** Với phần mềm phục vụ cho công việc quản lý bán đĩa phim, có thể nhận thấy:

+ Mục đích của hệ thống: tin học hóa việc về dịch vụ bán đĩa cũng như quản lý của cửa hàng.

+ Các chức năng chính:

* Quản lý nhân viên của cửa hàng.
* Quản lý đĩa.
* Quản lý tài khoản người dùng

+ Đối tượng sử dụng và vai trò của mỗi đối tượng:

* Chủ cửa hàng: quản lý nhân viên, nhận báo cáo từ nhân viên.
* Nhân viên: Quản lý hóa đơn và việc bán đĩa

1. **Đặc tả yêu cầu**

* Quản lý nhân viên:

Có thể thực hiện thêm nhân viên mới vào danh sách, sửa đổi khi có những biến đổi xảy ra và xóa bỏ nhân viên khi bị sa thải. Các thông tin về nhân viên gồm có: Mã nhân viên, Tên nhân viên, Điện thoại liên hệ và các mô tả khác.

* Quản lý đĩa phim:

Đĩa phim sẽ được thêm vào khi cửa hàng nhập hàng mới về. Sẽ được xóa đi mỗi khi có đĩa phim được bán. Khi nhân viên muốn tìm kiếm một đĩa phim theo tựa đề, đĩa phim sẽ được cập nhật số lượng và sẽ được sửa mỗi khi có sai xót. Thông tin về đĩa phim bao gồm: Mã đĩa, tựa đề, đạo diễn, hãng sản xuất, số đĩa, giá thành.

* Quản lý hóa đơn:

Mỗi khách hàng đến mua đĩa sẽ được nhận một hóa đơn. Trong đó bao gồm: mã hóa đơn, mã đĩa, số lượng mua, mã nhân viên, ngày lập.

* Quản lý tài khoản:

Mỗi nhân viên sử dụng phần mềm sẽ được cung cấp một tài khoản. Tài khoản bao gồm: Tài khoản, mật khẩu và quyền hạn.

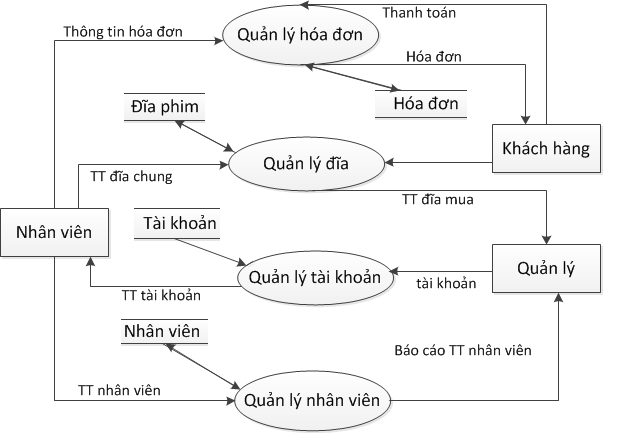
1. **Thiết kế cơ sở dữ liệu**
2. **Mô hình phân cấp chức năng**



