**TRƯỜNG ĐẠI HỌC TÀI CHÍNH – MARKETING**

**KHOA CÔNG NGHỆ THÔNG TIN**

**A picture containing drawing

Description automatically generated**

**ĐỒ ÁN HỌC PHẦN**

TÊN HỌC PHẦN: **LẬP TRÌNH WEB (4 tín chỉ)**

MÃ SỐ LỚP HP**: 2111112005703**

**Tên đề tài: Lập trình phát triển WebApp bán vé sự kiện âm nhạc trực tuyến**

**Họ tên sinh viên: Đinh Văn Thành [22]**

**Mã số sinh viên: 1921006822**

**Lớp: 19DTH2 [G3]**

**Ngày nộp: 2/12/2021**

**Ký tên:**

**TP.HCM, ngày 2 tháng 12 năm 2021**

**TRƯỜNG ĐẠI HỌC TÀI CHÍNH – MARKETING**

**KHOA CÔNG NGHỆ THÔNG TIN**

**ĐỀ TÀI ĐỒ ÁN HỌC PHẦN**

TÊN HỌC PHẦN: **LẬP TRÌNH WEB (4 tín chỉ)**

MÃ SỐ LỚP HP**: 2111112005703**

**Tên đề tài: Lập trình phát triển WebApp bán vé sự kiện âm nhạc trực tuyến (online)**

**Giảng viên giảng dạy: ThS. GVC. Võ Xuân Thể**

**Sinh viên thực hiện: Đinh Văn Thành [22]**

**Mã số sinh viên: 1921006822**

**Lớp: 19DTH2 [G3]**

**Tên sản phẩm đề tài : G322DinhVanThanhVeOnline (.Net framework 4.7.2, Visual Studio 2019)**

**Account quản lý hệ thống:**  Username: **Admin**  với Password: **admin@vn**

Username: **Guest**  với Password: **guest@vn**

**Tên DB SQL: 1G322DinhVanThanhVeOnline.bak (MS. SQL 2018)**

**Tên máy chủ DB SQL:**

Local:

MS SQL Server address: .

Tên đăng nhập DB SQL: sa

Mật khẩu (password) đăng nhập SQL: 123

Online somee.com:

MS SQL Server address: 1G322DinhVanThanhVeOnline.mssql.somee.com

Login name: gthreedvthanh\_SQLLogin\_1

Login password: hzp3ww338k

**URL:** http://ticketg322dvthanh.somee.com OR http://www.ticketg322dvthanh.somee.com

**Nhận xét của giảng viên:**

**…………………………..**

**…………………………..**

**Điểm đánh giá:………….(…………………)**

**Ngày……./……../202…..**

**Giảng viên Ký tên**

**TP.HCM, ngày 2 tháng 12 năm 2021**

# LỜI CẢM ƠN

Lời đầu tiên, em xin gửi lời cảm ơn đến toàn thể Quý thầy cô khoa Công nghệ thông tin trường Đại học - Tài Chính Marketing đã giảng dạy và truyền đạt cho chúng em các kiến thức bổ ích. Thầy cô luôn tạo điều kiện thuận lợi cho tôi hoàn thành tốt đề tài. Đặc biệt, em xin chân thành cảm ơn sự hướng dẫn tận tình của Thầy Võ Xuân Thể – người luôn giải thích các thắc mắc, góp ý và sửa chữa những phần còn thiếu sót trong qua trình thực hiện đề tài để bài làm đạt được kết quả tốt nhất. Tuy vậy, do thời gian thực hiện và kinh nghiệm còn hạn chế nên không tránh khỏi những sai sót trong bài làm. Em mong nhận được những ý kiến, nhận xét của Quý thầy cô để em có thể bổ sung, hoàn thiện kiến thức của bản thân.

Cuối cùng, chúc thầy cô trong nhà trường lời chúc sức khỏe và luôn hành công trong sự nghiệp của mình.

TP Hồ Chí Minh, tháng 12 năm 2021

# DANH MỤC CÁC TỪ VIẾT TẮT

|  |  |
| --- | --- |
| CSDL hoặc DB | Cơ sở dữ liệu: DataBase |
| HQTCSDL = DBMS | Hệ quản trị Cơ sở dữ liệu |
| NSD = Users = Account | Người Sử Dụng = là quyền làm việc trên WebApp được đảm bảo bởi tối thiểu là 2 yếu tố: tên đăng nhập (Username) và mật khẩu (Password) |
| RBTV | Ràng buộc toàn vẹn dữ liệu khi thao tác |
|  |  |
| CAPTCHA | **C**ompletely **A**utomated **P**ublic **T**uring test to tell **C**omputers and **H**umans **A**part |
| Digital ID | Digital Identification: Chữ ký số |
| e-Doc | e-Document: Chứng từ điện tử |
| MS | Công ty Microsoft |
| OS | Operating System: Hệ điều hành |
| PC | Personal Computer: Máy tính cá nhân |
| PIN | Personal Identification Number: Mã số nhận diện cá nhân |
| SQL | Ngôn ngữ vấn tin có cấu trúc: Structured Query Language |
| SSIS | An toàn và bảo mật hệ thống thông tin: Safety and Security of Information Systems |
| URL | Uniform Resource Locator = địa chỉ thống nhất trên mạng máy tính, đặc biệt là Internet, cho phép định vị các tài nguyên dịch vụ ứng dụng, đặc biệt là trang điện tử đa phương tiện (Multimedia), ví dụ: địa chỉ web, … |
| VPN | Virtual Private Network: mạng riêng ảo. |
| VMWARE | Virtual Machine Software: Phần mềm máy ảo |
| WebApp | Web Application Software: phần mềm ứng dụng dạng Web |

# DANH MỤC CÁC THUẬT NGỮ

|  |  |
| --- | --- |
| Giao diện người dùng | Là hệ thống các màn hình giao tiếp cho phép người sử dụng tương tác với các thành phần hệ thống, điều khiển hệ thống hoạt động theo yêu cầu của người dùng - tương ứng các chức năng hiện có của phần mềm. |
| Mạng xã hội: Social Network = Social Media | [[1]](#footnote-1)Mạng xã hội là dịch vụ dựa trên nền tảng công nghệ (*Công nghệ Web hoặc công nghệ App trên SMART Phone*) cho phép nối kết các thành viên trên mạng Internet (*là các tài khoản người làm việc trên mạng thông qua e-mail, số điện thoại hoặc các dạng thông tin nhận diện cá nhân nào đó*) lại với nhau với nhiều mục đích khác nhau không phân biệt không gian và thời gian. Thông thường các thành viên tham gia vào một mạng xã hội nào đó sẽ hình thành một công đồng mạng xã hội có gần như cơ bản các hoạt động xã hội như công đồng dân cư thực tế, nhưng tất cả đều dựa trên tương tác “ảo” trên Internet. Các thành viên trên một mạng xã hội có thể gôm nhóm lại với nhau dựa trên cơ sở chung nhau một số yếu tố nào đó, như: cùng sở thích, cùng nhóm làm việc, cùng cộng đồng xã hội, … |
| Người dùng (User):  Tài khoản (Account) | Là một quyền làm việc trên hệ thống được cấp phát cho một cá nhân thông qua tên tài khoản (username) và mật khẩu (password). |
| Trực tuyến = Online | Là các hoạt động được thực hiện từ xa thông qua các dịch vụ trên Internet, đặc biệt là dịch vụ Web |
|  |  |
| Firewall | Bức tường lửa |
| Network | Mạng máy tính |
| Intranet | Mạng nội bộ |
| Internet | Mạng toàn cầu (International Network) |
| Extranet | Mạng ngoại bộ / VPN |
| Host Web | Là không gian lưu trữ và phát hành WebApp (Websites) trên mạng máy tính, đặc biệt là mạng Internet |

# MỤC LỤC

[LỜI CẢM ƠN 3](#_Toc88606118)

[DANH MỤC CÁC TỪ VIẾT TẮT 4](#_Toc88606119)

[DANH MỤC CÁC THUẬT NGỮ 5](#_Toc88606120)

[PHIẾU GIAO NHIỆM VỤ ĐỒ ÁN HỌC PHẦN 11](#_Toc88606122)

[Chương 1: GIỚI THIỆU ĐỒ ÁN CỦA HỌC PHẦN 12](#_Toc88606123)

[1.1. Tổng quan về ĐỒ ÁN HỌC PHẦN 12](#_Toc88606124)

[1.2. Nội dung chuyên môn chính của ĐỒ ÁN HỌC PHẦN 12](#_Toc88606125)

[1.3. Công cụ và nền tảng kỹ thuật thực hiện ĐỒ ÁN HỌC PHẦN 12](#_Toc88606126)

[1.4. Sản phẩm của ĐỒ ÁN HỌC PHẦN 12](#_Toc88606127)

[1.5. Bố cục của báo cáo 13](#_Toc88606128)

[Chương 2: CƠ SỞ LÝ THUYẾT CỦA HỌC PHẦN 14](#_Toc88606129)

[2.1. Giới thiệu về lập trình WebApp trên ASP.NET 14](#_Toc88606130)

[2.2. Phát triển WebApp ASP.NET đơn giản có CSDL dùng ADO.NET 14](#_Toc88606131)

[2.3. Master Page = Mẫu trang web 14](#_Toc88606132)

[2.3.1. Căn bản về MASTER PAGE 14](#_Toc88606133)

[2.3.1.1. Khái niệm Masterpage (mẫu trang web) 14](#_Toc88606134)

[**2.3.1.1.1.** **Masterpage ?** 14](#_Toc88606135)

[2.3.1.1.2. Số lượng Masterpage trong 1 Websites 15](#_Toc88606136)

[2.3.1.1.3. Các thành phần trên Masterpage 15](#_Toc88606137)

[2.3.1.2. Các bước thực hiện 15](#_Toc88606138)

[2.3.1.3. Xây dựng Mẫu trang (Masterpage) 16](#_Toc88606139)

[2.3.1.3.1. Thiết kế Masterpage 16](#_Toc88606140)

[2.3.1.3.2. VIẾT THỦ TỤC ẨN / HIỆN Login 16](#_Toc88606141)

[2.4. Phát triển các chức năng GIAO DỊCH trên WebApp ASP.NET 17](#_Toc88606142)

[2.4.1. Trưng bày sản phẩm (dịch vụ) trên DataList 17](#_Toc88606143)

[2.4.2. Session và giỏ hàng 17](#_Toc88606144)

[2.4.3. Đặt hàng và thanh toán 17](#_Toc88606145)

[2.5. Phát triển các chức năng QUẢN LÝ trên WebApp ASP.NET 18](#_Toc88606146)

[2.5.1. Quản lý đăng nhập và bảo mật 18](#_Toc88606147)

[2.5.2. Quản lý sản phẩm 18](#_Toc88606148)

[2.5.3. Các chức năng thao tác dữ liệu: Thêm \_ Sửa \_ Xóa 18](#_Toc88606149)

[2.6. Xử lý Ràng buộc toàn vẹn dữ liệu (RBTV) trên CSDL trong ASP.NET 18](#_Toc88606150)

[2.6.1. Giới thiệu về lập trình xử lý RBTV dữ liệu trên CSDL trong ASP.NET 18](#_Toc88606151)

[2.6.1.1. Giới thiệu về RBTV 18](#_Toc88606152)

[2.6.1.2. Cách thực hiện: 18](#_Toc88606153)

[2.6.1.3. Các bước thực hiện 18](#_Toc88606154)

[2.6.2. Thiết kế hệ thống xử lý RBTV dữ liệu 18](#_Toc88606155)

[2.6.3. Lập trình hệ thống xử lý RBTV dữ liệu 18](#_Toc88606156)

[2.7. Lập trình thủ tục Chuyển nhóm dữ liệu trong ASP.NET 18](#_Toc88606157)

[2.7.1. Giới thiệu về lập trình Chuyển nhóm dữ liệu trên CSDL trong ASP.NET 18](#_Toc88606158)

[2.7.2. Thiết kế hệ thống xử lý Chuyển nhóm dữ liệu 18](#_Toc88606159)

[2.7.3. Lập trình hệ thống xử lý Chuyển nhóm dữ liệu 19](#_Toc88606160)

[2.8. Lập trình thủ tục giao dịch Đặt hàng (Order) 19](#_Toc88606161)

[2.8.1. Giới thiệu về lập trình xử lý giao dịch đặt hàng trong ASP.NET 19](#_Toc88606162)

[2.8.2. Chức năng khách hàng 19](#_Toc88606163)

[2.8.3. Chức năng đơn đặt hàng 19](#_Toc88606164)

[2.8.4. Chức năng chi tiết đơn hàng 19](#_Toc88606165)

[2.8.5. Xử lý lưu trữ vào DB 19](#_Toc88606166)

[2.9. Triển khai phát hành WebApp ASP.NET trên Internet (hoặc Intranet) 19](#_Toc88606167)

[2.10. Giới thiệu về lập trình WebApp trên PHP 20](#_Toc88606168)

[Chương 3: THIẾT KẾ VÀ CẬP NHẬT CSDL PHỤC VỤ WEBAPP 22](#_Toc88606169)

[3.1. Phân tích và thiết kế DB 22](#_Toc88606170)

[3.2. Diagram của DB bán vé sự kiện âm nhạc online 23](#_Toc88606171)

[3.3. Thiết kế các Tables dữ liệu 23](#_Toc88606172)

[3.3.1. Thiết kế các Table DanhMucVe 23](#_Toc88606173)

[3.3.2. Thiết kế các Table DonMuaVe 24](#_Toc88606174)

[3.3.3 Thiết kế các Table ChiTietDonMuaVe 24](#_Toc88606175)

[3.3.4 Thiết kế các Table DanhMucLoaiVe 24](#_Toc88606176)

[3.3.5 Thiết kế các Table DanhMucNgheSi 24](#_Toc88606177)

[3.3.6 Thiết kế các Table DanhMucTheLoai 24](#_Toc88606178)

[3.3.7 Thiết kế các Table DiemBanVe 25](#_Toc88606179)

[3.3.8 Thiết kế các Table KhachHang 25](#_Toc88606180)

[3.3.9 Thiết kế các Table KhuVuc 25](#_Toc88606181)

[3.3.10 Thiết kế các Table NhanVien 25](#_Toc88606182)

[3.3.11 Thiết kế các Table NhanVien 25](#_Toc88606183)

[3.4. Cập nhật dữ liệu DB thực nghiệm cho WebApp bán vé online 25](#_Toc88606184)

[3.5. Giới thiệu một số SP và Trigger trong DB quản lý bán vé online 28](#_Toc88606185)

[Chương 4: THIẾT KẾ VÀ LẬP TRÌNH WEBAPP PHẦN GIAO DỊCH 29](#_Toc88606186)

[4.1. Giới thiệu về WebApp bán vé online 29](#_Toc88606187)

[4.2. MasterPage giao dịch : MasterPage.Master 29](#_Toc88606188)

[4.2.1. Thiết kế Masterpage giao dịch 29](#_Toc88606189)

[4.2.2. Lập trình (Codes) MasterPage giao dịch 30](#_Toc88606190)

[4.3. Trang chủ trưng bày các loại vé: HomePage.aspx 31](#_Toc88606191)

[4.3.1. Thiết kế (Design) WebApp giao dịch vé online 31](#_Toc88606192)

[4.3.2. Lập trình (Codes) WebApp giao dịch vé online: Homepage.aspx 31](#_Toc88606193)

[4.4. Trang “Giỏ hàng” của khách hàng: Cart.aspx 33](#_Toc88606194)

[4.4.1. Thiết kế (Design) WebPage “Giỏ hàng” 33](#_Toc88606195)

[4.4.2. Mã lệnh (Codes) WebPage “Giỏ hàng” 33](#_Toc88606196)

[4.5. Đơn đặt vé: FrOrder.aspx 34](#_Toc88606197)

[4.5.1. Giới thiệu 34](#_Toc88606198)

[4.5.2. Thiết kế trang Đặt hàng: FrOrder.aspx 35](#_Toc88606199)

[4.5.3. Lập trình mã lệnh (Codes) của trang Đặt hàng: Managements.aspx 37](#_Toc88606200)

[Chương 5: THIẾT KẾ VÀ LẬP TRÌNH WEBAPP PHẦN QUẢN LÝ 43](#_Toc88606201)

[5.1. Giới thiệu về WebApp quản lý hệ thống bán vé Online 43](#_Toc88606202)

[5.2. Masterpage quản lý hệ thống bán vé online 44](#_Toc88606203)

[5.2.1. Thiết kế Masterpage quản lý 44](#_Toc88606204)

[5.2.2. Lập trình bổ sung Masterpage quản lý 45](#_Toc88606205)

[5.3. Trang chính của hệ thống quản lý hệ thống bán vé online 47](#_Toc88606206)

[5.3.1. Thiết kế (Design) trang quản lý chung (chính): Managements.aspx 47](#_Toc88606207)

[5.3.2. Lập trình (Codes) trang quản lý chung (chính): Mannagements.aspx 47](#_Toc88606208)

[5.4. Trang quản lý danh mục quản lý vé: FrTicketManagements.aspx 48](#_Toc88606209)

[5.4.1. Phân tích trang Quản lý vé 48](#_Toc88606210)

[5.4.2. Thiết kế (Design) trang quản lý danh mục vé: FrTicketsManagements.aspx 49](#_Toc88606211)

[5.4.3. Lập trình (Codes) trang quản lý danh mục vé: FrTicketsManagements.aspx 55](#_Toc88606212)

[5.4.4. Lập trình (Codes) THÊM \_ SỬA \_ XÓA trang quản lý danh mục vé: FrTicketsManagements.aspx 55](#_Toc88606213)

[5.4.4.1. Thêm vé 57](#_Toc88606214)

[5.4.4.2. Sửa thông tin vé 60](#_Toc88606215)

[5.4.4.3. Xóa vé 63](#_Toc88606216)

[5.4.5. Lập trình (Codes) xử lý RBTV dữ liệu khi Xóa danh mục vé: FrTicketsManagements.aspx 63](#_Toc88606217)

[5.4.5.1. Giới thiệu 63](#_Toc88606218)

[5.4.5.2. Thiết kế 2 GridView phụ ngay trên Form quản lý FrGoodsManagements.aspx 65](#_Toc88606219)

[5.4.5.3. Xử lý các RBTV trên 2 GridView phụ 66](#_Toc88606220)

[5.5. Trang quản lý nhân sự: FrHumanResource.aspx 67](#_Toc88606221)

[5.6. Trang quản lý điểm bán vé (FrBookingOffice.aspx) 69](#_Toc88606222)

[5.7. Trang quản lý thể loại âm nhạc (FrMusicGenre.aspx) 72](#_Toc88606223)

[5.8. Trang quản lý thể loại nghệ sĩ biểu diễn (FrArtist.aspx) 74](#_Toc88606224)

[5.9. Trang quản lý khu vực (FrArea.aspx) 75](#_Toc88606225)

[5.10. Trang quản lý loại vé (FrTicketClass.aspx) 77](#_Toc88606226)

[5.11. Trang quản lý vé tồn (FrStock.aspx) 80](#_Toc88606227)

[5.12. Chuyển Nhóm: Chuyển một mặt hàng từ nhóm này sang nhóm khác 82](#_Toc88606228)

[Chương 6: TRIỂN KHAI PHÁT HÀNH WEBAPP 83](#_Toc88606229)

[6.1. Giới thiệu về Host miễn phí somee 83](#_Toc88606230)

[6.2. Đăng ký tài khoản somee 83](#_Toc88606231)

[6.3. Restore BD lên host somee 84](#_Toc88606232)

[6.4. Upload đăng tải Source WebApp lên host somee để phát hành 85](#_Toc88606233)

[6.5. Giới thiệu sản phẩm WebApp quản lý bán vé online trên somee 86](#_Toc88606234)

[Chương 7: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN CỦA ĐỒ ÁN 87](#_Toc88606235)

[**7.1.** **Kết luận** 87](#_Toc88606236)

[7.1.1. Những kết quả đạt được 87](#_Toc88606237)

[7.1.2. Hạn chế 87](#_Toc88606238)

[7.2. Hướng phát triển 87](#_Toc88606239)

[7.2.1. Hướng khắc phục các hạn chế 87](#_Toc88606240)

[7.2.2. Hướng mở rộng ĐỒ ÁN HỌC PHẦN 87](#_Toc88606241)

[DANH MỤC TÀI LIỆU THAM KHẢO 88](#_Toc88606242)

[CÁC PHỤ LỤC 89](#_Toc88606243)

**TRƯỜNG ĐẠI HỌC TÀI CHÍNH - MARKETING**

**KHOA CÔNG NGHỆ THÔNG TIN**

# PHIẾU GIAO NHIỆM VỤ ĐỒ ÁN HỌC PHẦN

|  |  |
| --- | --- |
| **Giảng viên: ThS. GVC. Võ Xuân Thể** | |
| Tên học phần: **Lập trình web (4 tín chỉ); Mã lớp HP: 2111112005703** | |
| Tên đề tài: **Lập trình phát triển WebApp bán vé sự kiện âm nhạc trực tuyến (online)** | |
| **Sinh viên thực hiện: Đinh Văn Thành** | |
| **Thời gian thực hiện: 30/08/2021**  đến **3/12/2021** | |
| **Yêu cầu của ĐỒ ÁN HỌC PHẦN & Cơ cấu điểm**  **Lý thuyết:** Cơ sở lý thuyết HP Lập trình Web (4 tín chỉ) bao gồm lập trình Web giao dịch và lập trình Web quản lý trên nền tảng Công nghệ ASP.NET  **Thực hành:**  Lập trình phát triển WebApp hệ thống bán vé sự kiện âm nhạc trực tuyến (online), bao gồm 2 phần: vé sự kiện âm nhạc  + Giao dịch mua bán  . MasterPage.master: Mẫu trang giao dịch kiêm quản lý bán vé sự kiện âm nhạc  . HomePage.aspx: Trang chủ, trưng bày các loại vé sự kiện âm nhạc;  . Cart.aspx: Giỏ hàng chọn mua vé sự kiện âm nhạc;  . Order.aspx: Đặt mua vé;  + Quản lý MasterPage.master: Sử dụng Master Giao dịch (nêu trên)  . Quản lý chung: có xử lý đăng nhập Login  . Quản lý danh mục các loại vé sự kiện âm nhạc: Thêm, Xóa, Sửa, RBTV;  . Chuyển nhóm loại vé sự kiện âm nhạc  + Phát hành WebApp lên Host miễm phí: somee.com | |
| **GIẢNG VIÊN**  **Võ Xuân Thể** | **Ngày 2 tháng 12 năm 2021**  **SV Thực hiện**  **Đinh Văn Thành** |

# GIỚI THIỆU ĐỒ ÁN CỦA HỌC PHẦN

## Tổng quan về ĐỒ ÁN HỌC PHẦN

ĐỒ ÁN HỌC PHẦN thiết kế và lập trình phát triển WebApp bán vé sự kiện âm nhạc trực tuyến trên nền tảng công nghệ ASP.NET : phiên bản 4.7.2 với công cụ MS. Visual Studio.NET 2019, dùng ngôn ngữ lập trình C#.NET: **G322DinhVanThanhVeOnline**

Hệ thống WebApp nêu trên được triển khai thực nghiệm trên Host Web của www.somee.com với URL là:

http://ticketg322dvthanh.somee.com OR http://www.ticketg322dvthanh.somee.com

Hệ thống CSDL dùng cho WebApp nên trên **1G322DinhVanThanhVeOnline.bak (MS. SQL 2018)** được thiết kế và cập nhật trên MS. SQL Server 2018 (trên somee là : MS SQL 2018 Express)

## Nội dung chuyên môn chính của ĐỒ ÁN HỌC PHẦN

Vận dụng kiến thức về thiết kế và lập trình WebApp 1 lớp trên nền tảng công nghệ ASP.NET vào việc phát triển một WebApp đơn giản sử dụng trong thực tiễn là: Hệ thống Websites bán vé sự kiện âm nhạc trực tuyến.