1. **Mô hình luồng dữ liệu mức khung cảnh**



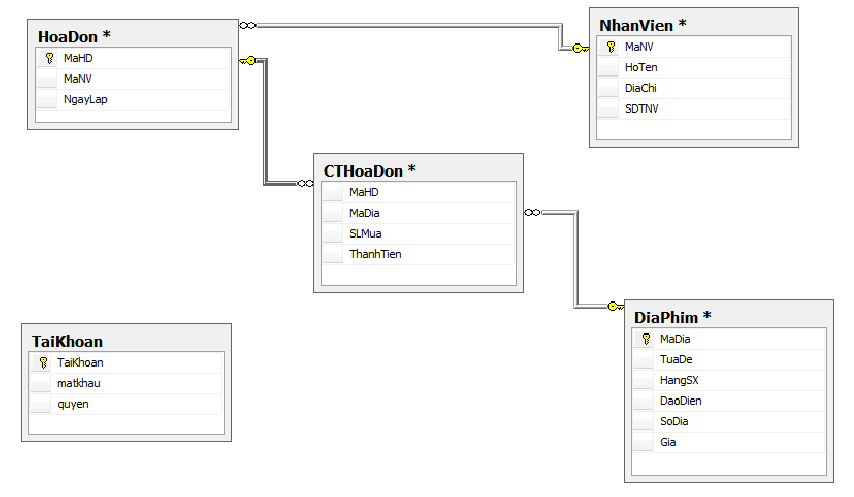
1. **Mô hình dữ liệu mức đỉnh**



1. **Xác định các thực thể**

|  |  |
| --- | --- |
| Tên thực thể | Thuộc tính |
| Nhân viên | * Mã nhân viên * Họ tên * Địa chỉ * Số điện thoại |
| Tài khoản | * Tên tài khoản * Mật khẩu * Quyền hạn |
| Đĩa phim | * Mã đĩa * Tựa đề * Hãng sản xuất * Đạo diễn * Số đĩa * Giá đĩa |
| Hóa đơn | * Mã hóa đơn * Mã nhân viên * Ngày lập |
| Chi tiết hóa đơn | * Mã hóa đơn * Mã đĩa * Số lượng mua * Thành tiền |

1. **Mô hình dữ liệu mức quan hệ**



1. **Các bảng dữ liệu**

Bảng tài khoản:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu(Độ rộng) | Ràng buộc | Khóa chính/  Khóa Phụ  (PK/FK) | Mô tả |
| 1 | TaiKhoan | Nvchar(50) | Not null | PK | Tài khoản |
| 2 | matkhau | Nvchar(50) | Null |  | Mật khẩu |
| 3 | Quyen | Int | Null |  | Quyền hạn |

Bảng nhân viên:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu(Độ rộng) | Ràng buộc | Khóa chính/  Khóa Phụ  (PK/FK) | Mô tả |
| 1 | MaNV | NVChar(5) | Not null | PK | Mã nhân viên |
| 2 | HoTen | NVChar(100) | Not null |  | Tên nhân viên |
| 3 | DiaChi | NVChar(150) | Not null |  | Địa chỉ |
| 4 | SDTNV | NVChar(11) | Not null |  | SDT nhân viên |

Bảng đĩa phim:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu(Độ rộng) | Ràng buộc | Khóa chính/  Khóa Phụ  (PK/FK) | Mô tả |
| 1 | MaDia | NVChar(5) | Not null | PK | Mã đĩa |
| 2 | TuaDe | NVChar(250) | Null |  | Tựa đề |
| 3 | HangSX | NVChar(150) | Null |  | Hãng SX |
| 4 | DaoDien | NVChar(100) | Null |  | Đạo diễn |
| 5 | SoDia | Int | Null |  | Số đĩa |
| 6 | Gia | Decimal(18,0) | Null |  | Giá |

Bảng hóa đơn:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu(Độ rộng) | Ràng buộc | Khóa chính/  Khóa Phụ  (PK/FK) | Mô tả |
| 1 | MaKH | NChar(10) | Not null | PK | Mã khách hàng |
| 2 | TenKH | NChar(100) | Not null |  | Tên KH |
| 3 | Email | NChar(150) | Not null |  | Email |
| 4 | GioiTinh | NChar(10) | Not null |  | Giới tính |
| 5 | SDT | NChar(11) | Null |  | Số điện thoại |
| 6 | TrangThai | NChar(20) | Not null |  | Trạng thái |

Bảng chi tiết hóa đơn:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu(Độ rộng) | Ràng buộc | Khóa chính/  Khóa Phụ  (PK/FK) | Mô tả |
| 1 | MaKH | NChar(10) | Not null | PK | Mã khách hàng |
| 2 | TenKH | NChar(100) | Not null |  | Tên KH |
| 3 | Email | NChar(150) | Not null |  | Email |
| 4 | GioiTinh | NChar(10) | Not null |  | Giới tính |

1. **Thiết kế và cài đặt chương trình**
2. **Các module chính**

Class TaoXML( Gồm phương thức thêm, sửa, xóa , tìm kiếm, kiểm tra và hàm tạo XML, Load lên DataGridview)

|  |
| --- |
| class TaoXML  {  string strCon = " Data Source =.; Initial Catalog = QLDiaPhim; Integrated Security = True";  public void taoXML(string sql, string bang, string \_FileXML)  {  SqlConnection con = new SqlConnection(strCon);  con.Open();  SqlDataAdapter ad = new SqlDataAdapter(sql, con);  DataTable dt = new DataTable(bang);  ad.Fill(dt);  dt.WriteXml(Application.StartupPath + \_FileXML, XmlWriteMode.WriteSchema);  }  public DataTable loadDataGridView(string \_FileXML)  {  DataTable dt = new DataTable();  string FilePath = Application.StartupPath + \_FileXML;  if (File.Exists(FilePath))  {  //tao luong xu ly file xml  FileStream fsReadXML = new FileStream(FilePath, FileMode.Open);  //doc file xml vao datatable  dt.ReadXml(fsReadXML);  fsReadXML.Close();  }  else  {  MessageBox.Show("File không tồn tại");  }  return dt;  }  public void Them(string FileXML, string xml)  {  try  {  XmlTextReader textread = new XmlTextReader(FileXML);  XmlDocument doc = new XmlDocument();  doc.Load(textread);  textread.Close();  XmlNode currNode;  XmlDocumentFragment docFrag = doc.CreateDocumentFragment();  docFrag.InnerXml = xml;  currNode = doc.DocumentElement;  currNode.InsertAfter(docFrag, currNode.LastChild);  doc.Save(FileXML);  }  catch  {  MessageBox.Show("lỗi");  }  }  public void xoa(string \_FileXML, string xml)  {  try  {  string fileName = Application.StartupPath + \_FileXML;  XmlDocument doc = new XmlDocument();  doc.Load(fileName);  XmlNode nodeCu= doc.SelectSingleNode(xml);  doc.DocumentElement.RemoveChild(nodeCu);  doc.Save(fileName);  }  catch  {  MessageBox.Show("lỗi");  }  }  public void sua(string FileXML, string sql, string xml, string bang)  {  XmlTextReader reader = new XmlTextReader(FileXML);  XmlDocument doc = new XmlDocument();  doc.Load(reader);  reader.Close();  XmlNode oldValue;  XmlElement root = doc.DocumentElement;  oldValue = root.SelectSingleNode(sql);  XmlElement newValue = doc.CreateElement(bang);  newValue.InnerXml = xml;  root.ReplaceChild(newValue, oldValue);  doc.Save(FileXML);  }  public void TimKiem(string \_FileXML, string xml, DataGridView dgv)  {  XmlDocument xDoc = new XmlDocument();  xDoc.Load(Application.StartupPath + \_FileXML);  string xPath = xml;  XmlNode node = xDoc.SelectSingleNode(xPath);  DataSet ds = new DataSet();  DataTable dt = new DataTable();  XmlNodeReader nr = new XmlNodeReader(node);  ds.ReadXml(nr);  dgv.DataSource = ds.Tables[0];  nr.Close();  }  public string LayGiaTri(string duongDan, string truongA, string giaTriA, string truongB)  {  string giatriB = "";  DataTable dt = new DataTable();  dt = loadDataGridView(duongDan);  int soDongNhanVien = dt.Rows.Count;  for (int i = 0; i < soDongNhanVien; i++)  {  if (dt.Rows[i][truongA].ToString().Trim().Equals(giaTriA))  {  giatriB = dt.Rows[i][truongB].ToString();  return giatriB;  }  }  return giatriB;  }  public bool KiemTra(string \_FileXML, string truongKiemTra, string giaTriKiemTra)  {  DataTable dt = new DataTable();  dt = loadDataGridView(\_FileXML);  dt.DefaultView.RowFilter = truongKiemTra + " ='" + giaTriKiemTra + "'";  if (dt.DefaultView.Count > 0)  return true;  return false;  }  public string txtMa(string tienTo, string \_FileXML, string tenCot)  {  string txtMa = "";  DataTable dt = new DataTable();  dt =loadDataGridView(\_FileXML);  int