## Công cụ và nền tảng kỹ thuật thực hiện ĐỒ ÁN HỌC PHẦN

+ ASP.NET 4.7.2 trên MS. Visual Studio .NET 2019

+ Hệ quản trị CSDL MS. SQL Server 2018 trên Local,

trên online somee.com: MS. SQL Server 2018 Express

+ Free Host Web: some.com

## Sản phẩm của ĐỒ ÁN HỌC PHẦN

+ Hệ thống WebApp bán sự kiện âm nhạc trực tuyến (online) dạng Local trên PC: **G322DinhVanThanhVeOnline**

+ Hệ thống WebApp bán sự kiện âm nhạc trực tuyến (online) dạng Online trên somee:

http://ticketg322dvthanh.somee.com OR <http://www.ticketg322dvthanh.somee.com>

## Bố cục của báo cáo

Báo cáo gồm những nội dung như sau:

Chương 1 Giới thiệu đồ án của HP là Hệ thống WebApp bán sự kiện âm nhạc online

Chương 2 Các cơ sở lý thuyết của HP Lập trình Web (4 TC) phục việc thực hiện đề tài HP

Chương 3 Thiết kế và cập nhật CSDL phục vụ WebApp bán sự kiện âm nhạc online

Chương 4 Thiết kế và lập trình WebApp phần giao dịch trực tuyến của khách hàng mua sự kiện âm nhạc

Chương 5 Thiết kế và lập trình WebApp phần quản lý trực tuyến của cơ sở bán sự kiện âm nhạc

Chương 6 Triển khai và phát hành WebApp bán sự kiện âm nhạc trực tuyến trên Host: somee

Chương 7 Tổng kết các kết quả đạt được và còn hạn chế của đồ án, đồng thời đề xuất hướng khắc phục hạn chế và phát triển Đồ án.

# CƠ SỞ LÝ THUYẾT CỦA HỌC PHẦN

## Giới thiệu về lập trình WebApp trên ASP.NET

ASP.NET là một nền tảng ứng dụng web (web application framework) được phát triển và cung cấp bởi Microsoft, cho phép những người lập trình tạo ra những trang web động, những ứng dụng web và những dịch vụ web. Lần đầu tiên được đưa ra thị trường vào tháng 1 năm 2002 cùng với phiên bản 1.0 của.NET framework, là công nghệ nối tiếp của Microsoft's Active Server Pages(ASP). ASP.NET được biên dịch dưới dạng Common Language Runtime (CLR), cho phép những người lập trình viết mã ASP.NET với bất kỳ ngôn ngữ nào được hỗ trợ bởi.NET language.

## Phát triển WebApp ASP.NET đơn giản có CSDL dùng ADO.NET

ADO.NET là tập hợp các thư viện được cung cấp bởi Microsoft nhằm giúp cho ứng dụng có thể thao tác (select, insert, update, delete) tới cơ sở dữ liệu (CSDL) bất kỳ như Sql Server, Oracle, MySql, SQLite...

ADO.NET có thể được sử dụng để phát triển bất kỳ loại ứng dụng .NET nào. Sau đây là một số ứng dụng .NET mà bạn có thể sử dụng công nghệ truy cập dữ liệu ADO.NET để tương tác với nguồn dữ liệu.

* ASP.NET Web Form Applications
* Windows Applications
* ASP.NET MVC Applications
* Console Applications
* ASP.NET Web API Applications

## Master Page = Mẫu trang web

## Căn bản về MASTER PAGE

## Khái niệm Masterpage (mẫu trang web)

## **Masterpage ?**

Trong một hệ thống website thường sẽ có nhiều trang con khác nhau, mỗi trang sẽ phụ trách một mảng nội dung nào đó, có thể kết cấu trang cũng có những điểm khác biệt nhất định nhưng tất cả sẽ được thiết kế theo một khuôn mẫu nào đó, và áp dụng chung cho tất cả các trang. Khuôn mẫu này thường chúng ta sẽ gọi là Theme hay Template, còn trong ASP.Net có khái niệm MasterPage.

## Số lượng Masterpage trong 1 Websites

MasterPage là một trang dùng để tạo ra cấu trúc giao diện chung cho các trang web trong hệ thống, những trang thừa kế từ MasterPage sẽ áp dụng lại giao diện mẫu của nó. Trong một website của thể có 1 hoặc nhiều MasterPage.

## Các thành phần trên Masterpage

Master Pages gồm 2 phần chính:

* Master page

Dùng để xây dựng một bố cục chung cho Web site

Tập tin lưu trữ có phần mở rộng là .master

Thay thế tag định hướng <%@ Page …%> bằng tag <%@ Master …%>

Thao tác thiết kế: giống trang .aspx, nhưng phải bổ sung ít nhất một điều khiển

* ContentPalceHolder

Content page: chính là trang .aspx nhưng có các đặc điểm sau

Bổ sung thuộc tính MasterPageFile=“Tên tập tin master” trong định hướng <%@ Page … %>

Không chứa các tag tạo ra cấu trúc trang HTML

Chứa tag <asp:Content …> </asp:Content>: số lượng tag này tùy thuộc vào trang master được kết hợp

Chỉ được phép tạo nội dung trong tag <asp:Content …>

## Các bước thực hiện

Bước 1: tạo trang Master => chọn icon Master Page

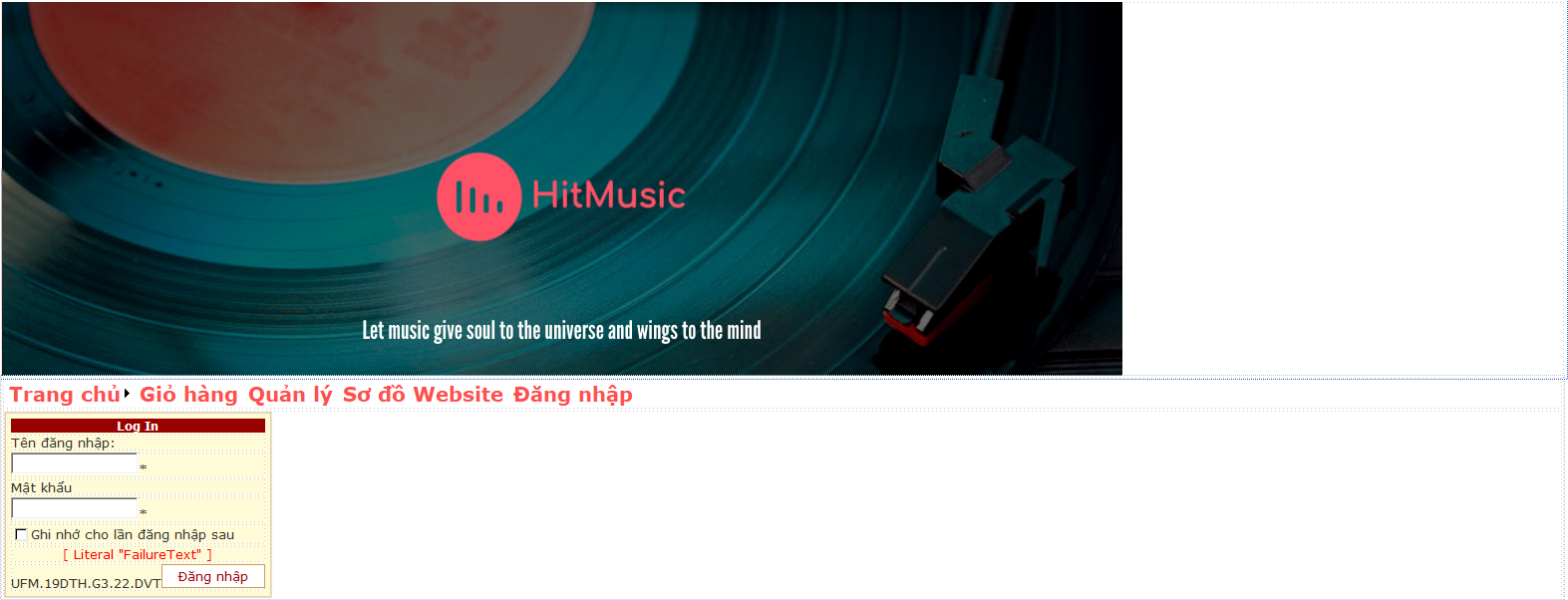
Bước 2: thiết kế bố cục chung và bổ sung điều khiển ContentPlaceHolder vào trang Master

Bước 3: tạo trang Content và khai báo trang Master

Bước 4: thiết kế nội dung trên trang Content

## Xây dựng Mẫu trang (Masterpage)

## Thiết kế Masterpage



## VIẾT THỦ TỤC ẨN / HIỆN Login

public partial class MasterPage : System.Web.UI.MasterPage

{

/// <summary>

/// Thủ tục tự chạy khi mở trên web browser

/// ẩn login nv

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

protected void Page\_Load(object sender, EventArgs e)

{

LoginNv.Visible = false;

}

/// <summary>

/// Viết thủ tục ẩn hiện login (trên MasterPage.master)

/// </summary>

/// <param name="sender"></param>

/// <param name="e">lưu trữ thông tin mục chọn trên menumain mà nhân viên đãchọn</param>

protected void MenuMain\_MenuItemClick(object sender, MenuEventArgs e)

{

if(e.Item.Value == "ql" || e.Item.Value == "dn") // KHINSD(NV) chọn quản lý hoặc đăng nhập

{

LoginNv.Visible = true; // mở công cụ loginnv để nv đăng nhập để thực hiện các chức năng quản lý hệ thống

}

else // khi nv chọn các mục khách

{

LoginNv.Visible = false;

}

}

## Phát triển các chức năng GIAO DỊCH trên WebApp ASP.NET

### Trưng bày sản phẩm (dịch vụ) trên DataList

Trong thiết kế trang chủ trưng bày sản phẩm như:

+ Thiết kế Table có 1 cột và nhiều dòng tạo thuận lợi cho việc kết cấu vị trí các thành phần thiết kế trên HomepagePage.

+ Dùng DataList để trưng bày các loại vé cho phép khách hàng chọn mua trực tuyến

+…

### Session và giỏ hàng

Sử dụng 3 đối tượng chính là: DataTable và Session và GridView để tạo và hiển thị giỏ hàng.

Trong đó:

+ DataTable đóng vai trò là giỏ hàng.

+ Session được sử dụng để lưu trữ giỏ hàng trong một phiên làm việc.

+ GridView sử dụng để hiển thị thông tin và chỉnh sửa số lượng sản phẩm trong giỏ hàng.

Với các thao tác:

* Tạo và lưu trữ giỏ hàng
* Thêm sản phẩm vào giỏ hàng
* Sửa thông tin giỏ hàng
* Xóa sản phẩm khỏi giỏ hàng
* Duyệt qua thông tin từng sản phẩm trong giỏ hàng

### Đặt hàng và thanh toán

+ Trang đặt hàng được gọi từ trang giỏ hàng không rỗng, tức là đã có hàng trong “Giỏ hàng”.

+ Trang đặt có 4 chức năng chính:

. Chức năng khách hàng: tìm khách hàng cũ, thêm khách hàng mới.

. Chức năng đơn đặt hàng: mã đơn, ngày lập, thông tin giao nhận,…

. Chức năng chi tiết đơn hàng: lấy từ giỏ hàng sang.

+ Xử lý lưu trữ : sqlDataSource / ToolBox.

## **Phát triển các chức năng QUẢN LÝ trên WebApp ASP.NET**

### Quản lý đăng nhập và bảo mật

Sử dụng công cụ Login trong Toolbox để quản lý phần đăng nhập.

Tài khoản bao gồm 2 loại:

+ Tài khoản của nhân viên bao gồm cả quản trị: Tài khoản này là do Admin cấp và người dùng không tự đăng ký được, nếu như quên mật khẩu thì nhân viên phải thông báo và liên hệ với Admin để cấp lại tài khoản.

+ Tài khoản của khách hàng: Khách hàng tự đăng ký mới, nếu như quên mật khẩu có thể tự xin cấp lại mật khẩu mới mà không cần nhờ đến Admin. Vì vậy, khi đăng ký tài khoản mới cần phải cung cấp rất nhiều thông tin và khi muốn lấy lại mật khẩu thì phải trả lời đúng các thông tin đã đăng kí, câu hỏi bí mật,… sau đó mật khẩu sẽ được gởi lại qua các địa chỉ liên hệ nhưng thường là Email. Ngoài ra khi lập tài khoản mới còn có Captcha, hiện nay lập trình WebApp có 2 loại thông dụng như: Text ngẫu nhiên, hình ảnh.

+ Ngoài ra còn một số công cụ .NET khác hỗ trợ cho các tài khoản: ChangePassWord, LoginName, LoginStatus, LoginView,…

### Quản lý sản phẩm

Thực hiện quản lý sản phẩm bằng cách sử dụng các câu lệnh SQL.

### Các chức năng thao tác dữ liệu: Thêm \_ Sửa \_ Xóa

Sử dụng thư viện ADO.NET thực hiện các câu truy vấn trên mỗi GridView được tạo.

## **Xử lý Ràng buộc toàn vẹn dữ liệu (RBTV) trên CSDL trong ASP.NET**

### Giới thiệu về lập trình xử lý RBTV dữ liệu trên CSDL trong ASP.NET

### Giới thiệu về RBTV

Ràng buộc toàn vẹn (Integrity Constraint – RBTV) là những quy định trong CSDL xuất phát từ thực tiễn hay từ các mô hình dữ liệu, RBTV được đưa ra nhằm loại bỏ các bất thường trong CSDL.

### Cách thực hiện:

Khi thực hiện xóa dữ liệu phải xóa khóa các ngoại trước khi xóa khóa chính.

### Các bước thực hiện

Tạo các GridView có khóa ngoại liên quan đến dữ liệu trong GridView cần xử lí

Thực hiện câu lệnh SQL xóa trên các GridView phụ rồi mới thực hiện câu lệnh SQL xóa trên GridView có khóa chính

### Thiết kế hệ thống xử lý RBTV dữ liệu

Thiết kế các GridView phụ có liên quan đến khóa chính cần xử lí

### Lập trình hệ thống xử lý RBTV dữ liệu

Sử dụng các câu lệnh SQL được tích hợp để xử lý ràng buộc khi xóa

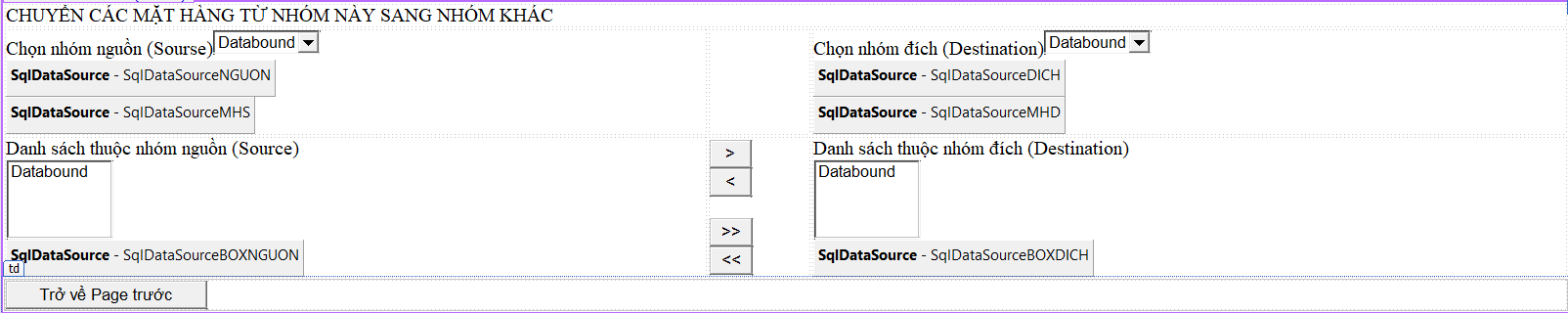
## **Lập trình thủ tục Chuyển nhóm dữ liệu trong ASP.NET**

### Giới thiệu về lập trình Chuyển nhóm dữ liệu trên CSDL trong ASP.NET

Chuyển mặt hàng này từ một nhóm hàng sang một nhóm hàng khác

### Thiết kế hệ thống xử lý Chuyển nhóm dữ liệu

Thiết kế WebForm chuyển nhóm dữ liệu như sau:



### Lập trình hệ thống xử lý Chuyển nhóm dữ liệu

public partial class FrChuyenNhomVe : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

DropDownListS\_SelectedIndexChanged(sender, e);

}

protected void btnBack\_Click(object sender, EventArgs e)

{

Response.Redirect("~\\Managements.aspx");

}

protected void LR\_Click(object sender, EventArgs e)

{

SqlDataSourceMHS.Update(); // gọi Update() bên Nguồn (S) đề chuyển

ListBoxS.DataBind();//Tải DL sau Update lên Source

ListBoxD.DataBind();//Tải DL sau Update lên Destinatio

}

protected void RL\_Click(object sender, EventArgs e)

{

// CHUYỂN 1 MH TỪ NHÓM "ĐÍCH" SANG NHÓM "NGUỒN" : RIGHT to LEFT

SqlDataSourceMHD.Update(); // gọi Update() bên Đích (D) đề chuyển

ListBoxS.DataBind();//Tải DL sau Update lên Source

ListBoxD.DataBind();//Tải DL sau Update lên Destination

}

protected void ButtonLAR\_Click(object sender, EventArgs e)

{

SqlDataSourceMHS.UpdateCommand = "UPDATE DanhMucVe SET masoloaive = @masoloaive WHERE masoloaive = @masoloaive";

//Viết lại lệnh Update....không Where..

SqlDataSourceMHS.UpdateParameters.Clear(); //Xóa các tham số cũ

SqlDataSourceMHS.UpdateParameters.Add("masoloaive", DropDownListD.SelectedValue.Trim());

//Khai báo lại tham số cho Update...

SqlDataSourceMHS.UpdateParameters.Add("masoloaive", DropDownListS.SelectedValue.Trim());

SqlDataSourceMHS.Update(); // gọi Update() bên Nguồn (S) đề chuyển

ListBoxS.DataBind();//Tải DL sau Update lên Source

ListBoxD.DataBind();//Tải DL sau Update lên Destination

}

protected void ButtonRAL\_Click(object sender, EventArgs e)

{

SqlDataSourceMHD.UpdateCommand = "UPDATE DanhMucVe SET masoloaive = @masoloaive WHERE masoloaive = @masoloaive";

//Viết lại lệnh Update....không Where..

SqlDataSourceMHD.UpdateParameters.Clear(); //Xóa các tham số cũ

SqlDataSourceMHD.UpdateParameters.Add("masoloaive", DropDownListD.SelectedValue.Trim());

//Khai báo lại tham số cho Update...

SqlDataSourceMHD.UpdateParameters.Add("masoloaive", DropDownListS.SelectedValue.Trim());

SqlDataSourceMHD.Update(); // gọi Update() bên Nguồn (S) đề chuyển

ListBoxD.DataBind();//Tải DL sau Update lên Source

ListBoxS.DataBind();//Tải DL sau Update lên Destination

}

## **Lập trình thủ tục giao dịch Đặt hàng (Order)**

### Giới thiệu về lập trình xử lý giao dịch đặt hàng trong ASP.NET

Trang đặt hàng được gọi từ trang giỏ hàng không rỗng, tức là đã có hàng trong “Giỏ hàng”.

Điền các thông tin về khách đặt hàng nếu là khách hàng cũ thì chỉ việc chọn thông tin khách hàng đó để hoàn thành đơn và chi tiết đơn, nếu là khách hàng mới phải tạo thông tin khách hàng mới rồi mới thực hiện được các bước sau.

### Chức năng khách hàng

Tìm khách hàng cũ và thêm khách hàng mới

### Chức năng đơn đặt hàng

Tạo đơn hàng với các thông tin như: mã đơn, ngày lập, thông tin giao nhận,…

### Chức năng chi tiết đơn hàng

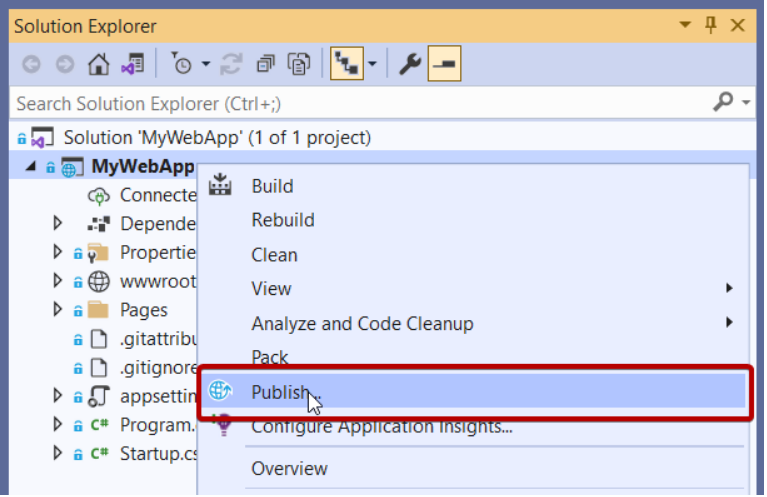
Lấy thông tin từ giỏ hàng sang sau khi đơn đặt đã được duyệt

### Xử lý lưu trữ vào DB

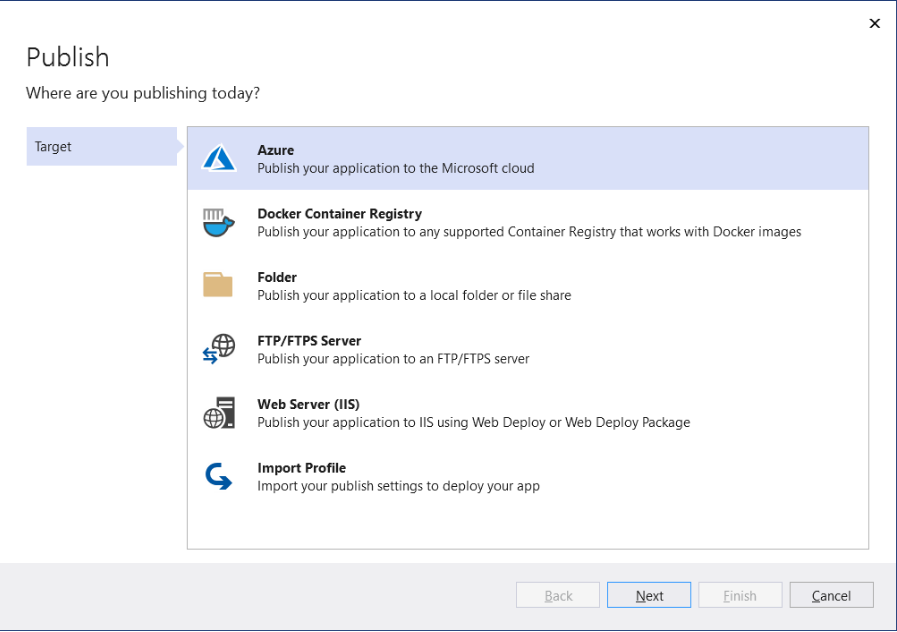
Xử lý lưu trữ : sqlDataSource / ToolBox.

## **Triển khai phát hành WebApp** ASP.NET trên Internet (hoặc Intranet)

Trong Solution Explorer, nhấp chuột phải vào dự án của bạn và chọn Publish.



Nếu đây là lần đầu tiên xuất bản ứng dụng web này, tiếp theo, sẽ thấy trình hướng dẫn Publish.



Thực hiện các bước sau theo chỉ dẫn.

## Giới thiệu về lập trình WebApp trên PHP

PHP là cụm từ viết tắt của PHP: HyperText Preprocessor, có nghĩa là bộ tiền xử lý siêu văn bản PHP.

PHP là một ngôn ngữ lập trình được tạo ra bởi Rasmus Lerdorf vào năm 1994, chủ yếu được dùng để phát triển các ứng dụng web chạy trên máy chủ. Tên "cúng cơm" của PHP là "Personal Home Page". Cùng với sự hậu thuẫn của thế giới mã nguồn mở, liên minh LAMP (Linux, Apache, MySQL và PHP) đã nhanh chóng đánh bại nền tảng công nghệ web "thời thượng" hồi đó của Microsoft, trở thành một nền tảng hàng đầu trong phát triển các ứng dụng web. Những website thuộc dạng top trên thế giới như Google, Facebook, Yahoo, YouTube, Wikipedia, Flickr, ....đều được viết bằng PHP hoặc sử dụng PHP để phát triển các module bên trong chúng.

Mặc dù có tuổi thọ tương đối cao, nhưng PHP vẫn là một trong top 5 các ngôn ngữ lập trình phổ biến và nên học nhất trên thế giới. Trong lĩnh vực lập trình web, PHP chỉ đứng sau JavaScript về mức độ phổ biến. Trong lĩnh vực lập trình back-end, PHP vẫn chiếm vị trí dẫn đầu.

Các chức năng của PHP:

* PHP có thể tạo ra các trang web động (sản sinh ra các nội dung HTML/CSS/JavaScript), và đó chính là lý do mà chúng ta nên tìm hiểu về HTML/CSS trước khi đến với PHP.
* PHP có thể mở, đọc, ghi, xóa file trên máy chủ, tương tác với các phần mềm khác trên máy chủ
* PHP có thể kết hợp với các hệ quản trị CSDL, lưu trữ, truy vấn, xử lý dữ liệu và trả kết quả về cho người dùng
* PHP có thể gửi/nhận cookie, mã hóa, giải mã dữ liệu
* PHP có thể phát triển các dịch vụ web (web services)
* ...