dem = dt.Rows.Count;  if (dem == 0)  {  txtMa = tienTo + "001";//HD001  }  else  {  int duoi = int.Parse(dt.Rows[dem - 1][tenCot].ToString().Substring(2, 3)) + 1;  string cuoi = "00" + duoi;  txtMa = tienTo + "" + cuoi.Substring(cuoi.Length - 3, 3);  }  return txtMa;  }  public bool KTMa(string \_FileXML, string cotMa, string ma)  {  bool kt = true;  DataTable dt = new DataTable();  dt = loadDataGridView(\_FileXML);  for (int i = 0; i < dt.Rows.Count; i++)  {  if (dt.Rows[i][cotMa].ToString().Trim().Equals(ma))  {  kt = false;  }  else  {  kt = true;  }  }  return kt;  }  public void exCuteNonQuery(string sql)  {  SqlConnection con = new SqlConnection(strCon);  con.Open();  SqlCommand com = new SqlCommand(sql, con);  com.ExecuteNonQuery();  }  public void Them\_Database(string tenBang, string \_FileXML) {  string duongDan = \_FileXML;  DataTable table = loadDataGridView(duongDan);  int dong = table.Rows.Count - 1;  string sql = "insert into " + tenBang + " values(";  for (int j = 0; j < table.Columns.Count - 1; j++)  {  sql += "N'" + table.Rows[dong][j].ToString().Trim() + "',";  }  sql += "N'" + table.Rows[dong][table.Columns.Count - 1].ToString().Trim() + "'";  sql += ")";  exCuteNonQuery(sql);  }  public void Sua\_Database(string tenBang, string \_FileXML, string tenCot, string giaTri)  {  string duongDan = \_FileXML;  DataTable table = loadDataGridView(duongDan);  int dong = -1;  for (int i = 0; i < table.Rows.Count; i++)  {  if (table.Rows[i][tenCot].ToString().Trim() == giaTri)  { dong = i; }  }  if (dong > -1)  {  string sql = "update " + tenBang + " set ";  for (int j = 0; j < table.Columns.Count-1; j++)  {  sql += table.Columns[j].ToString() + " = N'" + table.Rows[dong][j].ToString().Trim() + "', ";  }  sql += table.Columns[table.Columns.Count - 1].ToString() + " = N'" + table.Rows[dong][table.Columns.Count - 1].ToString().Trim() + "' ";  sql += "where " + tenCot + "= '" + giaTri +"'";  exCuteNonQuery(sql);  }  }  public void Xoa\_Database(string \_FileXML, string tenCot, string giaTri,string tenBang)  {  string duongDan = \_FileXML;  DataTable table = loadDataGridView(duongDan);  int dong = -1;  for (int i = 0; i < table.Rows.Count; i++)  {  if (table.Rows[i][tenCot].ToString().Trim() == giaTri)  { dong = i; }  }  if (dong > -1)  {  string sql = "delete from " + tenBang + " where ";  for (int j = 0; j < table.Columns.Count - 1; j++)  {  if (table.Rows[dong][tenCot].ToString().Trim() == giaTri)  {  sql += tenCot + " = '" + giaTri+ "'";  }  }  exCuteNonQuery(sql);  }  }  public void CapNhapTungBang(string tenBang, string \_FileXML)  {  string duongDan= \_FileXML;  DataTable table= loadDataGridView(duongDan);  for (int i = 0; i < table.Rows.Count; i++)  {  string sql = "insert into " + tenBang + " values(";  for (int j = 0; j < table.Columns.Count - 1; j++)  {  sql += "N'" + table.Rows[i][j].ToString().Trim() + "',";  }  sql += "N'" + table.Rows[i][table.Columns.Count - 1].ToString().Trim() + "'";  sql += ")";  exCuteNonQuery(sql);  }  }  public void TimKiemXSLT(string data,string tenFileXML,string tenfileXSLT)  {  XslCompiledTransform xslt = new XslCompiledTransform();  xslt.Load(""+tenfileXSLT+".xslt");  XsltArgumentList argList = new XsltArgumentList();  argList.AddParam("Data", "", data);  XmlWriter writer = XmlWriter.Create(""+tenFileXML+".html");  xslt.Transform(new XPathDocument("" + tenFileXML + ".xml"), argList, writer);  writer.Close();  System.Diagnostics.Process.Start("" + tenFileXML + ".html");  }  } |

Code tìm kiếm hiển thị lên Web bằng XSLT

Chi tiết hóa đơn

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?>  <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"  xmlns:msxsl="urn:schemas-microsoft-com:xslt" exclude-result-prefixes="msxsl"  >  <xsl:output method="xml" indent="yes"/>  <xsl:param name="Data"></xsl:param>  <xsl:template match="/NewDataSet">  <html>  <body>  <br />  <br />  <center>  <h1>DANH SÁCH CHI TIẾT HÓA ĐƠN</h1>  </center>  <br />  <br />  <table border="1" width="100%">  <tr>  <th>STT</th>  <th>Mã hóa đơn</th>  <th>Mã đĩa</th>  <th>Số lượng mua</th>  <th>Thành tiền</th>  </tr>  <xsl:for-each select="CTHoaDon">  <xsl:if test="MaHD[.=$Data]">  <tr>  <td>  <xsl:value-of select="position()"/>  </td>  <td>  <xsl:value-of select="MaHD"/>  </td>  <td>  <xsl:value-of select="MaDia"/>  </td>  <td>  <xsl:value-of select="SLMua"/>  </td>  <td>  <xsl:value-of select="ThanhTien"/>  </td>  </tr>  </xsl:if>  </xsl:for-each>  </table>  </body>  </html>  </xsl:template>  </xsl:stylesheet> |

XSLTDiaPhim

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?>  <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"  xmlns:msxsl="urn:schemas-microsoft-com:xslt" exclude-result-prefixes="msxsl"  >  <xsl:output method="xml" indent="yes"/>  <xsl:param name="Data"></xsl:param>  <xsl:template match="/NewDataSet">  <html>  <body>  <br />  <br />  <center>  <h1>DANH SÁCH ĐĨA PHIM</h1>  </center>  <br />  <br />  <table border="1" width="100%">  <tr>  <th>STT</th>  <th>Mã đĩa phim</th>  <th>Tựa đề</th>  <th>Hãng sản xuất</th>  <th>Đạo điễn</th>  <th>Số đĩa</th>  <th>Giá</th>  </tr>  <xsl:for-each select="DiaPhim">  <xsl:if test="MaDia[.=$Data]">  <tr>  <td>  <xsl:value-of select="position()"/>  </td>  <td>  <xsl:value-of select="MaDia"/>  </td>  <td>  <xsl:value-of select="TuaDe"/>  </td>  <td>  <xsl:value-of select="HangSX"/>  </td>  <td>  <xsl:value-of select="DaoDien"/>  </td>  <td>  <xsl:value-of select="SoDia"/>  </td>  <td>  <xsl:value-of select="Gia"/>  </td>  </tr>  </xsl:if>  </xsl:for-each>  </table>  </body>  </html>  </xsl:template>  </xsl:stylesheet> |

XSLTDanhSachHoaDon

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?>  <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"  xmlns:msxsl="urn:schemas-microsoft-com:xslt" exclude-result-prefixes="msxsl"  >  <xsl:output method="xml" indent="yes"/>  <xsl:param name="Data"></xsl:param>  <xsl:template match="/NewDataSet">  <html>  <body>  <br />  <br />  <center>  <h1>DANH SÁCH HÓA ĐƠN</h1>  </center>  <br />  <br />  <table border="1" width="100%">  <tr>  <th>STT</th>  <th>Mã hóa đơn</th>  <th>Mã nhân viên</th>  <th>Ngày lập</th>  </tr>  <xsl:for-each select="HoaDon">  <xsl:if test="MaHD[.=$Data]">  <tr>  <td>  <xsl:value-of select="position()"/>  </td>  <td>  <xsl:value-of select="MaHD"/>  </td>  <td>  <xsl:value-of select="MaNV"/>  </td>  <td>  <xsl:value-of select="NgayLap"/>  </td>  </tr>  </xsl:if>  </xsl:for-each>  </table>  </body>  </html>  </xsl:template>  </xsl:stylesheet> |