# THIẾT KẾ VÀ CẬP NHẬT CSDL PHỤC VỤ WEBAPP

## Phân tích và thiết kế DB

Hệ thống DB phục vụ hoạt động webapp bán vé sự kiện trực tuyến như sau:

+ Thành phần chính là Table DanhMucVe gồm các thuộc tính về một loại vé: masove, tenve, masotheloai, masonghesi, hinhanh, thoigian, diadiem, khucuc, gia, masoloaive

+ Mỗi vé thuộc một danh mục loại vé: vé vip, vé thường và có thông tin về danh mục nghệ sĩ và danh mục thể loại

+ Vé được bán tại các điểm bán vé

+ Khách hàng tực tuyến thực hiện các thủ tục đặt hàng theo các trình tự:

. Cart / bộ nhớ = NSD đang chọn hàng (chưa mua): MUA lập

|| KH quyết định đặt hàng giữ chỗ

. Order || Book = Đơn đặt hàng || Phiếu đặt[giữ] chỗ [ko PL] = chưa chắc mua hàng = contract [PL]: MUA lập

||

. Phiếu tính tiền (Bill): BÁN lập

|| Mua trả tiền

. Phiếu thu (Receipt) = tạm thu (chưa chính thức thu), vì còn cần kiểm tra lại OR thu chưa đủ: BÁN lập

||

. Hóa đơn (Invoice) = người bán đã chính thức thu tiền, người mua đã chính thức mua hàng: hoàn tất thương vụ = giữa các bên cơ bản hết trách = kn: là chứng từ xác thực bên mua đã hoàn thành nghĩa vụ tài chính (diễn đạt theo Hợp đồng = contract = thỏa thuận = “lời hứa”) = “trả tiền” đối với bên bản = chứng từ này có giá trị trong hạch toán tài chính (đặc biệt với cơ quan Thuế nhà nước).

Do bên BÁN lập

+ Lập Tables từ DanhMucVe ra -> các Tables khác

+ Nhập dữ liệu vào các Tables từ ngoài -> vào

+ Không lưu hình ảnh vào DB (KHÔNG NÊN: image) ==trong DB chỉ lưu path và tên file hình   
 = nchar || nvarchar (tránh khả năng DB theo thời gian quá lớn => không sử dụng được)

NHẬP DỮ LIỆU VÀO CÁC BẢNG = ĐỦ ĐỂ CHẠY THỰC NGHIỆM WEBAPP

+ Nhập dữ liệu từ các Tables bên ngoài vào & chỉ nhập DL bên Bán (không nhập bên “Mua” = chi khi có KH mua hàng)

+ Thứ tự:

[1] KhuVuc, DanhMucLoaiVe, DanhMucNgheSi, DanhMucTheLoai

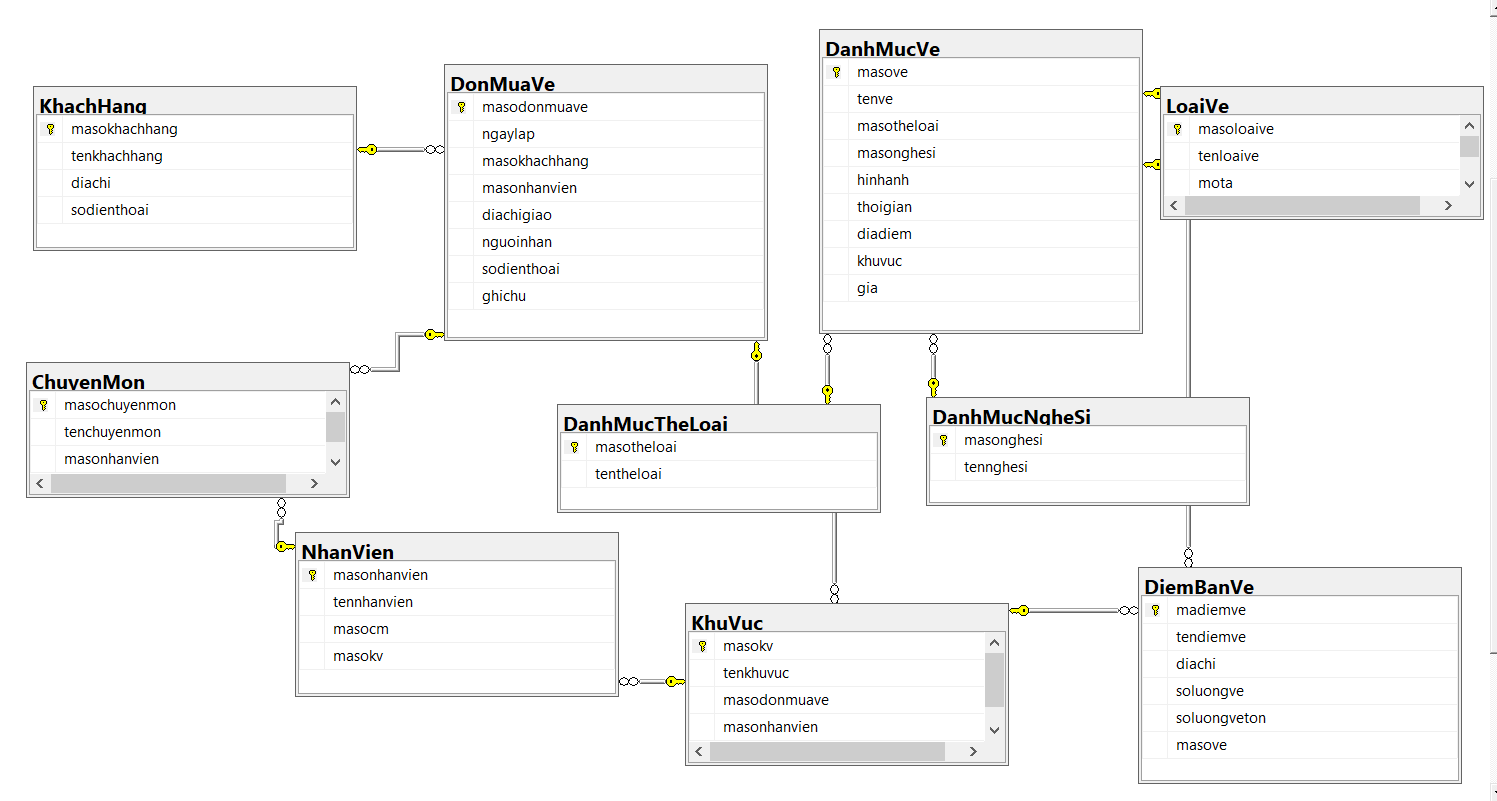
[2] DiemBanVe, NhanVien

[3] DanhMucVe

[4] TonKho

## Diagram của DB bán vé sự kiện âm nhạc online

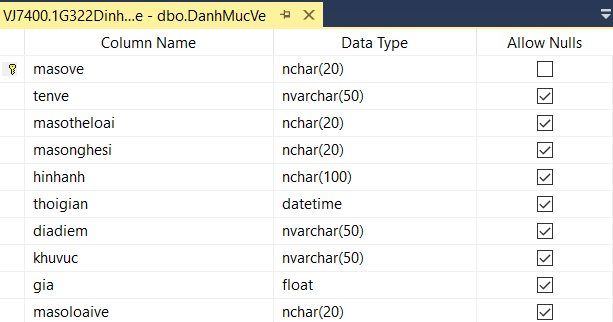
Căn cứ vào phân tích, thiết kế thành phần CSDL từ mục 3.1 nêu trên, ta có Diagram của CSDL phục vụ cho hoạt động của WebApp bán vé sự kiện âm nhạc như Hình 3. 1:



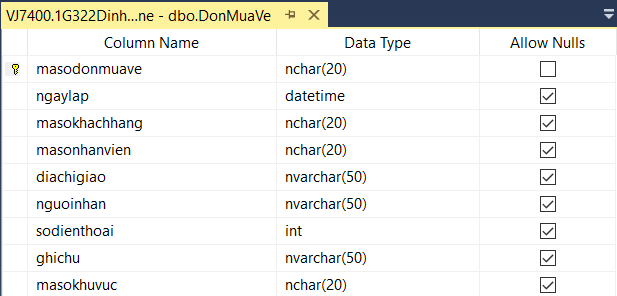
Hình 3. 1: Diagram

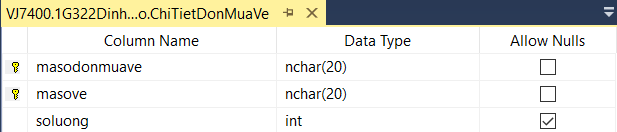
## Thiết kế các Tables dữ liệu

## Thiết kế các Table DanhMucVe

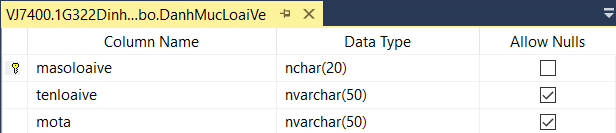


## Thiết kế các Table DonMuaVe

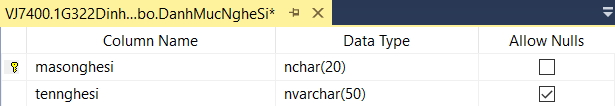


3.3.3 Thiết kế các Table ChiTietDonMuaVe 

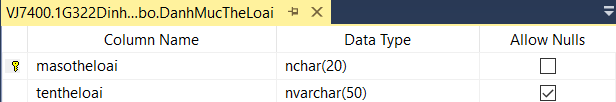
# 3.3.4 Thiết kế các Table DanhMucLoaiVe



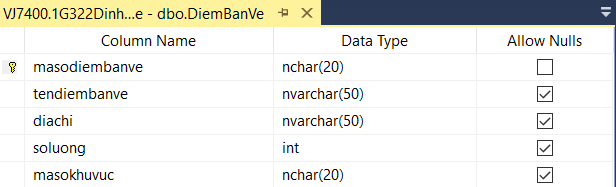
# 3.3.5 Thiết kế các Table DanhMucNgheSi



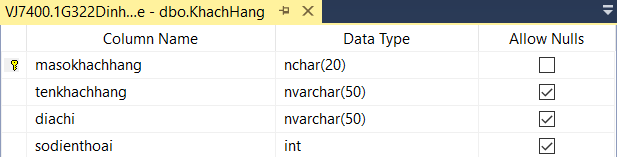
# 3.3.6 Thiết kế các Table DanhMucTheLoai



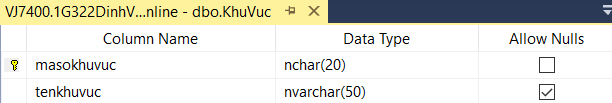
# 3.3.7 Thiết kế các Table DiemBanVe



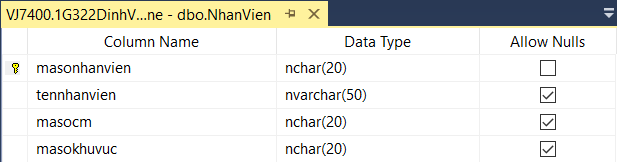
# 3.3.8 Thiết kế các Table KhachHang



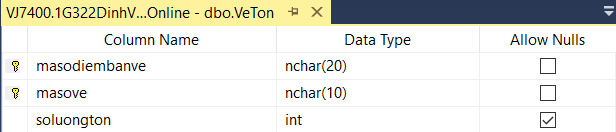
# 3.3.9 Thiết kế các Table KhuVuc



# 3.3.10 Thiết kế các Table NhanVien

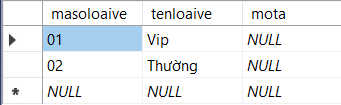


# 3.3.11 Thiết kế các Table NhanVien

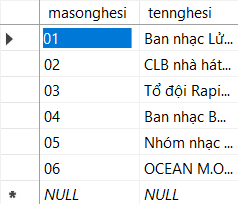


## Cập nhật dữ liệu DB thực nghiệm cho WebApp bán vé online

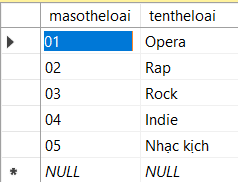
Dữ liệu DanhMucLoaiVe



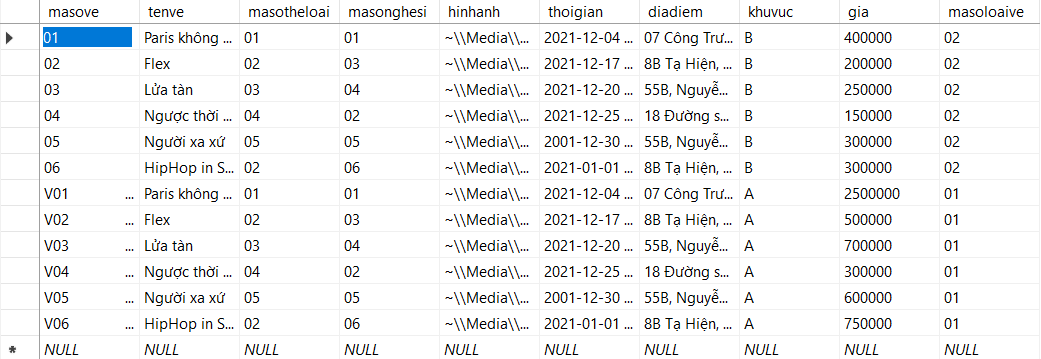
Dữ liệu DanhMucNgheSi



Dữ liệu DanhMucTheLoai



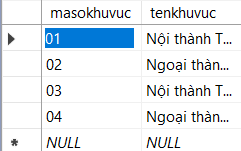
Dữ liệu DanhMucVe



Dữ liệu DiemBanVe



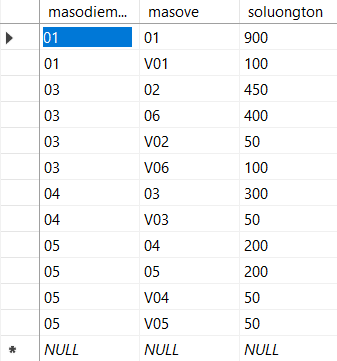
Dữ liệu KhuVuc



Dữ liệu NhanVien



Dữ liệu VeTon



## Giới thiệu một số SP và Trigger trong DB quản lý bán vé online

# THIẾT KẾ VÀ LẬP TRÌNH WEBAPP PHẦN GIAO DỊCH

## Giới thiệu về WebApp bán vé online

Đây là thành phần các WebFroms phục vụ cho khách hàng thực hiện các chức năng giao dịch đặt vé cần mua: thông thường đây là các chức năng phục vụ các cá nhân, đơn vị bên ngoài đơn vị bán vé tương tác online với hệ thống nên không cho phép làm thay đổi các thông tin bên trong hệ thống. Chủ yếu là các thao tác đặt hàng, góp ý, bình luận, đề xuất, đề nghị …. Không thực hiện Thêm, Xóa…

Phần giao dịch của WebApp này bao gồm:

+ MasterPage.Master: là Master mẫu dùng chung cho các trang giao dịch và các trang quản lý của Chương 5: trang 32.

+ HomePage.aspx: là trang chủ, trưng bày các loại vé cho phép khách hàng thực hiện chọn vé bỏ vào “vé đặt mua” để đặt mua vé.

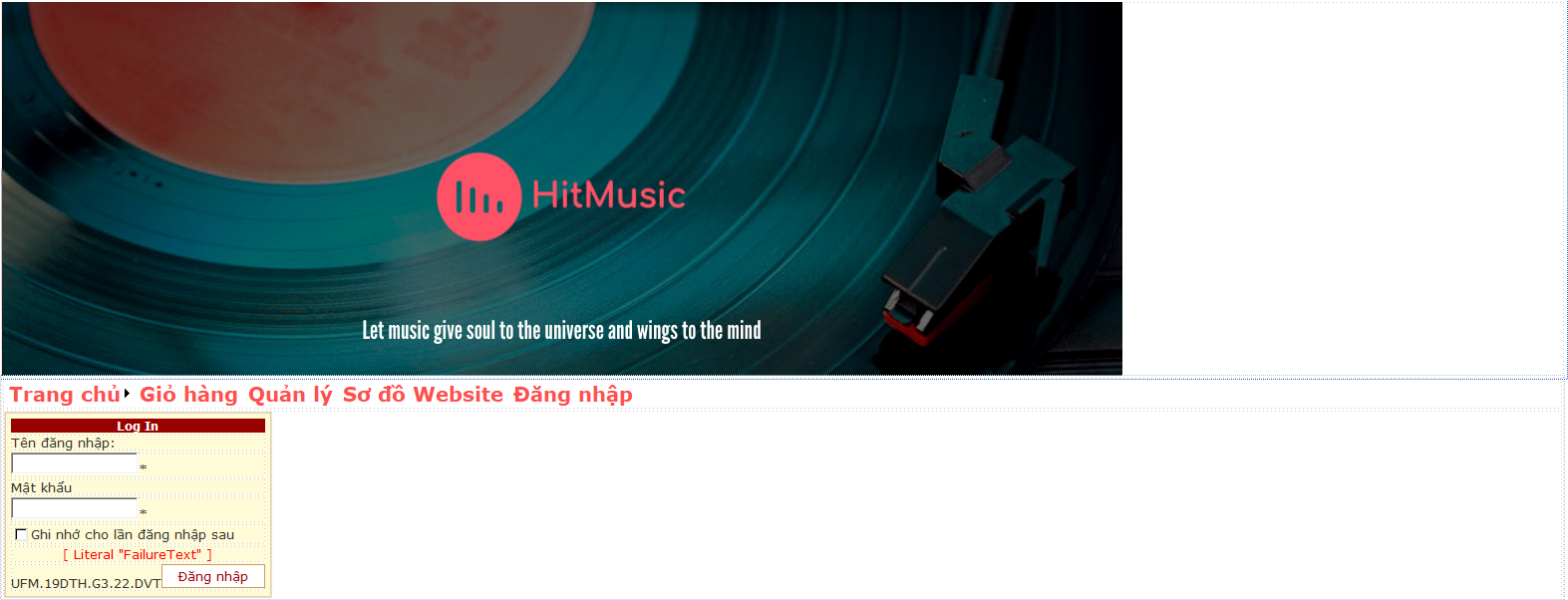
+ Cart.aspx: cho phép khách hàng xem nội dung của “vé đặt mua” hiện có và thực hiện một số thao tác cơ bản trên “vé đặt mua” như: “Mua thêm”, “Hủy giỏ hàng”, “Đặt mua”, “Đổi hàng” , Bỏ bớt hàng trong giỏ ra”. . . .

+ FrOrder.aspx: cho phép khách hàng thực hiện việc đặt mua vé khi “vé đặt mua” không rỗng (có vé).

## MasterPage giao dịch : MasterPage.Master

## Thiết kế Masterpage giao dịch

**THIẾT KẾ MASTERPAGE (Design View)**



Hình ‑. Thiết kế MasterPage Giao dịch dùng chung cho cả phần quản lý

Trong thiết kế MasterPage như Hình 4‑1 bao gồm:

+ Thiết kế Table có 1 cột và nhiều dòng tạo thuận lợi cho việc kết cấu vị trí các thành phần thiết kế trên các WebPage thừa kế MasterPage này.

+ Banner giới thiệu về cơ sở kinh doanh vé trực tuyến

+ Menu ngang (Horizontalization) cho phép khách hàng truy xuất các trang thông tin liên quan đế hệ thống Websites.

## Lập trình (Codes) MasterPage giao dịch

/// <summary>

/// Thủ tục tự chạy khi mở trên web browser

/// ẩn login nv

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

protected void Page\_Load(object sender, EventArgs e)

{

LoginNv.Visible = false;

}

/// <summary>

/// Viết thủ tục ẩn hiện login (trên MasterPage.master)

/// </summary>

/// <param name="sender"></param>

/// <param name="e">lưu trữ thông tin mục chọn trên menumain mà nhân viên đã chọn</param>

protected void MenuMain\_MenuItemClick(object sender, MenuEventArgs e)

{

if(e.Item.Value == "ql" || e.Item.Value == "dn") // KHINSD(NV) chọn quản lý hoặc đăng nhập

{

LoginNv.Visible = true; // mở công cụ loginnv để nv đăng nhập để thực hiện các chức năng quản lý hệ thống

}

else // khi nv chọn các mục khách

{

LoginNv.Visible = false;

}

}

/// <summary>

/// Thủ tục xử lí đăng nhập

/// </summary>

/// <param name="sender"></param>

/// <param name="e">Ghi nhận việc đăng nhập thành công</param>

protected void LoginNv\_Authenticate(object sender, AuthenticateEventArgs e)

{

if ((LoginNv.UserName == "Admin" && LoginNv.Password == "admin@vn") ||

(LoginNv.UserName == "Guest" && LoginNv.Password == "guest@vn"))

{

e.Authenticated = true; // đăng nhập thành công, tự chuyển đến trang quản lý

}

else // đăng nhập không thành công

{

e.Authenticated = false;

}

}

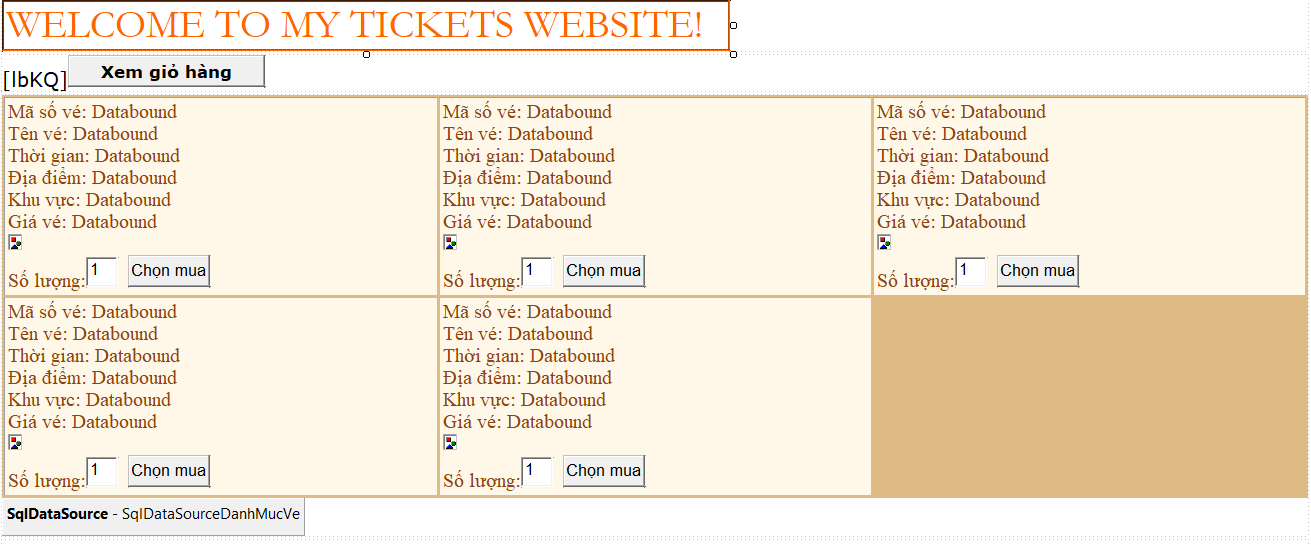
**MÃ LỆNH LẬP TRÌNH CỦA MASTERPAGE (Code View)**

## Trang chủ trưng bày các loại vé: HomePage.aspx

## Thiết kế (Design) WebApp giao dịch vé online

\* Hệ Websites hỗ trợ chức năng giao dịch vé trực tuyến thừa kế Masterpage nên trên, gồm các thành phần như sau:

**THIẾT KẾ TRANG CHỦ HomePage (Design View)**



Hình ‑. Thiết kế trang chủ HomePage

Trong thiết kế trang chủ HomePage như Hình 4‑2 bao gồm:

+ Thiết kế Table có 1 cột và nhiều dòng tạo thuận lợi cho việc kết cấu vị trí các thành phần thiết kế trên HomepagePage.