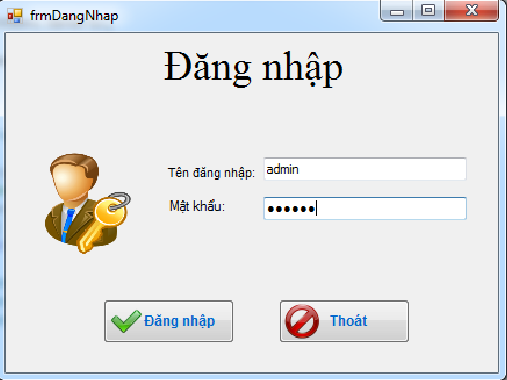
XSLTNhanVien

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?>  <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"  xmlns:msxsl="urn:schemas-microsoft-com:xslt" exclude-result-prefixes="msxsl"  >  <xsl:output method="xml" indent="yes"/>  <xsl:param name="Data"></xsl:param>  <xsl:template match="/NewDataSet">  <html>  <body>  <br />  <br />  <center>  <h1>DANH SÁCH NHÂN VIÊN</h1>  </center>  <br />  <br />  <table border="1" width="100%">  <tr>  <th>STT</th>  <th>Mã nhân viên</th>  <th>Họ và tên</th>  <th>Địa chỉ</th>  <th>số điện thoại</th>  </tr>  <xsl:for-each select="NhanVien">  <xsl:if test="MaNV[.=$Data]">  <tr>  <td>  <xsl:value-of select="position()"/>  </td>  <td>  <xsl:value-of select="MaNV"/>  </td>  <td>  <xsl:value-of select="HoTen"/>  </td>  <td>  <xsl:value-of select="DiaChi"/>  </td>  <td>  <xsl:value-of select="SDTNV"/>  </td>  </tr>  </xsl:if>  </xsl:for-each>  </table>  </body>  </html>  </xsl:template>  </xsl:stylesheet> |

XSLTTaiKhoan

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?>  <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"  xmlns:msxsl="urn:schemas-microsoft-com:xslt" exclude-result-prefixes="msxsl"  >  <xsl:output method="xml" indent="yes"/>  <xsl:param name="Data"></xsl:param>  <xsl:template match="/NewDataSet">  <html>  <body>  <br />  <br />  <center>  <h1>DANH SÁCH TÀI KHOẢN</h1>  </center>  <br />  <br />  <table border="1" width="100%">  <tr>  <th>STT</th>  <th>Tên tài khoản</th>  <th>Mật khẩu</th>  <th>quyền</th>  </tr>  <xsl:for-each select="TaiKhoan">  <xsl:if test="TaiKhoan[.=$Data]">  <tr>  <td>  <xsl:value-of select="position()"/>  </td>  <td>  <xsl:value-of select="TaiKhoan"/>  </td>  <td>  <xsl:value-of select="matkhau"/>  </td>  <td>  <xsl:value-of select="quyen"/>  </td>  </tr>  </xsl:if>  </xsl:for-each>  </table>  </body>  </html>  </xsl:template>  </xsl:stylesheet> |

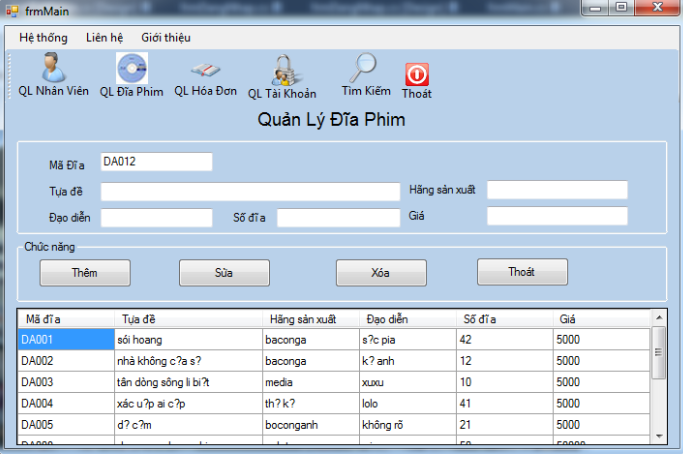
1. **Giao diện chương trình**

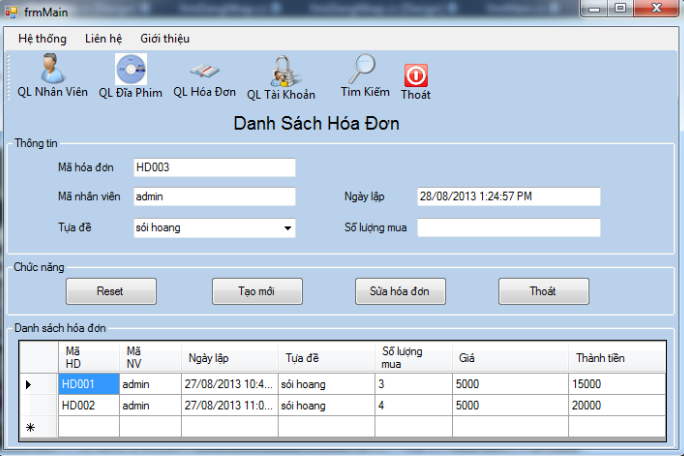


Form đăng nhập

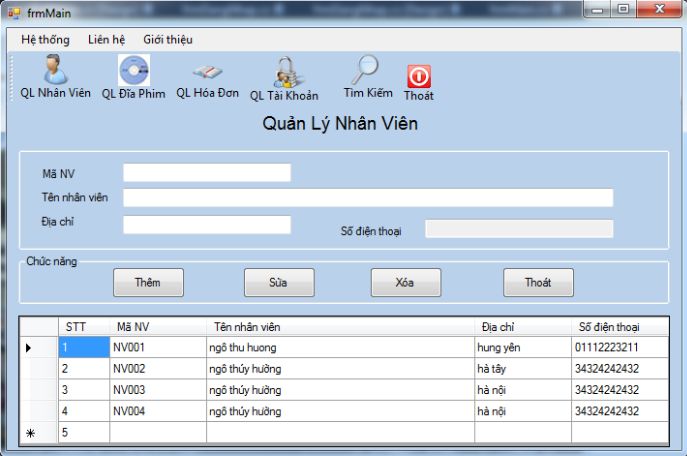


Form main

 Form quản lý đĩa phim



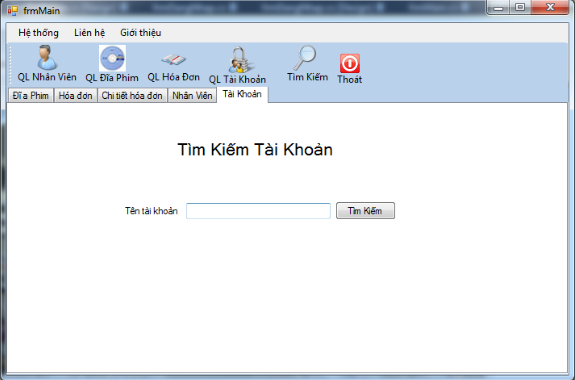
Form danh sách hóa đơn



Form quản lý nhân viên



Form quản lý tài khoản



Form tìm kiếm hiển thị lên trình duyệt

**Chương 3: Kết luận**

Đây là lần đầu tiên nhóm chúng em áp dụng cônghệ XML vào việc thiết kế, xây dựng phần mềm để quản lý việc bán đĩa cho các cửa hàng. Vận dụng kiến thức đã được học ở trường, kết hợp với những tài liệu tham khảo trên mạng , sách báo, và đặc biệt là sự hướng dẫn của thầy, cô nhóm thực hiện chúng em đã hoàn thành bài tập lớn. Qua bài báo chúng em muốn gửi lời cảm ơn giảng viên hướng dẫn Ths Đỗ Ngọc Sơn , rất cảm ơn thầy đã nhiệt tình dạy bảo và hướng dẫn chúng em hoàn thành bài tập lớn này. Vì kiến thức còn hạn chế, điều kiện về thời gian hơi gấp nên bài báo cáo không tránh khỏi thiếu sót .Vì thế, nhóm chúng em rất mong nhận được chỉ dẫn, bổ sung, sửa chữa, những đóng góp và sự cảm thông sâu sắc của thầy, cô cũng như bạn bè để bài tập của chúng em được hoàn thiện hơn.Nhóm chúng em xin chân thành cảm ơn!

BẢNG CÔNG VIỆC

|  |  |  |
| --- | --- | --- |
| **STT** | **Thành viên** | **Công việc thực hiện** |
| 1 | NgôThị Lúa | * Form đĩa phim * Form danh sách hóa đơn * Tìm kiếm * Main |
| 2 | Nguyễn Thị Tuyển | * Form đăng nhập * Tài khoản * Nhân viên |
| 3 | Hoàng Văn Hiệp |  |