+ Dùng DataList để trưng bày các loại vé cho phép khách hàng chọn mua trực tuyến

+…

## Lập trình (Codes) WebApp giao dịch vé online: Homepage.aspx

static DataTable cart = new DataTable();

// biến toàn cục cart dùng để dùng lưu giỏ hàng, static cấp phát vùng nhớ "tĩnh"

// Nhiệm vụ chính: khởi tạo giỏ hàng nếu có trong sesion thì lấy xuống, nếu chưa thì khởi tạo

protected void Page\_Load(object sender, EventArgs e)

{

if(!IsPostBack) // chỉ thiết lập giỏ hàng mới khi homepage được load lần đầu ,các lần sau không

{

if(Session["cart"] != null) // đã có giỏ hàng

{

cart = Session["cart"] as DataTable; // lấy giỏ hàng từ ""session" xuống để dùng

}

else

{

cart.Rows.Clear(); // xóa các dòng và cột "rác" trong giỏ hàng

cart.Columns.Clear();

cart.Columns.Add("masove", typeof(string));

cart.Columns.Add("tenve", typeof(string));

cart.Columns.Add("gia", typeof(float));

cart.Columns.Add("soluong", typeof(int));

cart.Columns.Add("tien", typeof(float), "soluong \* gia"); // thuoc tính tahnhf tiền được tính tự động = số lượng\*đơn giá

} //else if

}

} // page load

/// <summary>

/// Thủ tục = Bỏ hàng vào giỏ = Khi khách hàng online chọn nút lệnh "Chọn mua" một mặt hàng item/ datalistdmve

/// thục tục thực hiện khi người dùng chọn 1 lệnh (nhận diện nhờ thuộc commandname) nào đó trong 1 item datalist

/// </summary>

/// <param name="source"></param>

/// <param name="e"></param>

protected void DataListDMVe\_ItemCommand(object source, DataListCommandEventArgs e) // tham số e.Item lưu giữ các thông tin cơ bản

{

if(e.CommandName == "chon")// nsd bấm nút "chọn mua"

{

//B1: Lấy các thông tin món hàng [item] mà khách hàng đang chọn

string masove = ((Label)e.Item.FindControl("masoveLabel")).Text.Trim(); //datalistDMVe.Datakeys(e.item.itemindex)

string tenve = ((Label)e.Item.FindControl("tenveLabel")).Text;

float gia = float.Parse(((Label)e.Item.FindControl("giaLabel")).Text.Trim());

int soluong = int.Parse(((TextBox)e.Item.FindControl("txtSL")).Text.Trim());

//B2: Kiểm tra item (item có chưa, nếu có thì chỉ tăng số lượng)

foreach(DataRow r in cart.Rows) // dò từng dòng hàng có trong vỏ

{

if (r["masove"].ToString() == masove)//if ((string)r["masove"] == masove) || Mặt hàng kh chọn đã có trong giỏ

{

r["soluong"] = (int)r["soluong"] + soluong;//tăng số lượng vừa chọn

goto kt; // đã tìm thấy hàng trong giỏ kết thúc foreach => nhảy đến B4 (bỏ qua B3: Thêm mặt hàng cart vào giỏ)

}

}

//B3: Thêm mặt hàng mới vào giỏ khi giỏ hàng trống

cart.Rows.Add(masove,tenve,gia,soluong);

//B4: Gửi giỏ hàng lên session để chuyển sang các trang khác sử dụng sau khi đã thêm vào giỏ

kt:

Session["cart"] = cart;

//B5: Thông báo kết quả

lbKQ.Text = "Hiện tại trong giỏ hàng đang có: " + cart.Compute("Count(masove)","").ToString() + " Vé, Tổng tiền: " + cart.Compute("Sum(tien)","").ToString() + " đồng";

} // if chọn mua

} // itemcommand của datalistdmve

/// <summary>

/// Xem giỏ hàng

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

protected void btnCart\_Click(object sender, EventArgs e)

{

Response.Redirect("~\\Cart.aspx");

}

**MÃ LỆNH LẬP TRÌNH CỦA HOMEPAGE (Code View)**

B1: Khai báo đường dẫn thư viện NameSpace

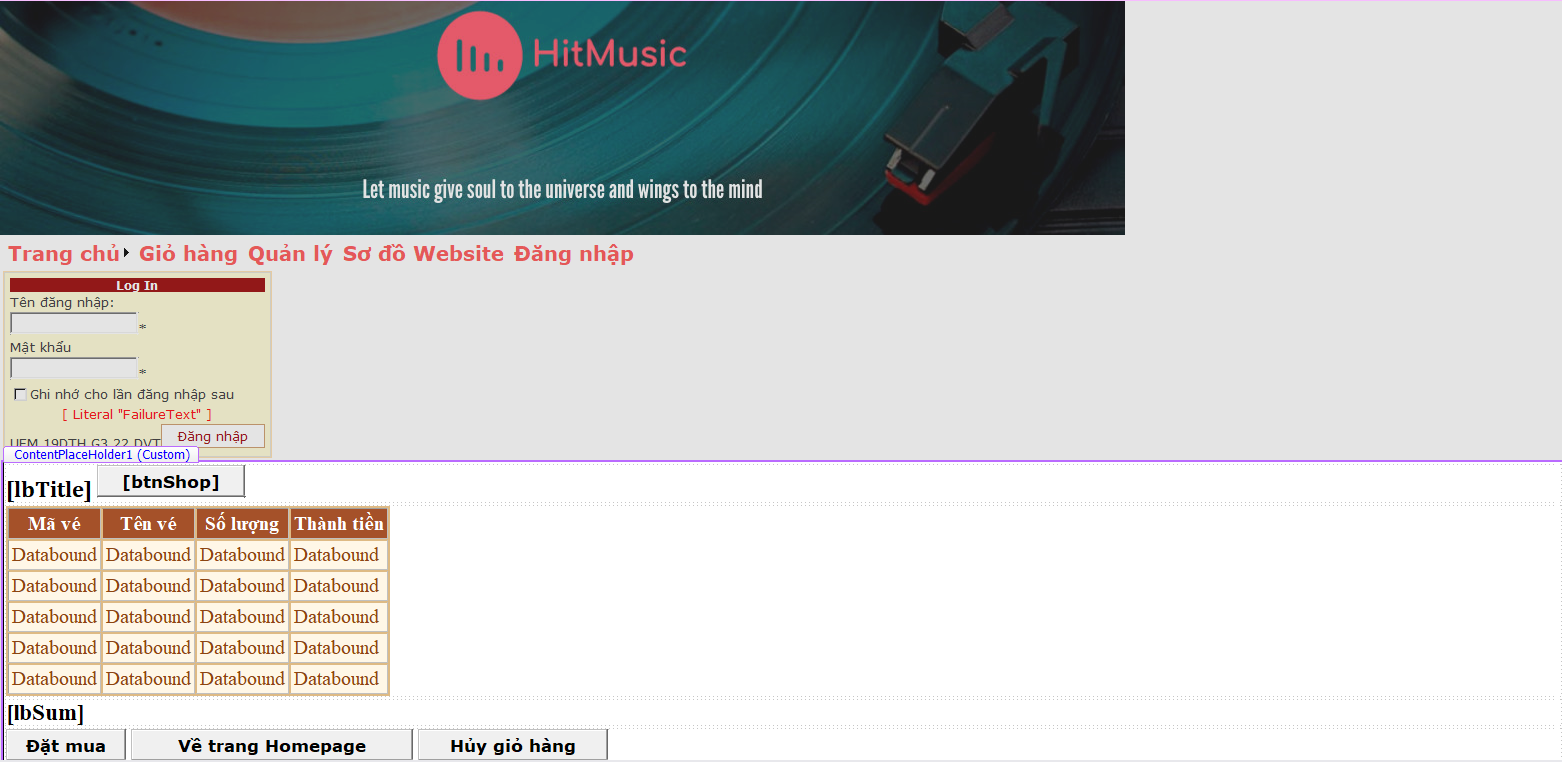
B2: Khai báo biến toàn cục cart thuộc kiểu DataTable để lưu giữa “giỏ hàng”:

B3: Khởi tạo “giỏ hàng” trong Page\_Load()

## Trang “Giỏ hàng” của khách hàng: Cart.aspx

## Thiết kế (Design) WebPage “Giỏ hàng”

\* Webpage “giỏ hàng” của vé trực tuyến thừa kế Masterpage nên trên, gồm các thành phần như sau:



## Mã lệnh (Codes) WebPage “Giỏ hàng”

static DataTable cart = new DataTable(); // biến toàn cục cart dùng để dùng lưu giỏ hàng, static cấp phát vùng nhớ "tĩnh"

/// <summary>

/// Thủ tục tự động chạy mỗi khi trang này được tải. nhiệm vụ chính: tải giỏ hàng lên nếu chưa thì thông báo rỗng

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

protected void Page\_Load(object sender, EventArgs e)

{

if (Session["cart"] == null) // giỏ hàng rỗng

{

lbTitle.Text = "HIỆN TẠI, GIỎ HÀNG CỦA QUÝ KHÁCH ĐANG TRỐNG, XIN MỜI SANG TRANG CHỦ CHỌN VÉ";

btnShop.Text = "Chọn mua!";

GridViewCart.DataSource = null; // Xóa trống danh mục vé trong giỏ

lbSum.Text = ""; // Xóa trống tính tiền

btnOrder.Visible = false; // giỏ hàng trống => Không có gì để mua để hủy

BtnCancel.Visible = false;

}

else // giỏ hàng đã có

{

lbTitle.Text = "GIỎ HÀNG HIỆN GIỜ CỦA QUÝ KHÁCH NHƯ SAU: ";

btnShop.Text = "Mua tiếp!"; // vì đã có hàng trong giỏ => về trang chủ đẻ mua tiếp

//B1: Lấy giỏ hàng từ Session xuống

cart = Session["cart"] as DataTable; // cart = (datatable)Session["cart"]

// B2: Gán giỏ hàng vào Gridviews

GridViewCart.DataSource = cart;

//B3: Tải dữ liệu lên Gridviews

GridViewCart.DataBind();

//B4: Tính tiền

lbSum.Text= "Giỏ hàng hiện có: " + cart.Compute("Count(masove)", "").ToString() + " Vé, Tổng tiền: " + cart.Compute("Sum(tien)", "").ToString() + " đồng";

btnOrder.Visible = true; // giỏ hàng có => có thể đặt mua , hủy

BtnCancel.Visible = true;

} // if else session

}

/// <summary>

/// Về trang chủ (trưng bày hàng hóa) để chọn mua hoặc mua tiếp

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

protected void btnShop\_Click(object sender, EventArgs e)

{

Response.Redirect("~\\HomePage.aspx");

}

/// <summary>

/// Hủy giỏ hàng

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

protected void BtnCancel\_Click(object sender, EventArgs e)

{

Session["cart"] = null; //Hủy giỏ hàng (session) trên bộ nhớ trong (ram)

//Response.Redirect("~\\HomePage.aspx");

btnShop\_Click(sender, e); //Gọi sự kiện (Event) của nút btnShop

}

## Đơn đặt vé: FrOrder.aspx

## Giới thiệu

Trang này (FrOrder.aspx) được gọi từ trang “Gỏi hàng” [FrCart.aspx] không rỗng

\* Nếu giỏ hàng đã có thì có thể gọi được trang đặt hàng: **FrOrder.asp**

protected void btnOrder\_Click(object sender, EventArgs e)

{

Response.Redirect("~\\FrOrder.aspx");

}

\* Để thực hiện được trang đặt hàng: **FrOrder.aspx** thì cần phải “chạy” từ trang chủ (FrHomePage.aspx: trưng bày hàng hóa), cho phép khách hàng chọn hàng “bỏ vào giỏ” và đã có “giỏ hàng” : Set As Start Page từ HomePage.aspx

\* ĐẶT HÀNG PHẢI XÁC ĐỊNH 3 NỘI DUNG:

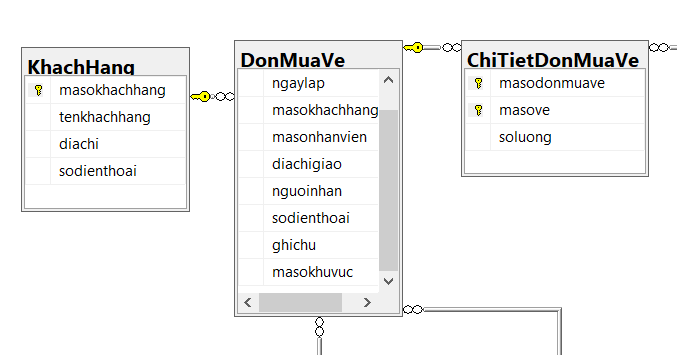
+ Khách hàng <=> Table [KhachHang]

+ Thông tin đơn hàng: msddh, ngày lập,… <=> Table [DonMuaVe]

+ Đơn này đặt mua những mặt hàng nào <=> Table [ChiTietDonMuaVe]

Sau khi đặt hàng => Giao hàng ở đâu, cho ai, số điện thoại người nhận, hạn giao

=> Phương thức thanh toán, số tiền ?



## Thiết kế trang Đặt hàng: FrOrder.aspx

Thiết kế FORM “Đặt hàng”: có 3 phần

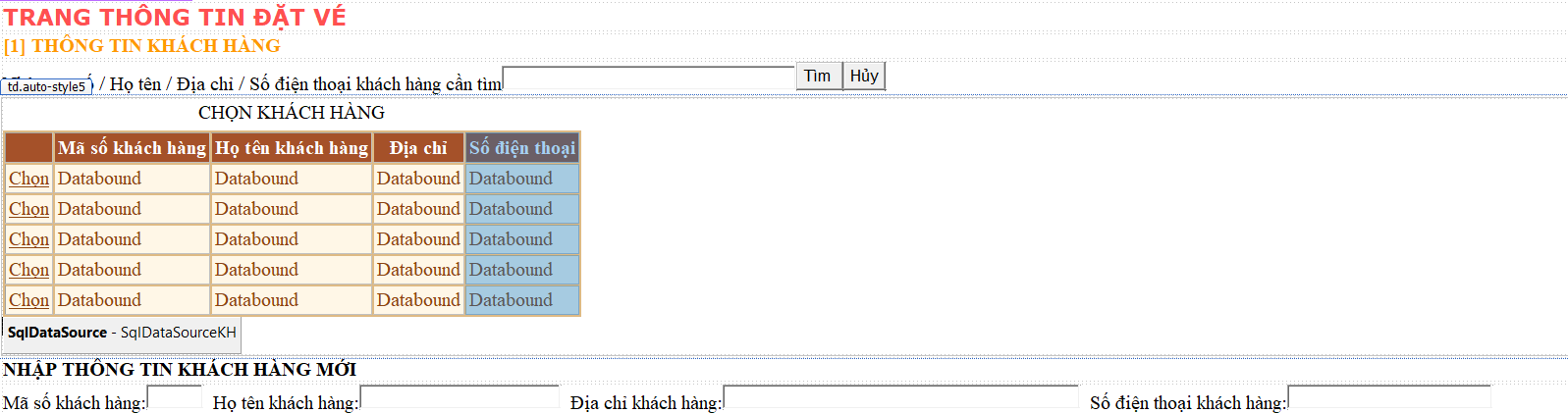
\* Thông tin <Khách hàng>: Tìm (mskh, hoten, diachi, sodienthoai,…); nếu tìm ko có thì cho Thêm <Khách hàng> mới

\* Thông tin <Đơn mua vé> (xem table DonMuaVe trong DB): má số đơn mua, nhân viên giao, ngày lập (ngày giờ hiện tại), địa chỉ giao hàng, người nhận, số điện thoại người nhận, khu vực, ghi chú.

\* Thông tin “giỏ hàng”: lấy từ Trang “giỏ hàng” sang, để khách hàng xem và kiểm tra các loại vé đã đặt, tổng tiền, . . .Không Thêm / Xóa /Sửa (việc này thực hiện trên trang “giỏ hàng”).

Vì vậy,

**PHẦN THÔNG TIN KHÁC HÀNG:** Tìm khách hàng, nếu chưa có thì thêm mới, nếu đã có thì chọn KH (vì có thể tìm thấy nhiều KH), được thiết kế như Hình 4‑3



Hình ‑. Thiết kế phần thông tin KH của Đơn hàng

+ Nếu tìm có KH thì phần thông tin KH mới sẽ bị ẩn; .Visible = False

+ Ghi chú: Khi GridView rỗng (không có dòng nào) nó sẽ tự ẩn.

+ Bổ sung Nút “**Chọn**” trên mỗi dòng của GridViewKH

**PHẦN THÔNG TIN ĐƠN HÀNG**

+ Thông tin đơn hàng được thiết kế theo Table DonMuaVe đã có trong DB:

DonMuaVe(masodonmuave, ngaylap, diachi,……)

. Mã số đơn mua được xác định khi khách hàng được “Chọn” (.Visible = false)

. Khi Chọn khách hàng (phần trên): cho .Visible = true

. Ngày lập là ngày hiện tại *(sau này có thể cho khách hàng) -> dùng Calendar thay đổi ngày khác)*

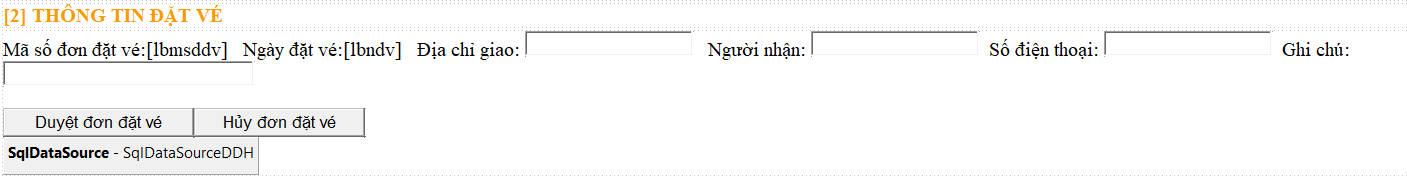
. masokhachhang = lấy từ txtTimKH (đã xác định duy nhất 1 khách hàng trong phân trên)

. masonhanvien: khi khách hàng đặt hàng (ở đây: FrOrder) nên chưa biết nhân viên giao hàng đơn vì vậy null. Mã số nhân viên này chỉ có khi Trưởng bộ phân phân công giao hàng phân công nhiệm vụ giao nên sẽ cập nhật masonhanvien sau. Đặt masonhanvien / Table DonMuave / MS.SQL Allow Null

. masokhuvuc: do Trưởng bộ phận phân công xác định nên Allow Null

+ Thông tin đơn hàng chỉ hiển thi khi đã có thông tin KH, tức là trong phần thông tin KH đã “Chọn”, vì vậy, ban đầu, các đối tượng thiết kế trong phần thông tin Đơn hàng sẽ có .Visible = False.

+ Các nội dung thiết kế cơ bản là Lable, TextBox, như Hình 4‑4



Hình ‑. Thiết kế phần thông tin Đơn mua trên trang Order

**PHẦN THÔNG TIN CÁC MẶT HÀNG ĐẶT MUA CỦA ĐƠN HÀNG (GIỎ HÀNG)**

## Lập trình mã lệnh (Codes) của trang Đặt hàng: Managements.aspx

**PHẦN THÔNG TIN KHÁC HÀNG**

\* Tìm khách hàng, nếu chưa có thì thêm mới, nếu đã có thì chọn khách hàng vì có thể tìm thấy nhiều khách hàng:

**+ Lệnh SQL tìm khách hàng**

SELECT masokhachhang, tenkhachhang, diachi, sodienthoai

FROM KhachHang

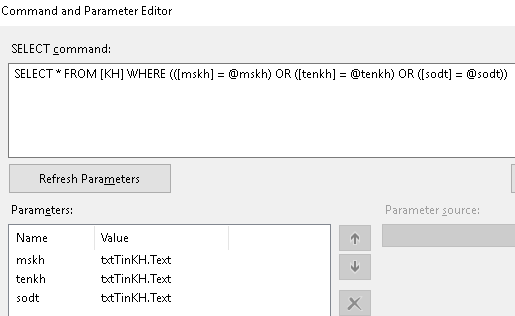
WHERE (masokhachhang = @masokhachhang) OR

(tenkhachhang = @tenkhachhang) OR

(diachi = @diachi) OR

(sodienthoai = @sodienthoai)

Với tham số như Hình 4‑4



Hình ‑4. Tham số lệnh SQL tìm KH

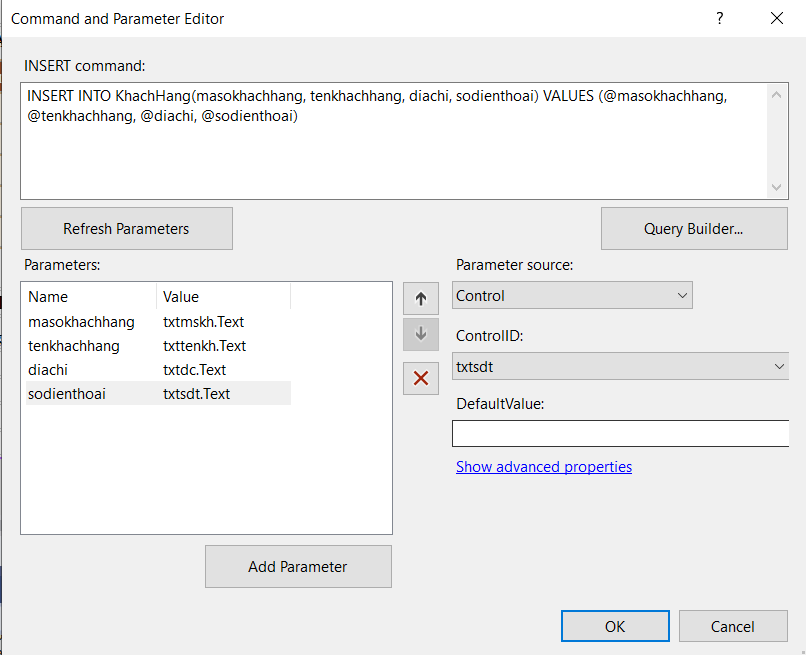
**+ Lệnh SQL thêm khách hàng mới mới: Insert: SqlDataSourceKH**

INSERT INTO KhachHang

(masokhachhang, tenkhachhang, diachi, sodienthoai)

VALUES (@masokhachhang,@tenkhachhang,@diachi,@sodienthoai)

Với tham số như Hình 4‑5



Hình ‑5. Tham số lệnh SQL thêm KH mới

**+ Lệnh tìm khách hàng:**

\* Viết lại mã lệnh (codes) “Tìm KH”:

**[1] Chỉ cho phép tìm khách hàng khi đã nhập thông tin vào ô Textbox tìm**

**[2] Tìm khách hàng** [thủ tục chính] = Tim KH + Nhập KH mới + Lưu DB + chọn KH

// [1].TÌM KH + [2].NHẬP KH MỚI + [3] LƯU THÔNG TIN KH MỚI VỪA NHẬP +   
 [4]CHỌN KH TRONG GRIDVIEW TÌM THẤY (TẠI VỊ TRÍ vt ĐÃ CHỌN TRÊN GIRDVIEW)

**[3] Hủy kết quả tìm và tìm lại KH khác**

// HỦY : TÌM LẠI KH KHÁC

**PHẦN THÔNG TIN ĐƠN HÀNG**

[1] Hiển thị mã số đơn hàng khi đã chọn KH

PHẦN THÔNG TIN CÁC MẶT HÀNG ĐẶT MUA CỦA ĐƠN HÀNG (GIỎ HÀNG)

**FULL CODES**

//TỰ ĐỘNG CHAY KHI TRANG ĐẶT HÀNG ĐƯỢC TẢI LÊN

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Data;

namespace G322DinhVanThanhVeOnline

{

public partial class FrOrder : System.Web.UI.Page

{

static DataTable cart = new DataTable();//biến toàn cục cart = sẽ dùng để lưu giữ "giỏ hàng"

//static = cấp phát vùng nhớ "tĩnh" cho biến cart => giá trị trong cart khi ra khỏi 1 thủ tục KHÔNG bị mất

protected void Page\_Load(object sender, EventArgs e)

{

}

protected void btnTimKH\_Click(object sender, EventArgs e)

{

btnTimKH.Visible = true;//cho phép hủy tìm lại (bất kỳ lúc nào)

if (btnTimKH.Text.Trim() == "Tìm")//[1] TÌM KHÁCH HÀNG

{

GridViewKH.DataBind(); // tìm kh

if(GridViewKH.Rows.Count <1)

{

btnTimKH.Text = "Thêm khách hàng mới"; //không tìm thấy KH như vậy (theo nd NSD nhập txtTimKH) =>THÊM KH MỚI

}

else // tìm thấy ít nhất 1 khả năng

{

btnTimKH.Text = "Chọn khách hàng"; // được chọn = xử lí ở dưới

txtmskh.Text = GridViewKH.Rows[0].Cells[1].Text.Trim(); //tạm gán mã kh đầu tiên tìm thấy row[0] vào txtmskh, cell[1]: phòng khi nsd kh chọn ai cả

}

}

else if(btnTimKH.Text.Trim() == "Thêm khách hàng mới") // nhập kh mới

{ //hiện thị các ô

lbnhaptt.Visible = true;

lbmskh.Visible = true;

lbtenkh.Visible = true;

lbdc.Visible = true;

lbsdt.Visible = true;

txtmskh.Visible = true;

txttenkh.Visible = true;

txtdc.Visible = true;

txtsdt.Visible = true;

// hien thi cac o nhap tt moi

txtmskh.Text = txtTimKH.Text.Trim();

txtmskh.ToolTip = "KHÔNG ĐỂ TRỐNG MÃ KHÁCH HÀNG & KHÔNG NHẬP TRÙNG";

//gán nd nsd nhập mskh đảm bảo kh trùng // sau này cho thuộc tính này tự động

txttenkh.Text = "";

txtdc.Text = "";

txtsdt.Text = "";

//đổi nhãn nút lệnh

btnTimKH.Text = "Lưu thông tin khách hàng mới";

}

else if(btnTimKH.Text.Trim() == "Lưu thông tin khách hàng mới")

{ //lưu thông tin khách hàng mới đã nhập vào db

try

{

SqlDataSourceKH.Insert(); //lưu kh mới vào db

txtTimKH.Text = txtmskh.Text.Trim(); // gán mskh tìm kh, tìm đúng kh vừa thêm

GridViewKH.DataBind();

this.Title = "ĐÃ THÊM THÀNH CÔNG";

}

catch (System.Exception ex)

{

this.Title = "LỖI THÊM KHÁCH HÀNG MỚI" + ex.Message;

}

// an tt

lbnhaptt.Visible = false;

lbmskh.Visible = false;

lbtenkh.Visible = false;

lbdc.Visible = false;

lbsdt.Visible = false;

txtmskh.Visible = false;

txttenkh.Visible = false;

txtdc.Visible = false;

txtsdt.Visible = false;

//đổi nhãn

btnTimKH.Text = "Chọn khách hàng"; //CÓ 2 tr. h CHỌN = khi tìm thấy KH {phía trên} hoặc khi nhập mới KH, tại đây

}

else if (btnTimKH.Text == "Chọn khách hàng") // chọn khách hàng mới trong gridview tìm thấy

{// gán mskh vào ô txt tìm (thống nhất chọn 1 kh)

txtTimKH.Text = txtmskh.Text.Trim(); //khi chọn kh phải gán mã kh đã chọn vào txtmskh (SelectedIndexChanged)

// tải dl khách hàng

GridViewKH.DataBind();

// khong cho chon lai, muon chon lai phai huy roi chon lai

btnTimKH.Enabled = false;

// xu li thong tin don hang

lbtd2.Visible = true;

lbmasoddv.Visible = true;

lbngaydv.Visible = true;

lbdcg.Visible = true;

lbnguoinhan.Visible = true;

lbsdtg.Visible = true;

lbgc.Visible = true;

lbmsddv.Visible = true;

lbndv.Visible = true;

txtdcg.Visible = true;

//gán ĐT KH vào đây = mặc định ĐT giao hàng là DT của KH

txtdcg.Text = GridViewKH.Rows[vt].Cells[3].Text.Trim();

txtnguoinhan.Visible = true;

//gán Tên KH vào đây = mặc định Người nhận là KH

txtnguoinhan.Text = GridViewKH.Rows[vt].Cells[2].Text.Trim();

txtsdtg.Visible = true;

//gán ĐT KH vào đây = mặc định ĐT giao hàng là DT của KH

txtsdtg.Text = GridViewKH.Rows[vt].Cells[4].Text.Trim();

txtgc.Visible = true;

DateTime t = System.DateTime.Now;

lbndv.Text = t.ToShortDateString().Trim(); //Ngày lập = ngày ngày hiệntại => sv cho KH đổi = Calendard

//msddv = SV nên cho mặc định tăng theo quy luật = năm \_ quý \_....(tạm thời gán như sau)

lbmsddv.Text = txtTimKH.Text.Trim() + t.Year.ToString().Substring(2, 2) + t.Month.ToString().Trim() + t.Day.ToString().Trim() + t.Hour.ToString().Trim() + t.Minute.ToString().Trim() + t.Second.ToString().Trim();

// Cùng 1 KH đặt 2 đơn hàng phải cách nhau ít nhất 1 s

btnAcceptDMV.Visible = true;

btnHuyDMV.Visible = true;

//4.3.2 Hiện thị thông tin CHI TIẾT "Đơn hàng"

lbtd3.Visible = true;

cart = Session["cart"] as DataTable;

GridViewCart.DataSource = cart;

GridViewCart.DataBind();

lbsum.Text = "Giỏ hàng hiện có " + cart.Compute("Count(masove)", "").ToString() + " MẶT HÀNG, TỔNG TIỀN = " + cart.Compute("Sum(gia)", "").ToString() + " đ";

GridViewCart.Visible = true;

lbsum.Visible = true;

}

}

static int vt = 0;

protected void GridViewKH\_SelectedIndexChanged(object sender, EventArgs e)

{

vt = GridViewKH.SelectedIndex;

txtmskh.Text = GridViewKH.Rows[vt].Cells[1].Text.Trim();//gán mã KH đã chọn vào TxtMaKH; ....

}

protected void btnHuyTimKH\_Click(object sender, EventArgs e)

{

//1. Cho "Tìm lại"

btnTimKH.Text = "Tìm";

btnTimKH.Enabled = true;

//2. Ẩn các ô nhập thông tin KH mới

lbnhaptt.Visible = false;

lbmskh.Visible = false;

lbtenkh.Visible = false;

lbdc.Visible = false;

lbsdt.Visible = false;

txtmskh.Visible = false;

txttenkh.Visible = false;

txtdc.Visible = false;

txtsdt.Visible = false;

//3. Xóa trống TextBox nhập thông tin tim, để nhập lại

txtTimKH.Text = "";

//4. Nạp lại (trống) GridView ds KH, tránh trường hợp trước đã tìm có ds KH

GridViewKH.DataBind();

//5. Ẩn chính nó

btnHuyTimKH.Visible = false;

//6. HỦY "ĐƠN HÀNG"

//6.1 Hiện thị thông tin "Đơn hàng"

lbtd2.Visible = false;

lbmasoddv.Visible = false;

lbngaydv.Visible = false;

lbdcg.Visible = false;

lbnguoinhan.Visible = false;

lbsdtg.Visible = false;

lbgc.Visible = false;

lbmsddv.Visible = false;

lbndv.Visible = false;

txtdcg.Visible = false;

txtdcg.Text = "";

lbmsddv.Text = "";

txtdcg.Visible = false;

txtnguoinhan.Visible = false;

txtsdtg.Visible = false;

txtgc.Visible = false;

btnAcceptDMV.Visible = false;

btnHuyDMV.Visible = false;

//4.3.2 Hiện thị thông tin CHI TIẾT "Đơn hàng"

lbtd3.Visible = false;

lbsum.Text = "";

GridViewCart.Visible = false;

lbsum.Visible = false;

}

protected void btnAcceptDMV\_Click(object sender, EventArgs e)

{

//THÊM ĐƠN & CTDDH vào DB

try

{

SqlDataSourceDDH.Insert();

foreach (DataRow r in cart.Rows)

{

SqlDataSourceCTDDH.InsertParameters.Clear();

SqlDataSourceCTDDH.InsertParameters.Add("masodonmuave", lbmsddv.Text.Trim());

SqlDataSourceCTDDH.InsertParameters.Add("masove", r["masove"].ToString().Trim());

SqlDataSourceCTDDH.InsertParameters.Add("soluong", DbType.Double, r["soluong"].ToString().Trim());

SqlDataSourceCTDDH.Insert();

}

this.Title = "Thêm DDH & CTDDH xong";

}

catch (System.Exception ex) { this.Title = "LỖI THÊM DDH & CTDDH = " + ex.Message; }

//KHÓA "Đơn hàng"

txtgc.Enabled = false;

txtdcg.Enabled = false;

txtnguoinhan.Enabled = false;

txtsdtg.Enabled = false;

btnAcceptDMV.Enabled = false;

//KHÔNG HỦY KH = HỦY ĐƠN HÀNG = HỦY

btnHuyTimKH.Enabled = false;

}

}

}

# THIẾT KẾ VÀ LẬP TRÌNH WEBAPP PHẦN QUẢN LÝ

## Giới thiệu về WebApp quản lý hệ thống bán vé Online

**[1] Chức năng chính của phần quản lý**

\* Hệ thống này dành cho nhân viên của đơn vị kinh doanh vé thực hiện các chức năng cập nhật thông về vé sự kiện âm nhạc, dịch vụ kinh doanh vé online, như: bổ sung vé mới, thay đổi giá bán, cập nhật chính sách khuyến mãi,…..

\* Vì vậy, các chức năng quản lý này chỉ cho phép nhân viên của đơn vị kinh doanh thực hiện nên cần phải thực hiện thủ tục đăng nhập thành công trước khi thực hiện các nghiệp vụ quản lý và các …..

Vì vậy chức năng LOGIN được thực hiện ở phần này là dành cho nhân viên sử dụng site online này quản lý hệ thống. Accounts do người quản trị hệ thống (sa: System Administrator) cấp; KHÔNG tự đăng kí account. (khác với LOGIN của khách hàng, do khách hàng tự đăng kí)

**[2] Thiết kế và lập trình chức năng đăng nhập dành cho nhân viên**

**PHÂN TÍCH:**

\* Từ trang HomePage, các nhân viên phải thực hiện đăng nhập thành công mới được truy xuất trang "Quản lý".

Trong đề tài tạm sử dụng 2 Accounts mặc định:

Username: Admin với Password: admin@vn

Username: Guest với Password: guest@vn

\* Vì các nhân viên có thể đăng nhập và thực hiện các chức năng quản lý liên quan từ bất kỳ trang web nào trong hệ thống nên thiết kế và lập trình chức năng đăng nhập (Login) phải được thực hiện trên Masterpage ( thay vì trên Homepage: muốn đăng nhập phải vào Homepage, các trang khác không đăng nhập được )

\* Ngoài ra còn có các chức năng phân quyền:

Sử dụng các công cụ khác của Login/Toolbox: đổi mật khẩu, hiện tên đăng nhập/ Managements.aspx, Hiện các chức năng Đăng nhập/Đăng xuất, LoginView

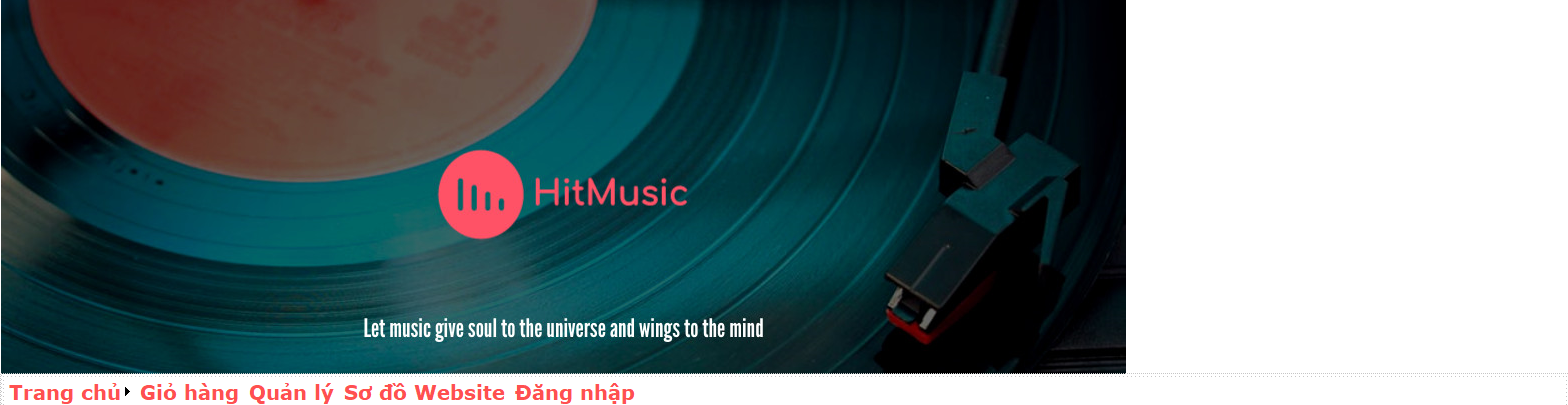
{Bên khách hàng có thêm các chức năng}

## Masterpage quản lý hệ thống bán vé online

## Thiết kế Masterpage quản lý

* Sử dụng lại Masterpage giao dịch MasterPage.master bằng cách thiết kế và lập trình bổ sung thực đơn (Menu quản lý) như Hình 5‑1 trang 17

**THIẾT KẾ: Designing (MasterPage.master):**



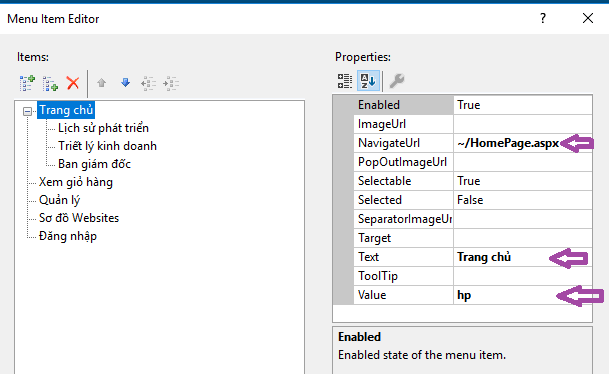
Hình ‑. Thiết kế Menu chính trên MsterPage

**Menu** / Navigation **: MenuMain**

.Orientation = Horizontal

**.** [Task]->Edit menu item || Properties: Item …: định nghĩa các mục chọn menu

Item : .Text .Value .NavigateURL : như Hình 5‑2 trang 17

****

Hình ‑. Thiết lập thuộc tính cho mục chọn (Item) của Menu

**LoginNv**

\*Visible: False : LoginNv sẽ ẩn chỉ khi NSD (Nhân viên) click vào “Quản lý” (Value = ql)

Or “Đăng nhập” (Value = dn) trên MenuMain trên MasterPage

\*Việt hóa:

+ Your login attempt was not successful. Please try again.

Quý vị đã đăng nhập không thành công. Vui lòng đăng nhập lại.

+ Log In: Đăng nhập = ĐĂNG NHẬP

+ Password: Mật khẩu:

+ User Name: Tên đăng nhập:

+ Remember me next time.: Ghi nhớ cho lần đăng nhập sau

+ Password is required.: Không được để trống mật khẩu

+ User Name is required.: Không được để trống tên đăng nhập

\*Khi sử dụng công cụ Login có thể có lỗi như sau: (liên quan đến bảo mật)

“WebForms UnobtrusiveValidationMode requires a ScriptResourceMapping for 'jquery'. Please add a ScriptResourceMapping named jquery(case-sensitive).”

Để khác phục lỗi này, trong file Web.cofig thêm đoạn mã lệnh sau:

<appSettings>

<add key ="ValidationSettings:UnobtrusiveValidationMode" value="None" /> <!--Xử lí lỗi login-->

</appSettings>

Chỉ khi đăng nhập thành công (LoginNv) thì mới mở được trang quản lý: Managements.aspx

**LonginNV**.DestinationPageUrl = ~/FrManagements.aspx **Ngoài ra, có thể cá nhân hóa bằng cách thiết kế lại (thiết kế bổ sung)**

Nhờ vào **LoginNV**[Task]->Convert to Template

## Lập trình bổ sung Masterpage quản lý

**MÃ LỆNH LẬP TRÌNH: Coding (MastePage.Master):**

**+ Viết thủ tục : Ẩn / Hiện LoginNV {trên MasterPage.Master}**

/// <summary>

/// Thủ tục tự chạy khi mở trên web browser

/// ẩn login nv

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

protected void Page\_Load(object sender, EventArgs e)

{

LoginNv.Visible = false;

}

/// <summary>

/// Viết thủ tục ẩn hiện login (trên MasterPage.master)

/// </summary>

/// <param name="sender"></param>

/// <param name="e">lưu trữ thông tin mục chọn trên menumain mà nhân viên đãchọn</param>

protected void MenuMain\_MenuItemClick(object sender, MenuEventArgs e)

{

if(e.Item.Value == "ql" || e.Item.Value == "dn") // KHINSD(NV) chọn quản lý hoặc đăng nhập

{

LoginNv.Visible = true; // mở công cụ loginnv để nv đăng nhập để thực hiện các chức năng quản lý hệ thống

}

else // khi nv chọn các mục khách

{

LoginNv.Visible = false;

}

}

**Khi dùng công cụ Login (.NET 2015 về trước) có thể có lỗi** sau (*liên quan bảo mật*):

“WebForms UnobtrusiveValidationMode requires a ScriptResourceMapping for 'jquery'.   
 Please add a ScriptResourceMapping named jquery(case-sensitive).”

**Để khắc phục lỗi này, trong file Web.config** chúng ta thêm đoạn mã lệnh sau:

<configuration>

. . .

<appSettings>

<add key="ValidationSettings:UnobtrusiveValidationMode" value="None" />   
 <!--XỬ LÝ LỖI: khi dùng LOGINNV--> </appSettings>

. . .

</configuration>

Chú giải: Code trên chỉ là tượng trưng; trong mở rộng và phát triển sản phẩm đề tài sẽ:

1. Tổ chức Username và Password: lưu trữ và kiểm tra từ một Table trong DB.

2. Xử lý bảo mật khi NSD nhớ (copy) Link url : như trình bày trong Chương 2: trang 11

**+ Viết thủ tục xứ lý Đăng nhập cho LoginNV** {trên MasterPage.Master}

CÓ 2 PHƯƠNG ÁN XỬ LÝ ĐĂNG NHẬP:   
2 phương án KHÔNG được sử dụng đồng thời, nếu dùng đồng thời thì phương án 1 **KHÔNG** có tác dụng.

**Phương án 1:** Sử dụng thư viện ASP.NET có sẵn [Login.Task]Administer Website

Phải mua bản quyền và cài đặt bổ sung

**Phương án 2:** Code thủ công: Event: Authenticate của Login

/// <summary>

/// //XỬ LÝ ĐĂNG NHẬP

/// </summary>

/// <param name="sender"></param>

/// <param name="e">Ghi nhận việc đăng nhập có thành công hay không</param>

protected void LoginNV\_Authenticate(object sender, AuthenticateEventArgs e)

{ //Nếu NSD đăng nhập với tên Admin và mật khẩu Admin@vn Hoặc ....

if ( (LoginNV.UserName == "Admin" && LoginNV.Password == "admin@vn")||

(LoginNV.UserName == "Guest" && LoginNV.Password == "guest@vn") )

{

e.Authenticated = true;// Đăng nhập thành công, tự động chuyển đến trang trong   
 LoginNV.DestinationPageURL đã gán(tức là trang Quản lý FrManagements)

Session["dn"] = LoginNV.UserName; //GỬI username sang trang quản lý

}

else// Đăng nhập ko thành công

{

e.Authenticated = false;

}

}//Authenticate của LoginNV

Ghi chú:

1 Code trên chỉ là tượng trưng;   
 thực tế là Username và Password : lưu trữ và kiểm tra từ một Table trong DB

2. Bảo mật khi NSD nhớ (copy) Link url

## Trang chính của hệ thống quản lý hệ thống bán vé online

Thừa kế MasterPage.master nêu trên, dùng Table để bố cục trang và ít nhất phải thực hiện các yêu cầu sau:

\*Quản lý có nhiều chức năng; như: Quản lý vé, Quản lý điểm bán vé, Quản lý nhân viên, Quản lý kho và Tồn kho, Khách hàng và đơn hàng, …. Tìm kiếm, thống kê, báo cáo tổng hợp …. Kết xuất ra bản in (Report…), vì vậy thiết kế trang quản lý chính như sau (chỉ khi đăng nhập thành công thì mới vào được trang này, từ đây chọn các chức năng cần thiết để sang các trang quản lý cụ thể khác).

## Thiết kế (Design) trang quản lý chung (chính): Managements.aspx



## Lập trình (Codes) trang quản lý chung (chính): Mannagements.aspx

static string us = null;

/// <summary>

/// TỬ CHẠY KHI TRANG QUẢN LÝ này ĐƯỢC MỞ

/// LỜI chào Người NV đăng nhập thành công

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

protected void Page\_Load(object sender, EventArgs e)

{

if (Session["dn"] != null) us = Session["dn"] as string; //Nếu đã có tên Đăng nhập (thành công) thì gán tên vào biến toàn cục us

lbTitle.Text = "XIN CHÀO :" + us + ": ĐÂY LÀ TRANG QUẢN LÝ CHÍNH: TỪ ĐÂY CHON CÁC CHỨC NĂNG QUẢN LÝ CỤ THỂ KHÁC";

}

//GỌI WebPage QUẢN LÝ HÀNG HÓA

/// <summary>

/// Gọi trang quản lý vé

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

protected void BtnVe\_Click(object sender, EventArgs e)

{

Response.Redirect("~\\FrTicketManagements.aspx");

us = Session["dn"] as string; //gởi tên đăng nhập sang trang quản lý vé: FrTicketManagements.aspx thông qua Session

}

## Trang quản lý danh mục quản lý vé: FrTicketManagements.aspx

## Phân tích trang Quản lý vé

+ Dành cho nhân viên sau khi đăng nhập thành công và chọn chức năng quản lý vé từ trang quản lý chung.

+ Liên quan 2 Tables sau:

***DanhMucVe và DanhMucLoaiVe***

+ Chủ yếu: thực hiện các chức năng: Thêm \_ Sửa \_ Xóa…..trong DanhMucLoaiVe

[1] New WebForm: FrTicketsManagements.aspx

[2] Từ trang Quản lý chung (FrManagements.aspx) gọi trang Quản lý vé (FrTicketsManagements.aspx)

[3] Design

- Dùng DropdownList: liệt kê và cho phép NV QL chọn nhóm mặt hàng cần quản lý (có mục chọn "tất cả")

- Dùng GridView: Liệt kê danh sách các mặt hàng thuộc nhóm đã chọn trong DropDownList   
 (sử dụng phân trang Paging cho GridView)

+ Chuỗi kết nối DB: 1G322DinhVanThanhVeOnlineConnectionString1

<connectionStrings>

workstation id=1G322DinhVanThanhVeOnline.mssql.somee.com;packet size=4096;

user id=gthreedvthanh\_SQLLogin\_1;

pwd=hzp3ww338k;

data source=1G322DinhVanThanhVeOnline.mssql.somee.com;

persist security info=False;

initial catalog=1G322DinhVanThanhVeOnline

## Thiết kế (Design) trang quản lý danh mục vé: FrTicketsManagements.aspx

* Thêm nút lệnh / FrManagement : Quản lý chung “Quản lý danh mục vé”:

protected void BtnVe\_Click(object sender, EventArgs e)

{

Response.Redirect("~\\FrTicketsManagements.aspx");

us = Session["dn"] as string; //gởi tên đăng nhập sang trang quản lý vé: FrTicketManagements.aspx thông qua Session

}

* DropDownListNhomMH: DataSourceID = SqlDataSourceNhom

. Câu lệnh kết nối DB: SELECT \* FROM [DanhMucLoaiVe]  
. Các thiết lập thuộc tính :

+ Enable Auto Postback

+ Append Data Bound Items = true

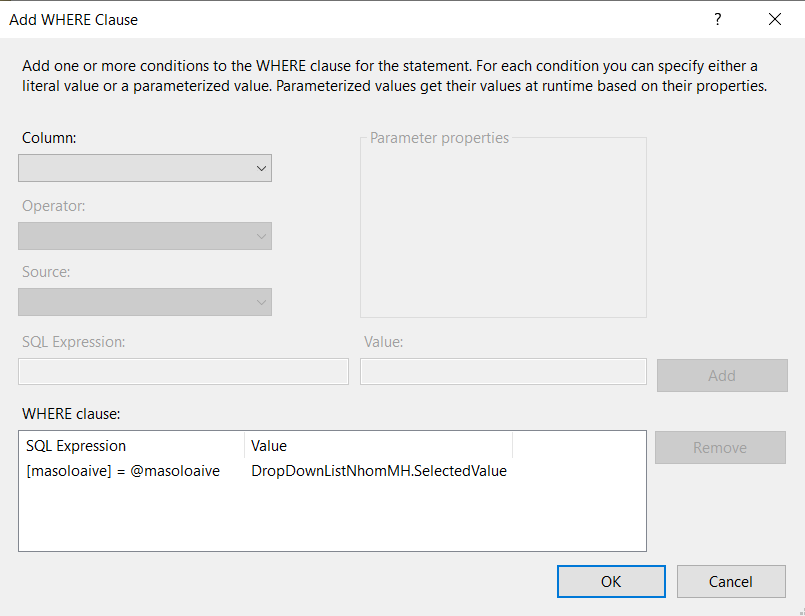
+ DataTextField = tenloaive

+ DataValueField = masoloaive

* GridViewMH: DataSourceID = SqlDataSourceMH

. Câu lệnh SQL kết nối DataBase:  
 SELECT \* FROM [DanhMucVe] WHERE ([masoloaive] = @masoloaive)

@masoloaive = DropDownListNhom. SelectedValue



. Task->Edit Columns: HeaderText, DataField (Tự động gán cột dữ liệu tương ứng)

**Đối với thuộc tính Hình ảnh**

+ Convert this field to a TemplateField (góc dưới phải của Properties của Field)

+ Đối với các Field là Template (Field tự thiết kế, không dùng mặc định): PHẢI THIẾT KẾ

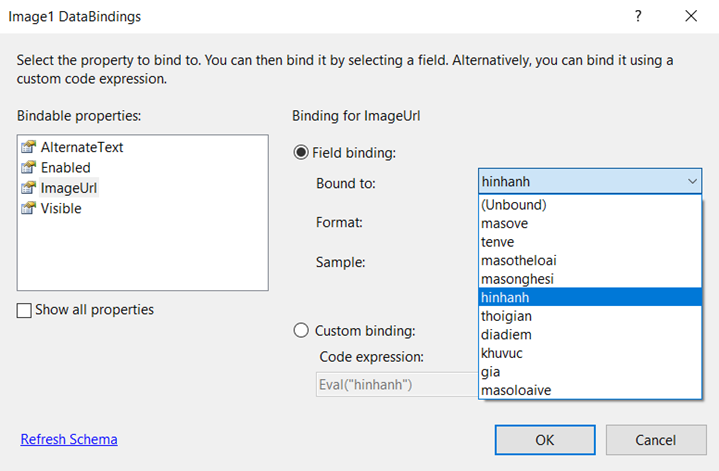
[Gridview: Task] -> Edit Template (chỉ làm khi ít nhất có 1 field Template trong Gridview)

*Chú giải: . Có thể có nhiều cột TemplateFields: phải chọn từng cột để thiết kế*

*. Một cột có thể phải thiết kế nhiều trạng thái: ItemTeplate : mặc định,…*

Field: “Hinh”: Xóa Label = đổi => Image [Task]->Edit DataBinding

Imageurl = Eval("hinhanh")



+ Ẩn / Hiện cột “Hình ảnh” trên GridView thông qua CheckBox (người sử dụng chọn)

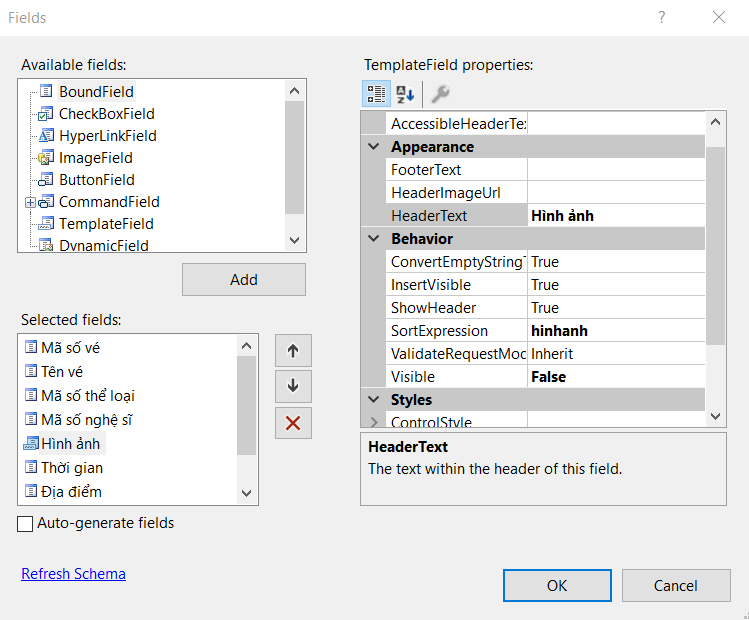
Checkbox trên trang Quản lý vé

.Text = Hiển thị Hình ảnh

.Autopostback = true

+ Bắt đầu không với cột “Hình ảnh”.Visible = false

Trạng thái ban đầu GridviewMH: cột “Hình ảnh” ẩn



.Event cho CheckBoxHinh

protected void CheckBoxHinh\_CheckedChanged(object sender, EventArgs e)

{

GridViewMH.Columns[4].Visible = !GridViewMH.Columns[4].Visible;

// ẩn hiện hình mẫu bằng cột 4, nếu thêm/bỏ cột phải điểu chỉnh lại vị trí cột

}

* CheckBoxHinh: Cho người sử dụng chọn có/không hiển thị hình ảnh của danh sách vé trong GridView (Mặc định “không” = vì tải hình ảnh sẽ làm chậm trang web):

. Text = Hiển thị hình ảnh

. Checked = False (ban đầu “không” chọn)

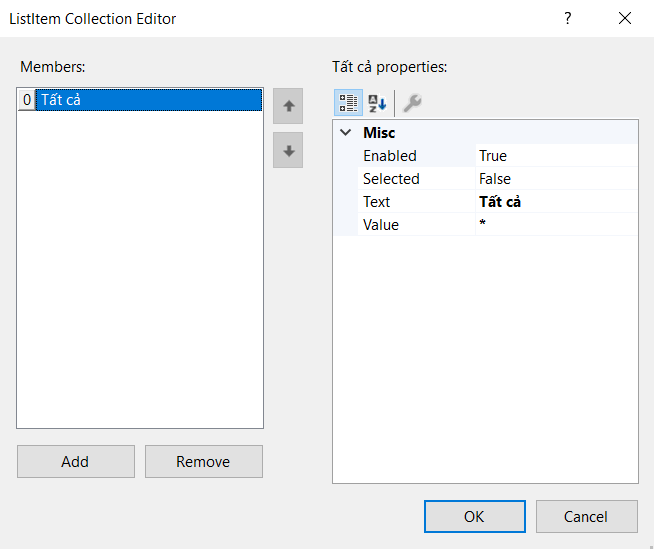
* Bổ sung mục chọn **“Tất cả”** trong DropDownList và hiển thị **masoloaive** kế bên

B1: Bổ sung Item mới=Dropdownlist[Task…]->Edit Item: Add [.Text = Tất cả, .Value = \*]

Bổ sung mục chọn “Tất cả”:

Trong Dropdownlist: [Task] -> Edit Items

Properties .Items: thêm “Tất cả” có Value = “\*”



B2: Properties[Dropdownlist] AppendDataBoundItems = True: các mục chọn trong *DDL* sẽ bao gồm các mục Edit Items ở B1 ghép (Append) với các mục chọn có trong DB được kết vào DDL này (NhomMH).

.AppendDataBoundItem = true

*Mặc định thuộc tính này = false*

*“Các mục chọn trong Item nêu trên [Tất cả] là ưu tiên và sẽ KHÔNG nối các mục chọn từ DB vào”*

B3: Viết bổ sung câu lệnh trong thủ tục sự kiện SelectedIndexChanged của DropDownListNhom

/// <summary>

/// THỦ TỤC TỰ ĐỘNG CHẠY MỖI KHI NSD (NV) THAY ĐỔI LỰA CHỌN TRONG Dropdownlist này

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

protected void DropDownListNhomMH\_SelectedIndexChanged(object sender, EventArgs e)

{

if (DropDownListNhomMH.SelectedValue.Trim() == "\*")// NSD (NV) đã chọn mục tất cả (Text) với Value = \*

{

SqlDataSourceMH.SelectCommand = "SELECT \* FROM [DanhMucVe]";//Viết lại câu lệnh Select ... From : không Where.. = liệt kê All

}

lbmsnhom.Text = DropDownListNhomMH.SelectedValue;

//Hiểm thị mã nhóm món ăn ở kế bên = tự động chay trên Page\_Load (nêu trên)

}

Ngoài ra, copy codes trên bổ sung vào Page\_Load(…) để khi WebPage vừa được tải lên đúng lúc trong Dropdownlist là “Tất cả|” (vì nó là mục đầu tiên) thì GridView bên đưới sẽ liệt kê tả cả các MH:

protected void Page\_Load(object sender, EventArgs e)

{

DropDownListNhomMH\_SelectedIndexChanged(sender, e);//Gọi thủ tục sự kiện ..SelectIndexChange của Dropdownlist => xử lý chọn \* và ...

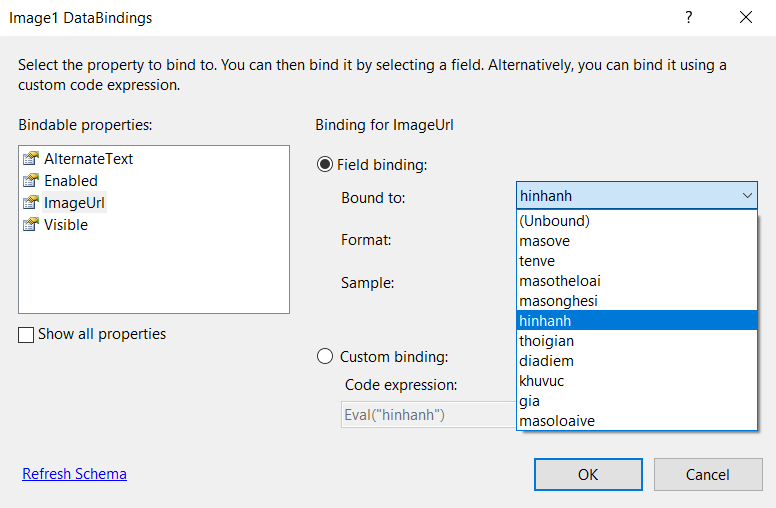
}

* **Checkbox** cho hiển thị / không hiển thị Hình ảnh của …. Trong GridView

B1: Chuyển đổi cột “Hình ảnh” GridView sang chế độ Tự thiết kế: Template Field

[GridView]Task->Edit Columns: “Hình ảnh” -> Chọn “Convert to Template Field”

B2: Thiết kế lại cột hình ảnh trong GridView: [GridView]Task->Edite Templates



Hình ‑. Thiết kế lại cột "Hình ảnh" trên GridView

Thay thế Label bởi Image (lấy từ Toolbox):   
 Image..Task->Edit DataBinding: ImageURL = hinhanh (trong DB)

B3: Codes thay đổi lựa chọn của CheckBox (.Auto Postback = True)

//THAY ĐỔI LỰA CHỌN : HIỂN || HAY KHÔNG HIỂN THỊ HÌNH ẢNH CỦA .... TRONG GRIDVIEW

protected void CheckBoxHinh\_CheckedChanged(object sender, EventArgs e)

{

GridViewMH.Columns[5].Visible = !GridViewMH.Columns[5].Visible;

}

CHÚ GIẢI:

.PostBack = True khi chỉ cần 1 lựa chọn (DropDownList || CheckBox) là thực hiện tải lại DB

.PostBack = False khi chỉ cần Nhiều lựa chọn (DropDownList, CheckBox, . .) thì mới tải lại DB

* **Paging**: Phân trang cho GridView

## Lập trình (Codes) trang quản lý danh mục vé: FrTicketsManagements.aspx

protected void Page\_Load(object sender, EventArgs e)

{

DropDownListNhomMH\_SelectedIndexChanged(sender, e);//Gọi thủ tục sự kiện ..SelectIndexChange của Dropdownlist => xử lý chọn \* và ...

}

## Lập trình (Codes) THÊM \_ SỬA \_ XÓA trang quản lý danh mục vé: FrTicketsManagements.aspx

[1] Các thao tác dữ liệu: Thêm \_ xóa \_ sửa \_ chọn được thực hiện ngày trên dòng dữ liệu trên GridView trên Gridview

[2] Với các thao tác dữ liệu cụ thể cần ghi chú:

**SỬA thông tin hoặc dữ liệu của vé**

+ Không sửa “masoloaive” mà CHUYỂN nhóm (chuyển vé từ loại này sang loại khác), làm bằng trang khác [không thực hiện trên trang Quản lý vé này]

+ Không sửa “hinhanh” vì trong DB lưu đường dẫn đến file hình. Sử dụng chức năng riêng CẬP NHẬT HÌNH bằng cách chọn hình mới thay thế hình cũ

+ Không sửa masove (vì là thuộc tính khóa chính) và ghi chú một vấn đề tương tự “Thêm mới”

**XÓA thông tin của vé**

+ Khi Xóa sẽ liên quan RBTV dữ liệu (xử lý trong phần sau). Trong phần này chưa xử lý RBTV (xóa sẽ báo lỗi: RBTV …….Reference Constraint….: ví dụ: Xóa vé đang còn tồn kho: chưa xóa được, sẽ có báo lỗi như trên)

**THÊM thông tin hoặc dữ liệu của vé mới**

+ masoloaive không nhập mà sẽ là masoloaive đã chọn trong Dropdownlist (NhomMH) phía trên. Vì vậy khi chọn “Tất cả” trong Dropdownlist (mã nhóm \*) thì phải ẩn nút “Thêm vé mới” (không cho thêm mới trong trường hợp này; cần có ToolTip để nhắc NSD chọn loại vé cụ thể trước khi thêm vé mới)

+ Có nhiều phương án tổ chức thêm vé mới, ví dụ:

Phương án 1: Khi NSD (NV) bấm nút “Thêm vé mới”: thì sẽ thêm 1 dòng trống mới trong Gridview để NSD (NV) nhập thông tin vé mới; có 3 phương án cụ thể: dòng đầu trong gridview, dòng cuối, tại vị trí đang chọn (thường là dòng cuối): trường hợp này, một số Field (hình mẫu: chọn file hình, kiểu date: mở lịch để chọn, …) phải chuyển các này thành TemplateField  
Phương án 2: (Trong bài làm này tạm dùng) Dùng các Textbox để nhập thông tin vé mới:   
. Chỉ khi NSD (NV) bấm “Thêm vé mới” thì các Textbox hiện ra cho NSD (NV) nhập thông tin vé mới vào, đổi nhãn “Thêm vé mới” thành “Lưu”;   
. Sau khi NSD (NV) nhập xong thông tin MH mới, sẽ bấm nút “Lưu”: lưu vé mới vào BD đồng thời ẩn các TextBox và đồi nhãn nhút “Lưu” => “Thêm vé mới”

+ masove : là khóa chính, không nên cho tự nhập (hệ thống tự tăng lên theo qui luật nào đó; không nên tăng dần vì giá trị mã càng ngày càng lớn và không liên tục do xóa: nên dùng phương án khác, VD: năm + quý + nhóm vé + thứ tự trong năm 21 4 03 123) tuy nhiên tạm thời bài làm này ta cho NSD (NV) tự nhập nhưng phải xử lý trùng khóa chính

+ Phải quan tâm một số RBTV khi nhập: phải chuyển các Field này thành TemplateField

VD:   
. giá >= 0 (=0 : khuyến mãi): dùng Validator (tạm thời chưa làm)  
. hình mẫu: chọn file hình mẫu từ máy tính cá nhân của NSD [NV] (Client PC) ->Upload lên Host: copy file hình (không nhập trực tiếp, vì là path lưu trong DB): dùng Tool FileUpload / ToolBox.

[3] Ghi chú thêm: Paging (phân trang)

\* CÁC BƯỚC THỰC HIỆN

Bước 1: Thiết kế nút lệnh [Thêm | Xóa || Sửa || Chọn] trên các dòng của Gridview

Thực hiện trên GridView MH

+ Gridview[Task]->Edit column

+ Available fields: CommandField: Add = Edit (Sửa) Delete (Xóa), Select (Chọn)

. Quan tâm thêm việc: “chọn” có thể chuyển TemplateField thành Checkbox: cho phép chọn nhiều để thực hiện thao tác nào đó, VD để xóa,…

B1: Thiết kế nút lệnh [Thêm | Xóa || Sửa || Chọn] trên các dòng của Gridview

## Thêm vé

Table: Insert Row below

Button ID = btnThem .Text = Thêm món ăn mới .Enable = False

Bước 1: **Không** cho phép “Thêm Mặt hàng mới” khi NSD chọn “**Tất cả**” trong **DropDownListNhomMH**. Vì khi thêm mới, được hiểu là thêm mới vào “nhóm loại vé” đã chọn trong **DropDownListNhomMH**

masoloaive ko nhập mà sẽ masoloaive đã chọn trong Dropdownlist. Vì vậy khi chọn “Tất cả” trong Dropdownlist (mã nhóm \*) thì phải ẩn nút “Thêm mới” (không cho thêm mới trong trường hợp này; cần có ToolTip để nhắc NSD chọn nhóm MH cụ thể trước khi thêm)

protected void DropDownListNhomMH\_SelectedIndexChanged(object sender, EventArgs e)

{

if (DropDownListNhomMH.SelectedValue.Trim() == "\*")// NSD (NV) đã chọn mục tất cả (Text) với Value = \*

{

SqlDataSourceMH.SelectCommand = "SELECT \* FROM [DanhMucVe]";//Viết lại câu lệnh Select ... From : không Where.. = liệt kê All

btnThem.Enabled = false; //nsd chọn mục tất cả, không chọn thêm

btnThem.ToolTip = DropDownListNhomMH.ToolTip = "Phải chọn nhóm cụ thể để có thể thêm Vé mới vào Nhóm đó";//thông báo nhắc người dùng

}

else

{

btnThem.Enabled = true; //thêm mới được vì đã chọn nhóm cụ thể

btnThem.ToolTip = DropDownListNhomMH.ToolTip = "";//không thông báo nhắc người dùng

}

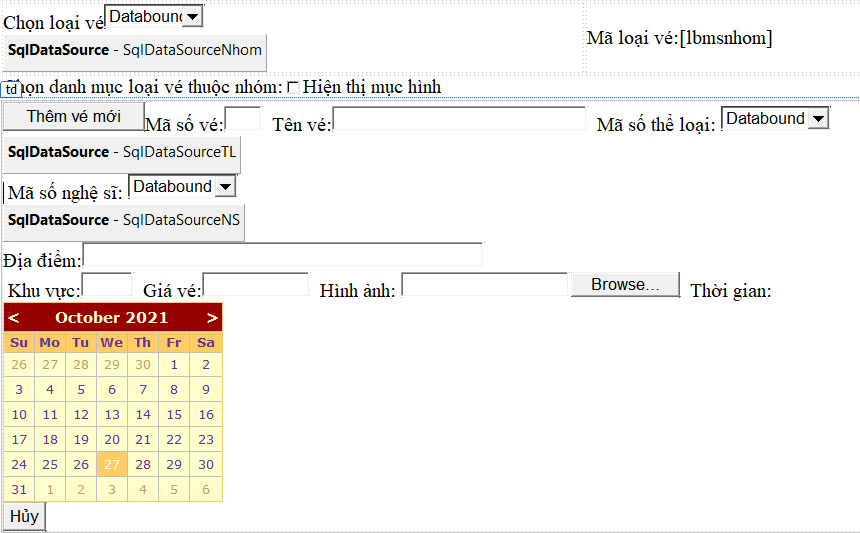
lbmsnhom.Text = DropDownListNhomMH.SelectedValue;

//Hiểm thị mã nhóm món ăn ở kế bên = tự động chay trên Page\_Load (nêu trên)

}

Bước 2: Thiết kế các textbox cho phép NSD thêm vé mới

Sử dụng phương án 2 đã nêu trên: Trong bài làm này tạm dùng các Textbox để nhập thông tin MH mới:   
+ Chỉ khi NSD bấm “thêm vé mới” thì các Textbox hiện ra cho nsd nhập thông tin MH mới, đổi nhãn “Thêm vé mới” thành “Lưu”;   
+ NSD sau khi nhập xong thông tin vé mới, sẽ bấm nút “Lưu” : lưu vé mới vào BD, đồng thời ẩn các TextBox và đồi nhãn nhút “Lưu” thành “Thêm vé mới”



Bước 3: Viết lệnh SQL thêm vé mới trrong **SqlDataSourceMH** tương ứng

INSERT INTO DanhMucVe(masove, tenve, masotheloai, masonghesi, hinhanh, thoigian, diadiem, khuvuc, gia, masoloaive)

VALUES (@masove, @tenve, @masotheloai, @masonghesi, @hinhanh, @thoigian, @diadiem, @khuvuc, @gia, @masoloaive)

KHAI BÁO THAM SỐ Parameters cho Insert nêu trên: VỚI CÁC GIÁ TRỊ THAM SỐ @... ĐƯỢC LẤY TỪ CÁC TEXTBOX THIẾT KẾ TRONG Bước 2

HOẶC KHAI BÁO THAM SỐ BÊN Views CODES: CHỈ ĐƯỢC DÙNG 1 KHÔNG DÙNG CẢ 2

Trong phần mềm sẽ dùng Codes vì (“~\\Media\\...”) khi tải lên hình ảnh

**Bước 4:** Viết mã lệnh (Codes) cho nút lênh “Thêm vé mới”

/// <summary>

/// VIẾT MÃ LỆNH CHO NÚT THÊM / LƯU

/// PA2: Trong bài làm này tạm dùng các Textbox để nhập thông tin MH mới:

/// + chỉ khi nsd bấm “thêm mới” thì các Textbox hiện ra cho nsd nhập thông tin MH mới, đổi nhãn “Thêm mới” => “Lưu”;

/// + nsd sau khi nhập xong thông tin MH mới, sẽ bấm nút “Lưu” : lưu MH mới vào BD đồng thời ẩn các TextBox và đồi nhãn nhút “Lưu” => “Thêm mới”

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

protected void btnThem\_Click(object sender, EventArgs e)

{//B1: ẨN || HIỆN CÁC Ô NHẬP DỮ LIỆU : TEXTBOX

lbmasove.Visible = !lbmasove.Visible;

lbtenve.Visible = !lbtenve.Visible;

lbtheloai.Visible = !lbtheloai.Visible;

lbnghesi.Visible = !lbnghesi.Visible;

lbhinhanh.Visible = !lbhinhanh.Visible;

lbthoigian.Visible = !lbthoigian.Visible;

lbdiadiem.Visible = !lbdiadiem.Visible;

lbkhuvuc.Visible = !lbkhuvuc.Visible;

lbgia.Visible = !lbgia.Visible;

txtmasove.Visible = !txtmasove.Visible;

txttenve.Visible = !txttenve.Visible;

DropDownListTL.Visible = !DropDownListTL.Visible;

DropDownListNS.Visible = !DropDownListNS.Visible;

FileUploadhinhanh.Visible = !FileUploadhinhanh.Visible;

Calendarthoigian.Visible = !Calendarthoigian.Visible;

txtdiadiem.Visible = !txtdiadiem.Visible;

txtkhuvuc.Visible = !txtkhuvuc.Visible;

txtgia.Visible = !txtgia.Visible;

btnHuy.Visible = !btnHuy.Visible;

//B2: ĐỔI NHÃN nút "Thêm..." <=> "Lưu..."

if(btnThem.Text == "Thêm vé mới")//"Thêm...Lưu"

{

//Xóa trống tất cả các TextBox để chuẩn bị cho nsd (NV) nhập thông tin mặt hàng mới

txtmasove.Text = "";

txtmasove.ToolTip = "Không để trống mã vé";

txttenve.Text = "";

txtdiadiem.Text = "";

txtkhuvuc.Text = "";

txtgia.Text = "";

txtgia.ToolTip = "Giá > 0";

//đổi nhãn nút "thêm ..." => "lưu..."

btnThem.Text = "Lưu vé mới";

}

else //"Lưu...Thêm"

{

try

{

SqlDataSourceMH.InsertParameters.Clear();//Xóa ALL tham số cũ;chuẩn bị Định nghĩa tham số mơi

SqlDataSourceMH.InsertParameters.Add("masove", txtmasove.Text.Trim());

SqlDataSourceMH.InsertParameters.Add("tenve", txttenve.Text);

SqlDataSourceMH.InsertParameters.Add("masotheloai", DropDownListTL.SelectedValue.Trim());

SqlDataSourceMH.InsertParameters.Add("masonghesi", DropDownListNS.SelectedValue.Trim());

SqlDataSourceMH.InsertParameters.Add("hinhanh", "~\\Media\\" + FileUploadhinhanh.FileName.Trim());

SqlDataSourceMH.InsertParameters.Add("thoigian",System.Data.DbType.DateTime, Calendarthoigian.SelectedDate.ToShortDateString());

SqlDataSourceMH.InsertParameters.Add("diadiem", txtdiadiem.Text.Trim());

SqlDataSourceMH.InsertParameters.Add("khuvuc", txtkhuvuc.Text.Trim());

SqlDataSourceMH.InsertParameters.Add("gia", System.Data.DbType.Double,txtgia.Text.Trim());

SqlDataSourceMH.InsertParameters.Add("masoloaive", DropDownListNhomMH.SelectedValue.Trim());

SqlDataSourceMH.Insert(); //LƯU MH MỚI VÀO DB = CÂU LỆNH QUAN TRỌNG 1

//COPY HÌNH VỪA CHON TỪ FileUpload vào THƯ MỤC =~\\Media\\

//System.IO.File.Copy(FileUploadhinhanh.FileName, "D:\\G322DinhVanThanhVeOnline\\Media\\"+FileUploadhinhanh.FileName,true); //chồng lên đã có file

this.Title = "THÊM THÀNH CÔNG !";

}

catch(System.Exception ex)

{

this.Title = "CÓ LỖI THÊM VÉ MỚI !" + ex.Message;

}

btnThem.Text = "Thêm vé mới";

//TẢI DỮ LIỆU SAU KHI THÊM LÊN GridviewMH

GridViewMH.DataBind();

}

} //btnThem\_Click

**Bước 5:** Viết mã lệnh (Codes) cho nút lệnh “Hủy”

protected void btnHuy\_Click(object sender, EventArgs e)

{ //B1: ẨN CÁC Ô NHẬP DỮ LIỆU : TEXTBOX

lbmasove.Visible = false;

lbtenve.Visible = false;

lbtheloai.Visible = false;

lbnghesi.Visible = false;

lbhinhanh.Visible = false;

lbthoigian.Visible = false;

lbdiadiem.Visible = false;

lbkhuvuc.Visible = false;

lbgia.Visible = false;

txtmasove.Visible = false;

txttenve.Visible = false;

DropDownListTL.Visible = false;

DropDownListNS.Visible = false;

FileUploadhinhanh.Visible = false;

Calendarthoigian.Visible = false;

txtdiadiem.Visible = false;

txtkhuvuc.Visible = false;

txtgia.Visible = false;

btnHuy.Visible = false;

//đổi nhãn thành nút "thêm ..."

btnThem.Text = "Thêm vé mới";

}//btnHuy

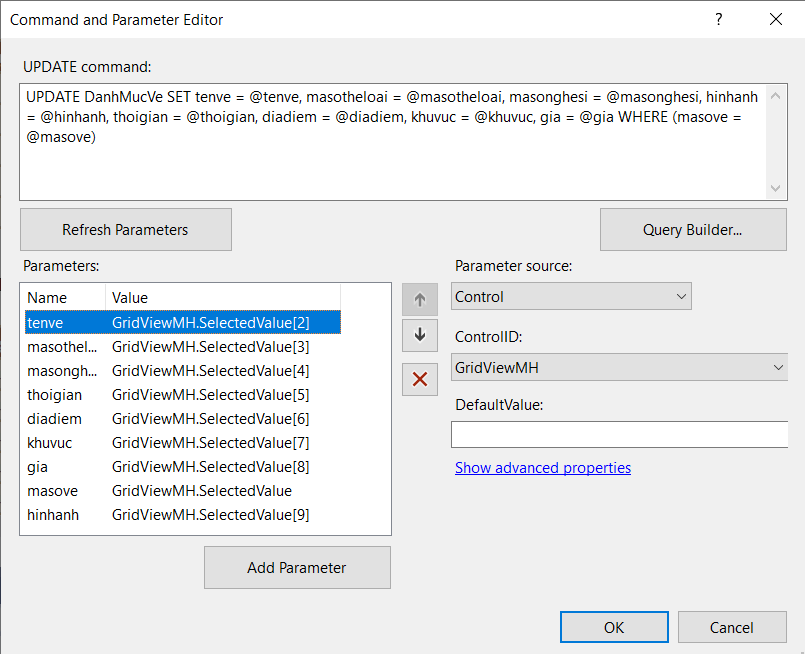
## Sửa thông tin vé

Viết lệnh SQL Sửa (Update) thông tin vé trong **SqlDataSourceMH** tương ứng

UPDATE DanhMucVe SET tenve = @tenve, masotheloai = @masotheloai, masonghesi = @masonghesi, thoigian = @thoigian, diadiem = @diadiem, khuvuc = @khuvuc, gia = @gia

WHERE (masove = @masove)

**KHAI BÁO THAM SỐ**



+ KHÔNG SỬA masove vì đây là thuộc tính khóa chính (Primary Key)

+ Nhóm vé, Thể loại, Nghệ sĩ: Chuyển nhóm và dùng WebForm riêng [.ReadOnly = True]

**Xử lý Template Fields: Quản lý danh mục vé**

B1: Điều chỉnh DB

Kiểm tra cột kiểu Datetime[thoigian] và các FK [masotheloai], FK[masonghesi] trong DB/Design/MS. SQL + nhập dữ liệu + Modify Diagram

B2: Refresh Schema: Tải lại DB từ SQL (nếu có modify)

B3: GridViewMH.Task-Edit Columns: [..Datetime: thoigian] => Convert to template field..

B4: GridViewMH.Task-Edit Template: [Datetime: thoigian] =>

. EditItem Template: Calendard =>*~~TextBox~~*: DataBinding

B5: Update Command: dbType

*SQL Float, Double => dbType: Double*

*Date => dbType: Date (DateTime)*

*Char: không thay đổi => dbType mặc định ( Object)*

**Sửa [Update] Hình: Template => UploadFile [Copy file và Update DB]**

B1: Viết lại câu lệnh SQL Update / SqlDataSourceMH {sửa thêm field “hình ảnh”}

UPDATE DanhMucVe SET tenve = @tenve, masotheloai = @masotheloai, masonghesi = @masonghesi, thoigian = @thoigian, diadiem = @diadiem, khuvuc = @khuvuc, gia = @gia

WHERE (masove = @masove)

KHAI BÁO THAM SỐ

hinh = GridViewMH.SelectedValue[9]

B2: DESIGN: GridViewMH{task..}->Edit Template: Columns[9]: Hình mẫu

EditItemTemplate:

+TextBox1: có sẵn (đã binding DB <-field [hinh]/DB: two-way..)

.Enable = False (không cho sửa vì là path của file)

+ FileUpload1 : bổ sung

+ btnHinh: Bổ sung (NSD (NV) chọn hình sau khi đã lấy từ FileUpload (nêu trên) lên

.CommandName: select

.Text = Chọn

B3:Viết thủ tục: **RowEditing** của **GridViewMH**

static int pos = 0; // biến toàn cục static lưu vị trí đang sửa edit

/// <summary>

/// LƯU GIỮ VỊ TRÍ {lưu trong biến toàn cục id [static] DÒNG đang Update["hinhanh"] trên GridViewMH

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

protected void GridViewMH\_RowEditing(object sender, GridViewEditEventArgs e)

{

pos = e.NewEditIndex; //e là tham số để lấy vị trí dong đang Editing

}

B4**:** Viết thủ tục: **RowCommand** của **GridViewMH**

Gán hình mà NSD (NVQL) vừa chọn từ FileUpload lên thay thế vào hình cũ / TextBox1   
 (đã DataBinding đến dữ liệu / DataBase)

*Thực hiện tương tự ItemCommand của DataList / Homepage*

protected void GridViewMH\_RowCommand(object sender, GridViewCommandEventArgs e)

{

if(e.CommandName.Trim() == "select")

{

string p = ((FileUpload)GridViewMH.Rows[pos].FindControl("FileUpload1")).FileName; //

if (p != "")//nsd [nv] đã chọn file hình trong FileUpload [filename]

{

//GÁN HÌNH VỪA CHỌN trong FileUpload vào textbox dataBinding

((TextBox)GridViewMH.Rows[pos].FindControl("TextBox1")).Text = "~\\Media\\" + p;

// copy file hình vừa chọn vào thư mục "~\\Media\\" của app

((FileUpload)GridViewMH.Rows[pos].FindControl("FileUpload1")).SaveAs("D:\\G322DinhVanThanhVeOnline\\Media\\" + p);

//chú ý: khi di chuyển app này đến một thư mục khác hoặc upload lên somee ->phải thay đổi path vật lý

//

}

((FileUpload)GridViewMH.Rows[pos].FindControl("FileUpload1")).Visible = false; //ẨN FileUpload sau khi gán xong

((Button)GridViewMH.Rows[pos].FindControl("btnHinh")).Text = "Chọn hình khác"; //đổi nhãn nút thành: "Chọn hình khác"

}//if

}

## Xóa vé

B2: Viết lệnh SQL Xóa (Delete) vé trong **SqlDataSourceMH** tương ứng

DELETE FROM DanhMucVe WHERE (masove = @masove)

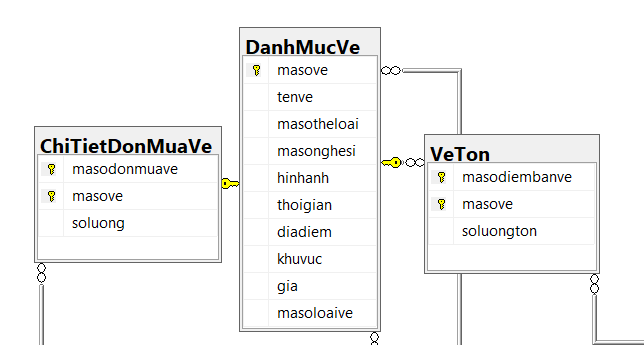
Khai báo tham số:

masove = GridViewMH.SelectedValue

## Lập trình (Codes) xử lý RBTV dữ liệu khi Xóa danh mục vé: FrTicketsManagements.aspx

### Giới thiệu

Khi xóa vé (Table DanhMucVe) thì sẽ bị ảnh hưởng RBTV dữ liệu đối với 2 bảng: CTDDH (Chi tiết đơn hàng) và TONKHO (Tôn kho), như Hình 5‑4



Hình ‑. Trích DB liên quan xử lý RBTFV dữ liệu khi Xóa MH

Có 2 trường hợp xử lý RBTV dữ liệu:

+ Vé cần xóa đang còn Tồn kho: xử lý “Thanh lý” = Xóa Vé tồn của vé đó: Xóa từng vé tồn OR xóa tất cả vé tồn của vé đó.

+ Vé cần xóa đang còn đơn đặt hàng chưa giải quyết (ChiTietDonMuaVe): xử lý đổi hàng (Xóa ChiTietDonMuaVe đó và chọn mua vé khác tương đương) HOẶC thuyết phục khách hàng từ chối mua vé đó = Xóa ChiTietDonMuaVe: Xóa từng ChiTietDonMuaVe OR xóa tất cả ChiTietDonMuaVe của vé đó.

**Một số lý thuyết về RBTV DL khi xóa MH**

\* Khi Xóa vé sẽ ảnh hưởng đến các tables: ChiTietDonMuaVe và VeTon của vé đó :   
\* Có 2 trường hợp xử lý: *Ghi chú: ngay từ đầu Ẩn cột “Xóa”/GridViewMH khi 2 GridView RBTV có DL (chưa xoá đc):* GridviewMH[Task]->Edit Columns: Cột “Xóa”.Visible = False

ChiTietDonMuaVe

- Chuyển ChiTietDonMuaVe đó sang vé khác bằng cách thuyết phục khách hàng chọn mua vé khác tương đương.

VD: ChiTietDonMuaVe

Mã đơn Mã vé Số Lượng

01 01 5 + 3 = 8

*01 ~~02~~-sửa->01 7->3*

. . .

VD: Đang cần xóa vé có mã là 02

GHI CHÚ: Nếu vé chuyển sang đã có sẵn trong DonMuaVe rồi thì chỉ cộng tăng thêm ssố lượng và xóa record hiện tại

=> Dùng cách khác:

Xóa cũ, mua mới (tóm lại: “Xóa” => thêm mới vào đơn): KHÔNG “Sửa”

Hủy các ChiTietDonMuaVe của vé đó bằng cách thuyết phục khách hàng không mua vé đó nữa

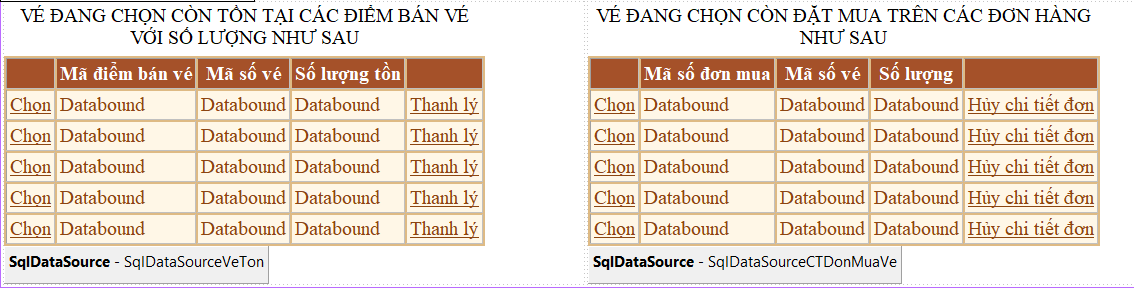
VeTon

- Thanh lý (“Xóa”) các VeTon của vé cần xóa

TÓM LẠI: ChiTietDonMuaVe và VeTon: xử lý “xóa” ALL , từng record..

### Thiết kế 2 GridView phụ ngay trên Form quản lý FrGoodsManagements.aspx

**\* Thiết kế 2 gridviews phụ** như Hình 5‑5



Hình ‑. Các Gridviews xử lý RBTV dữ liệu khi Xóa

**\* Các thuộc tính**

**GridViewVeTon**

. Caption = VÉ ĐANG CHỌN CÒN TỒN TẠI CÁC ĐIỂM BÁN VÉ VỚI SỐ LƯỢNG NHƯ SAU

**GridViewCTDonMuaVe**

. Caption = VÉ ĐANG CHỌN CÒN ĐẶT MUA TRÊN CÁC ĐƠN HÀNG NHƯ SAU

**\* Các câu lệnh Select**

**SqlDataSourceVeTon**

SELECT \* FROM [VeTon] WHERE ([masove] = @masove)

Với @masove = GridViewMH.SelectedValue

**SqlDataSourceCTDonMuaVe**

SELECT \* FROM [ChiTietDonMuaVe] WHERE ([masove] = @masove)

Với @masove = GridViewMH.SelectedValue

### Xử lý các RBTV trên 2 GridView phụ

**\* GridViewVeTon**

**Thanh lý[Delete]**:Chính là câu lệnh Delete được viết trong **SqlDataSourceVeTon**

DELETE FROM VeTon WHERE (masodiembanve = @masodiembanve) AND (masove = @masove)

Với masodiembanve = GridViewVeTon.SelectedValue[1]

masove = GridViewVeTon. SelectedValue[2]

**\* GridViewCTDonMuaVe**

**Khách hàng từ chối[Delete]**: Chính là câu lệnh Delete được viết trong **SqlDataSourceCTDDH**

DELETE FROM ChiTietDonMuaVe WHERE (masodonmuave = @masodonmuave) AND (masove = @masove)

Với masodonmuave = GridViewCTDonMuaVe.SelectedValue[1]

masove = GridViewCTDonMuaVe.SelectedValue[2]

*===ĐỔI HÀNG = xóa CTDDH & Mua mới*

Đổi hàng[Update]: Đổi hàng = xóa CTDonMuaVe và Mua mới

**GridViewCTDonMuaVe** [Task]->Edit Columns..:

Mã đơn hàng.ReadOnly = True

Mã hàng.ReadOnly = False (được phép đổi hàng)

Số lượng.ReadOnly = False (được phép thay đổi số lượng)

Viết câu lệnh Update … trong **SqlDataSourceCTDonMuaVe**

UPDATE ChiTietDonMuaVe SET masove = @masove\_new, soluong = @soluong WHERE (masove = @masove) AND (masodonmuave = @masodonmuave)

Với masodonmuave = GridViewCTDonMuaVe.SelectedValue[1]

masove = GridViewCTDonMuaVe.SelectedValue[2]

soluong = GridViewCTDonMuaVe.SelectedValue[3]

## Trang quản lý nhân sự: FrHumanResource.aspx

.Thêm nút lệnh / FrManagement : Quản lý chung “Quản lý nhân viên”:

protected void btnNv\_Click(object sender, EventArgs e)

{

Response.Redirect("~\\FrHumanResource.aspx");

}

.Thiết kế (Design)

. Thêm DropdownListKV [“Tất cả”[\*] & AppendDataBoundItem=True & AutoPostback=True]

. SqlDataSourceNhanVien SELECT \* FROM [NhanVien] WHERE (([masokuvuc] = @masokhuvuc)

. Khai báo tham số: mskv = DropdownListKV.SelectValue

Xử lý chọn: “Tất cả”

public partial class FrHumanResource : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

DropDownListKV\_SelectedIndexChanged(sender, e); //gọi thủ tục

}

protected void DropDownListKV\_SelectedIndexChanged(object sender, EventArgs e)

{

btnThemNV.Enabled = true;

btnThemNV.ToolTip = DropDownListKV.ToolTip = ""; //đã chọn cụ thể nên kh nhắc

if(DropDownListKV.SelectedValue.ToString() == "\*")

{

btnThemNV.Enabled = false;

btnThemNV.ToolTip = DropDownListKV.ToolTip = "Chọn một khu vực cụ thể";

SqlDataSourceNhanVien.SelectCommand = "SELECT \* FROM [NhanVien]";

}

lbmskv.Text = DropDownListKV.SelectedValue;

}

Xóa nhân viên:

DELETE FROM NhanVien WHERE (masonhanvien = @masonhanvien)

Sửa thông tin nhân viên:

UPDATE NhanVien SET tennhanvien = @tennhanvien, masocm = @masocm, masonhanvien = WHERE (masonhanvien = @masonhanvien)

Thêm nhân viên mới:

INSERT INTO NhanVien(masonhanvien, tennhanvien, masocm, masokhuvuc) VALUES (@masonhanvien, @tennhanvien, @masocm, @masokhuvuc)

**FULL CODES**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace G322DinhVanThanhVeOnline

{

public partial class FrHumanResource : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

DropDownListKV\_SelectedIndexChanged(sender, e); //gọi thủ tục

}

protected void DropDownListKV\_SelectedIndexChanged(object sender, EventArgs e)

{

btnThemNV.Enabled = true;

btnThemNV.ToolTip = DropDownListKV.ToolTip = ""; //đã chọn cụ thể nên kh nhắc

if(DropDownListKV.SelectedValue.ToString() == "\*")

{

btnThemNV.Enabled = false;

btnThemNV.ToolTip = DropDownListKV.ToolTip = "Chọn một khu vực cụ thể";

SqlDataSourceNhanVien.SelectCommand = "SELECT \* FROM [NhanVien]";

}

lbmskv.Text = DropDownListKV.SelectedValue;

}

protected void btnThemNV\_Click(object sender, EventArgs e)

{

lbmanv.Visible = !lbmanv.Visible;

lbhotennv.Visible = !lbhotennv.Visible;

lbmacm.Visible = !lbmacm.Visible;

txtmasocm.Visible = !txtmasocm.Visible;

txtmasonhanvien.Visible = !txtmasonhanvien.Visible;

txttennhanvien.Visible = !txttennhanvien.Visible;

btnHuyThem.Visible = !btnHuyThem.Visible;

if(btnThemNV.Text == "Thêm nhân viên")

{

txtmasonhanvien.Text = ""; txtmasonhanvien.ToolTip = "";

txttennhanvien.Text = "";

txtmasocm.Text = "";

btnThemNV.Text = "Lưu thông tin nhân viên";

}

else

{

try

{

SqlDataSourceNhanVien.Insert();

GridViewNhanVien.DataBind();

this.Title = "Thêm nhân viên thành công";

}

catch (System.Exception ex) { this.Title = "Lỗi thêm nhân viên mới " + ex.Message; };

btnThemNV.Text = "Thêm nhân viên";

}

}

protected void btnHuyThem\_Click(object sender, EventArgs e)

{

lbmanv.Visible = false;

lbhotennv.Visible = false;

lbmacm.Visible = false;

txtmasocm.Visible = false;

txtmasonhanvien.Visible = false;

txttennhanvien.Visible = false;

btnHuyThem.Visible = false;

btnThemNV.Text = "Thêm nhân viên";

}

}

}

## Trang quản lý điểm bán vé (FrBookingOffice.aspx)

.Thêm nút lệnh / FrManagement : Quản lý chung “Quản lý điểm bán vé”:

protected void btnDbv\_Click(object sender, EventArgs e)

{

Response.Redirect("~\\FrBookingOffice.aspx");

}

.Thiết kế (Design)

. Thêm DropdownListKV [“Tất cả”[\*] & AppendDataBoundItem=True & AutoPostback=True]

. SqlDataSourceDBV SELECT \* FROM [DiemBanVe] WHERE (([masokuvuc] = @masokhuvuc)

. Khai báo tham số: mskv = DropdownListKV.SelectValue

Xử lý chọn: “Tất cả”

protected void Page\_Load(object sender, EventArgs e)

{

DropDownListKV\_SelectedIndexChanged(sender, e);

}

protected void DropDownListKV\_SelectedIndexChanged(object sender, EventArgs e)

{

btnThemDBV.Enabled = true;

btnThemDBV.ToolTip = DropDownListKV.ToolTip = "";

if (DropDownListKV.SelectedValue.ToString() == "\*")

{

btnThemDBV.Enabled = false;

btnThemDBV.ToolTip = DropDownListKV.ToolTip = "Chọn một khu vực cụ thể";

SqlDataSourceDBV.SelectCommand = "SELECT \* FROM [DiemBanVe]";

}

lbmskv.Text = DropDownListKV.SelectedValue;

}

.Lệnh SQL:

Xóa điểm bán vé:

DELETE FROM DiemBanVe WHERE (masodiembanve = @masodiembanve)

Sửa thông tin điểm bán vé:

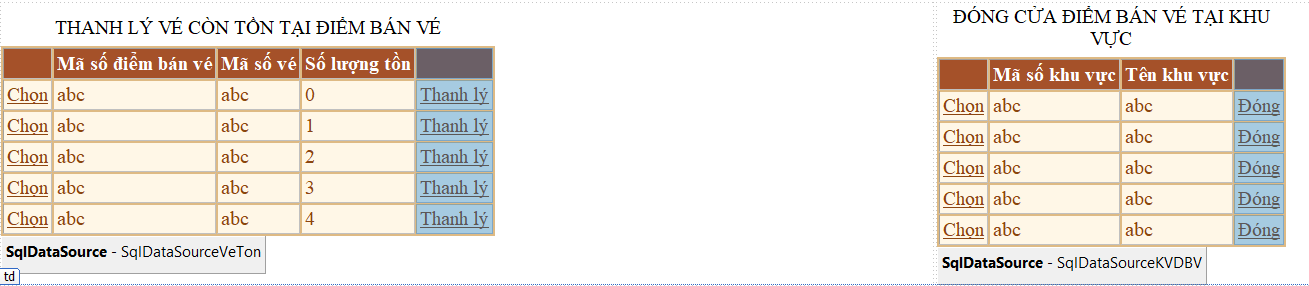
UPDATE DiemBanVe SET tendiembanve = @tendiembanve, diachi = @diachi WHERE (masodiembanve = @masodiembanve)

Thêm điểm bán vé mới:

INSERT INTO DiemBanVe(masodiembanve, tendiembanve, diachi, soluong, masokhuvuc) VALUES (@masodiembanve, @tendiembanve, @diachi, @soluong, @masokhuvuc)

. Xử lý RBTV khi “Xóa”

Thêm 2 GridView như sau:



GridViewVeTon

* SELECT \* FROM [VeTon] WHERE ([masodiembanve] = @masodiembanve)
* DELETE FROM VeTon WHERE (masodiembanve = @masodiembanve) AND (masove = @masove)

GridViewKhuVuc

* SELECT \* FROM [KhuVuc] WHERE ([masokhuvuc] = @masokhuvuc)
* DELETE FROM KhuVuc WHERE (masokhuvuc = @masokhuvuc)

**FULL CODES**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace G322DinhVanThanhVeOnline

{

public partial class FrBookingOffice : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

DropDownListKV\_SelectedIndexChanged(sender, e);

}

protected void DropDownListKV\_SelectedIndexChanged(object sender, EventArgs e)

{

btnThemDBV.Enabled = true;

btnThemDBV.ToolTip = DropDownListKV.ToolTip = "";

if (DropDownListKV.SelectedValue.ToString() == "\*")

{

btnThemDBV.Enabled = false;

btnThemDBV.ToolTip = DropDownListKV.ToolTip = "Chọn một khu vực cụ thể";

SqlDataSourceDBV.SelectCommand = "SELECT \* FROM [DiemBanVe]";

}

lbmskv.Text = DropDownListKV.SelectedValue;

}

protected void btnThemDBV\_Click(object sender, EventArgs e)

{

lbmsdbv.Visible = !lbmsdbv.Visible;

lbtendbv.Visible = !lbtendbv.Visible;

lbdiachi.Visible = !lbdiachi.Visible;

lbsoluong.Visible = !lbsoluong.Visible;

txtmsdbv.Visible = !txtmsdbv.Visible;

txttendbv.Visible = !txttendbv.Visible;

txtdiachi.Visible = !txtdiachi.Visible;

txtsoluong.Visible = !txtsoluong.Visible;

btnHuythem.Visible = !btnHuythem.Visible;

if( btnThemDBV.Text == "Thêm điểm bán vé")

{

txtmsdbv.Text = ""; txtmsdbv.ToolTip = "";

txttendbv.Text = "";

txtdiachi.Text = "";

txtsoluong.Text = "";

btnThemDBV.Text = "Lưu thông tin";

}

else

{

try

{

SqlDataSourceDBV.Insert();

GridViewDBV.DataBind();

this.Title = "Thêm điểm bán vé thành công";

}

catch(System.Exception ex)

{

this.Title = "Lỗi thêm điểm bán vé" + ex.Message;

};

btnThemDBV.Text = "Thêm điểm bán vé";

}

}

protected void btnHuythem\_Click(object sender, EventArgs e)

{

lbmsdbv.Visible = false;

lbtendbv.Visible = false;

lbdiachi.Visible = false;

lbsoluong.Visible = false;

txtmsdbv.Visible = false;

txttendbv.Visible = false;

txtdiachi.Visible = false;

txtsoluong.Visible = false;

btnThemDBV.Text = "Thêm điểm bán vé";

}

}

}

## Trang quản lý thể loại âm nhạc (FrMusicGenre.aspx)

Thêm nút lệnh / FrManagement : Quản lý chung “Quản lý thể loại âm nhạc”:

protected void btnTl\_Click(object sender, EventArgs e)

{

Response.Redirect("~\\FrMusicGenre.aspx");

}

DropDownListTL: DataSourceID = SqlDataSourceTheLoai

. Câu lệnh kết nối DB: SELECT \* FROM [DanhMucTheLoai]  
. Các thiết lập thuộc tính :

+ Enable Auto Postback

+ Append Data Bound Items = true

+ DataTextField = tennghesi

+ DataValueField = masonghesi

GridViewTL: DataSourceID = SqlDataSourceTL

. Câu lệnh SQL kết nối DataBase:  
SELECT \* FROM [DanhMucTheLoai] WHERE ([masotheloai] = @masotheloai)

@masotheloai = DropDownListTL.SelectedValue

. Các câu lệnh SQL Thêm, Xóa, Sửa, Chọn:

SELECT \* FROM [DanhMucTheLoai] WHERE ([masotheloai] = @masotheloai)

DELETE FROM DanhMucTheLoai WHERE (masotheloai = @masotheloai)

UPDATE DanhMucTheLoai SET tentheloai = @tentheloai WHERE (masotheloai = @masotheloai)

INSERT INTO DanhMucTheLoai(masotheloai, tentheloai) VALUES (@masotheloai, @tentheloai)

. Cho phép xem danh sách vé thuộc loại vé đã chọn và cập nhật thể loại:

+ Thêm GridViewVe

+ Các câu lệnh SQL:

* SELECT \* FROM [DanhMucVe] WHERE ([masotheloai] = @masotheloai)
* UPDATE DanhMucVe SET masotheloai = @masotheloai WHERE (masove = @masove)

**FULL CODES**

public partial class FrMusicGenre : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

DropDownListTL\_SelectedIndexChanged(sender, e);

}

protected void DropDownListTL\_SelectedIndexChanged(object sender, EventArgs e)

{

btnThem.Enabled = true;

if (DropDownListTL.SelectedValue.ToString() == "\*")

{

SqlDataSourceTL.SelectCommand = "SELECT \* FROM [DanhMucTheLoai]";

}

lbmstl.Text = DropDownListTL.SelectedValue;

}

protected void btnThem\_Click(object sender, EventArgs e)

{

lbmasotheloai.Visible = !lbmasotheloai.Visible;

lbtentheloai.Visible = !lbtentheloai.Visible;

txtmasotheloai.Visible = !txtmasotheloai.Visible;

txttentheloai.Visible = !txttentheloai.Visible;

btnHuyThem.Visible = !btnHuyThem.Visible;

if (btnThem.Text == "Thêm thể loại")

{

txtmasotheloai.Text = "";

txttentheloai.Text = "";

btnThem.Text = "Lưu thông tin";

}

else

{

try

{

SqlDataSourceTL.Insert();

GridViewTL.DataBind();

this.Title = "Thêm thành công";

}

catch (System.Exception ex)

{

this.Title = "Lỗi thêm điểm bán vé" + ex.Message;

};

btnThem.Text = "Thêm thể loại";

}

}

protected void btnHuyThem\_Click(object sender, EventArgs e)

{

lbmasotheloai.Visible = false;

lbtentheloai.Visible = false;

txtmasotheloai.Visible = false;

txttentheloai.Visible = false;

btnThem.Text = "Thêm thể loại";

}

}

## Trang quản lý thể loại nghệ sĩ biểu diễn (FrArtist.aspx)

Thêm nút lệnh / FrManagement : Quản lý chung “Quản lý nghệ sĩ biểu diễn”:

protected void btnNs\_Click(object sender, EventArgs e)

{

Response.Redirect("~\\FrArtist.aspx");

}

DropDownListNS: DataSourceID = SqlDataSourceNgheSi

. Câu lệnh kết nối DB: SELECT \* FROM [DanhMucNgheSi]  
. Các thiết lập thuộc tính :

+ Enable Auto Postback

+ Append Data Bound Items = true

+ DataTextField = tennghesi

+ DataValueField = masonghesi

GridViewNS: DataSourceID = SqlDataSourceNS

. Câu lệnh SQL kết nối DataBase:  
 SELECT \* FROM [DanhMucNgheSi] WHERE ([masonghesi] = @masonghesi)

@masonghesi = DropDownListNS. SelectedValue

. Các câu lệnh SQL Thêm, Xóa, Sửa, Chọn:

UPDATE DanhMucNgheSi SET tennghesi = @tennghesi WHERE (masonghesi = @masonghesi)

SELECT \* FROM [DanhMucNgheSi] WHERE ([masonghesi] = @masonghesi)

INSERT INTO DanhMucNgheSi(masonghesi, tennghesi) VALUES (@masonghesi, @tennghesi)

DELETE FROM DanhMucNgheSi WHERE (masonghesi = @masonghesi)

. Xử lý khi hủy buổi biểu diễn của nghệ sĩ

+ Thêm các GridView như sau:



+ GridViewVe

* Caption: DANH SÁCH VÉ THEO NGHỆ SĨ
* SELECT \* FROM [DanhMucVe] WHERE ([masonghesi] = @masonghesi)
* DELETE FROM DanhMucVe WHERE (masove = @masove)
* UPDATE DanhMucVe SET masonghesi = @masonghesi WHERE (masove = @masove)

+ GridViewVeMua

* Caption: VÉ ĐANG CHỌN CÒN ĐANG ĐẶT MUA NHƯ SAU:
* SELECT \* FROM [ChiTietDonMuaVe] WHERE ([masove] = @masove)
* DELETE FROM ChiTietDonMuaVe WHERE (masodonmuave = @masodonmuave) AND (masove = @masove)

+ GridViewVeTon

* Caption: VÉ ĐANG CHỌN CÒN ĐANG TỒN NHƯ SAU:
* SELECT \* FROM [VeTon] WHERE ([masove] = @masove)
* DELETE FROM VeTon WHERE (masodiembanve = @masodiembanve) AND (masove = @masove)

## Trang quản lý khu vực (FrArea.aspx)

Thêm nút lệnh / FrManagement : Quản lý chung “Quản lý khu vực”:

protected void btnKv\_Click(object sender, EventArgs e)

{

Response.Redirect("~\\FrArea.aspx");

}

DropDownListLoaiVe: DataSourceID = SqlDataSourceKV

. Câu lệnh kết nối DB: SELECT \* FROM [KhuVuc]  
. Các thiết lập thuộc tính :

+ Enable Auto Postback

+ Append Data Bound Items = true

+ DataTextField = tenkhuvuc

+ DataValueField = masokhuvuc

GridViewKhuVuc: DataSourceID = SqlDataSourceKhuVuc

. Câu lệnh SQL kết nối DataBase cho GridViewKhuVuc:  
 SELECT \* FROM [KhuVuc] WHERE ([masokhuvuc] = @masokhuvuc)

@masokhuvuc = DropDownListKV. SelectedValue

. Các câu lệnh SQL Thêm, Xóa, Sửa, Chọn:

SELECT \* FROM [KhuVuc] WHERE ([masokhuvuc] = @masokhuvuc)

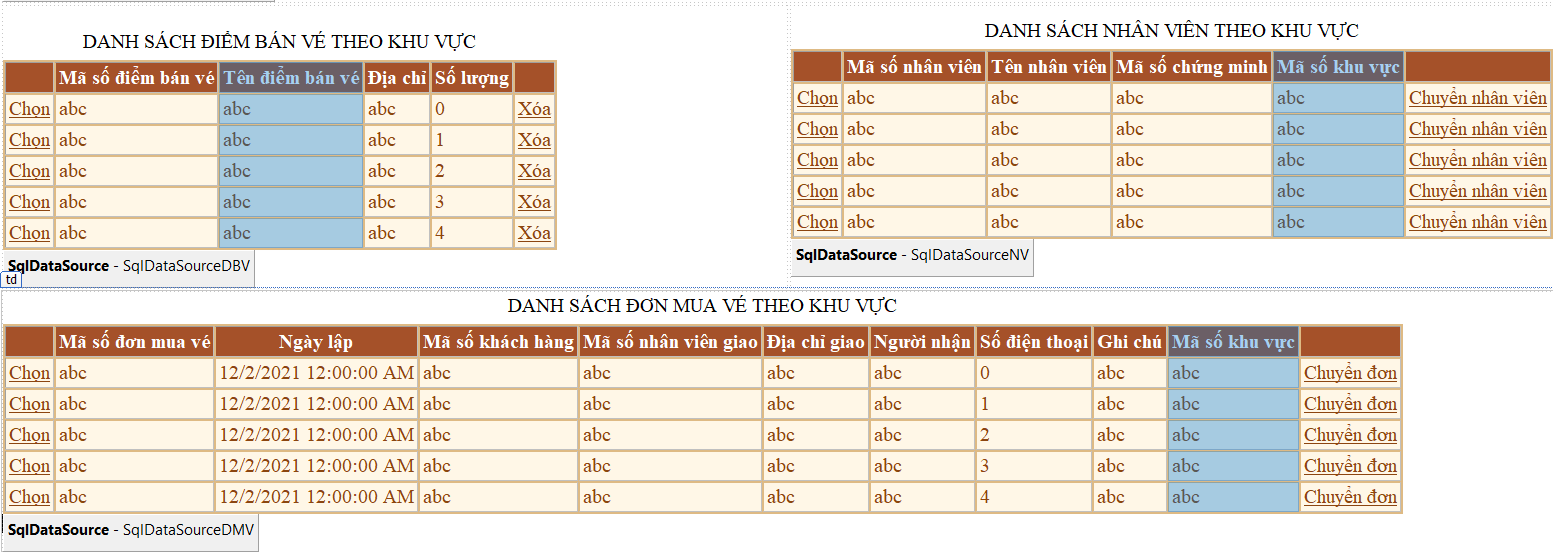
INSERT INTO KhuVuc(masokhuvuc, tenkhuvuc) VALUES (@masokhuvuc, @tenkhuvuc)

UPDATE KhuVuc SET tenkhuvuc = @tenkhuvuc WHERE (masokhuvuc = @masokhuvuc)

DELETE FROM KhuVuc WHERE (masokhuvuc = @masokhuvuc) (Xóa khu vực đồng nghĩa với việc ngừng kinh doanh ở khu vực đó)

. Xử lí RBTV khi xóa khu vực:

+ Thiết kế các GridView như sau:



+ GridViewDBV

* Caption: DANH SÁCH ĐIỂM BÁN VÉ THEO KHU VỰC
* SELECT \* FROM [DiemBanVe] WHERE ([masokhuvuc] = @masokhuvuc)
* DELETE FROM DiemBanVe WHERE (masokhuvuc = @masokhuvuc)

+ GridViewNV: Ngừng bán hàng ở khu vực, không xóa nhân viên ở khu vực đó mà chuyển sang khu vực khác

* Caption: DANH SÁCH NHÂN VIÊN THEO KHU VỰC
* SELECT \* FROM [NhanVien] WHERE ([masokhuvuc] = @masokhuvuc)
* UPDATE NhanVien SET masokhuvuc = @masokhuvuc WHERE (masonhanvien = @masonhanvien)

+ GridViewDMV: Ngừng bán hàng ở khu vực, các đơn hàng trong khu vực đó chuyển sang khu vực khác, không xóa đơn mua

* Caption: DANH SÁCH ĐƠN MUA VÉ THEO KHU VỰC
* SELECT \* FROM [DonMuaVe] WHERE ([masokhuvuc] = @masokhuvuc)
* UPDATE DonMuaVe SET masokhuvuc = @masokhuvuc WHERE (masodonmuave = @masodonmuave)

## Trang quản lý loại vé (FrTicketClass.aspx)

Thêm nút lệnh / FrManagement : Quản lý chung “Quản lý khu vực”:

protected void btnNhomve\_Click(object sender, EventArgs e)

{

Response.Redirect("~\\FrTicketClass.aspx");

}

DropDownListLoaiVe: DataSourceID = SqlDataSourceLoaiVe

. Câu lệnh kết nối DB: SELECT \* FROM [DanhMucLoaiVe]  
. Các thiết lập thuộc tính :

+ Enable Auto Postback

+ Append Data Bound Items = true

+ DataTextField = tenloaive

+ DataValueField = masoloaive

GridViewLV: DataSourceID = SqlDataSourceLV

. Câu lệnh SQL kết nối DataBase:  
 SELECT \* FROM [DanhMucVe] WHERE ([masoloaive] = @masoloaive)

@masoloaive = DropDownListNhom. SelectedValue

. Các câu lệnh SQL Thêm, Xóa, Sửa, Chọn:

UPDATE DanhMucLoaiVe SET tenloaive = @tenloaive, mota = @mota WHERE (masoloaive = @masoloaive)

SELECT \* FROM [DanhMucLoaiVe] WHERE ([masoloaive] = @masoloaive)

INSERT INTO DanhMucLoaiVe(masoloaive, tenloaive, mota) VALUES (@masoloaive, @tenloaive, @mota)

DELETE FROM DanhMucLoaiVe WHERE (masoloaive = @masoloaive)

. Cho phép xem danh sách vé thuộc loại vé đã chọn và cập nhật loại vé:

+ Thêm GridViewVe

+ Các câu lệnh SQL:

* SELECT \* FROM [DanhMucVe] WHERE ([masoloaive] = @masoloaive)
* UPDATE DanhMucVe SET masoloaive = @masoloaive WHERE (masove = @masove)

**FULL CODES**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace G322DinhVanThanhVeOnline

{

public partial class FrTicketClass : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

DropDownListLoaiVe\_SelectedIndexChanged(sender, e);

}

protected void DropDownListLoaiVe\_SelectedIndexChanged(object sender, EventArgs e)

{

btnThemLV.Enabled = true;

if (DropDownListLoaiVe.SelectedValue.ToString() == "\*")

{

SqlDataSourceLV.SelectCommand = "SELECT \* FROM [DanhMucLoaiVe]";

}

lbmslv.Text = DropDownListLoaiVe.SelectedValue;

}

protected void btnThemDBV\_Click(object sender, EventArgs e)

{

lbmsloaive.Visible = !lbmsloaive.Visible;

lbtenlv.Visible = !lbtenlv.Visible;

lbmota.Visible = !lbmota.Visible;

txtmsloaive.Visible = !txtmsloaive.Visible;

txttenlv.Visible = !txttenlv.Visible;

txtmota.Visible = !txtmota.Visible;

btnHuythem.Visible = !btnHuythem.Visible;

if (btnThemLV.Text == "Thêm loại vé")

{

txtmsloaive.Text = ""; txtmsloaive.ToolTip = "Nhập mã loại vé không được trùng";

txttenlv.Text = "";

txtmota.Text = "";

btnThemLV.Text = "Lưu thông tin";

}

else

{

try

{

SqlDataSourceLV.Insert();

GridViewLV.DataBind();

this.Title = "Thêm thành công";

}

catch (System.Exception ex)

{

this.Title = "Lỗi thêm điểm bán vé" + ex.Message;

};

btnThemLV.Text = "Thêm loại vé";

}

}

protected void btnHuythem\_Click(object sender, EventArgs e)

{

lbmsloaive.Visible = false;

lbtenlv.Visible = false;

lbmota.Visible = false;

txtmsloaive.Visible = false;

txttenlv.Visible = false;

txtmota.Visible = false;

btnHuythem.Visible = false;

btnThemLV.Text = "Thêm loại vé";

}

}

}

## Trang quản lý nhập xuất vé (FrStock.aspx)

Thêm nút lệnh / FrManagement : Quản lý chung “Quản lý vé tồn”:

protected void btnTon\_Click(object sender, EventArgs e)

{

Response.Redirect("~\\FrStock.aspx");

}

DropDownListDiemBanVe: DataSourceID = SqlDataSourceDiemBanVe

. Câu lệnh kết nối DB: SELECT \* FROM [DiemBanVe]  
. Các thiết lập thuộc tính :

+ Enable Auto Postback

+ Append Data Bound Items = true

+ DataTextField = tenloaive

+ DataValueField = masoloaive

GridViewVeTon: DataSourceID = SqlDataSourceVeTon

. Câu lệnh SQL kết nối DataBase:  
 SELECT \* FROM [VeTon] WHERE ([masodiembanve] = @masodiembanve)

@masodiembanve = DropDownListNhom. SelectedValue

. Các câu lệnh SQL Thêm, Xóa, Sửa, Chọn:

Xóa 1 tồn kho thì chính là xuất hết mặt hàng đó ra khỏi kho

DELETE FROM VeTon WHERE (masodiembanve = @masodiembanve) OR (masove = @masove)

Thêm mới một tồn kho đồng nghĩa với việc nhập kho mặt hàng mới hoặc nhập kho mặt hàng đã hết

INSERT INTO VeTon(masodiembanve, masove, soluongton) VALUES (@masodiembanve, @masove, @soluongton)

Sửa chữa thông tin một dữ liệu tồn kho thì có 2 trường hợp: Một là nếu sửa số lượng tăng lên thì tương đương với nhập thêm vé về điểm bán vé; hai là số lượng giảm xuống tức là xuất 1 loại vé ra khỏi điểm bán vé nhưng không xuất hết (không bằng không, không âm); sửa (tăng giảm slton; sửa masodiemban vé tức chuyển vé từ điểm bán vé này sang điểm bán vé khác; không sửa masove)

UPDATE VeTon SET masodiembanve = @masodiembanve, soluongton = @soluongton WHERE (masove = @masove)

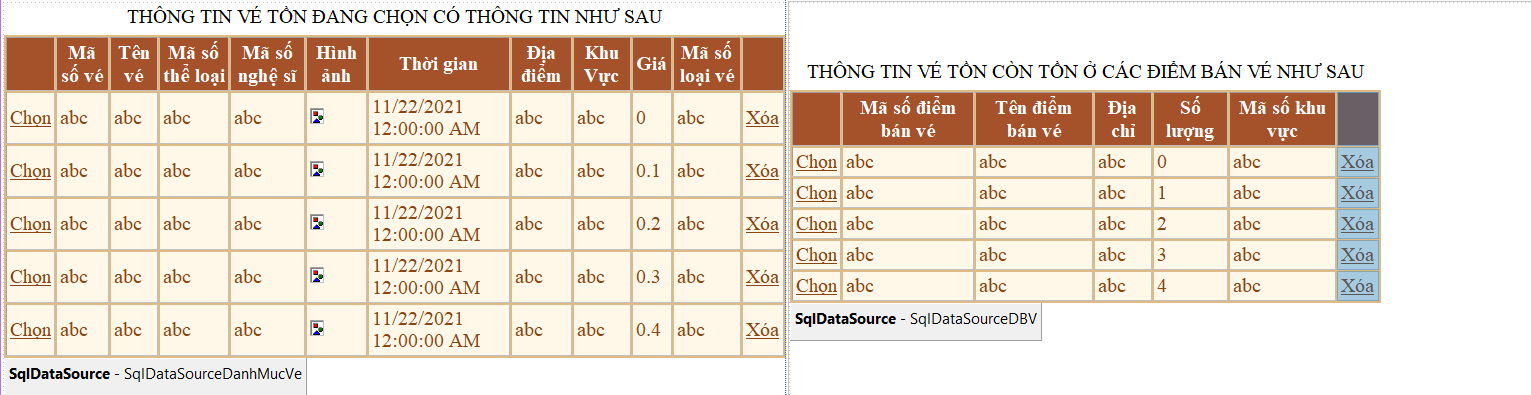
SELECT \* FROM [VeTon] WHERE ([masodiembanve] = @masodiembanve)

. Các chú thích cho phần Thêm và Xóa (Xử lý ràng buộc)

+ Khi thêm mới vào vé tồn người dùng phải tạo mới thông tin trên hai bảng DanhMucVe (trang FrTicketsManagements.aspx) và DiemBanVe (trang FrBookingOffice.aspx). Số lượng tồn được xem là số lượng mới nhập vào.

. Khi xóa cần xử lí ràng buộc

+ Thiết kế 2 GridView sau:



+ GridViewDBV: Thực hiện thanh lý vé tồn tại điểm bán vé

* Caption: THÔNG TIN VÉ TỒN CÒN TỒN Ở CÁC ĐIỂM BÁN VÉ NHƯ SAU
* SELECT \* FROM [DanhMucVe] WHERE ([masove] = @masove)
* DELETE FROM DanhMucVe WHERE (masove = @masove)

+ GridViewDanhMucVe: Xóa thông tin vé vì đã thanh lý xong

* Caption: THÔNG TIN VÉ TỒN ĐANG CHỌN CÓ THÔNG TIN NHƯ SAU
* SELECT \* FROM [DiemBanVe] WHERE ([masodiembanve] = @masodiembanve)
* DELETE FROM DiemBanVe WHERE (masodiembanve = @masodiembanve)

**FULL CODES**

public partial class FrStock : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

DropDownListDiemBanVe\_SelectedIndexChanged(sender, e);

}

protected void DropDownListDiemBanVe\_SelectedIndexChanged(object sender, EventArgs e)

{

btnThem.Enabled = true;

if (DropDownListDiemBanVe.SelectedValue.ToString() == "\*")// NSD (NV) đã chọn mục tất cả (Text) với Value = \*

{

SqlDataSourceVeTon.SelectCommand = "SELECT \* FROM [VeTon]";//Viết lại câu lệnh Select ... From : không Where.. = liệt kê All

btnThem.Enabled = false; //nsd chọn mục tất cả, không chọn thêm

btnThem.ToolTip = DropDownListDiemBanVe.ToolTip = "Phải chọn một điểm bán vé cụ thể";//thông báo nhắc người dùng

}

lbmsdbv.Text = DropDownListDiemBanVe.SelectedValue;

//Hiểm thị mã nhóm vé ở kế bên = tự động chay trên Page\_Load (nêu trên)

}

protected void btnThem\_Click(object sender, EventArgs e)

{

lbtenve.Visible = !lbtenve.Visible;

lbmsve.Visible = !lbmsve.Visible;

lbslt.Visible = !lbslt.Visible;

DropDownListVe.Visible = !DropDownListVe.Visible;

lbmsv.Visible = !lbmsv.Visible;

txtslt.Visible = !txtslt.Visible;

btnHuythem.Visible = !btnHuythem.Visible;

if (btnThem.Text == "Thêm")

{

DropDownListVe.Text = "";

lbmsv.Text = "";

txtslt.Text = "";

btnThem.Text = "Lưu thông tin";

}

else

{

try

{

SqlDataSourceVeTon.Insert();

GridViewVeTon.DataBind();

this.Title = "Thêm thành công";

}

catch (System.Exception ex)

{

this.Title = "Lỗi thêm" + ex.Message;

};

btnThem.Text = "Thêm";

}

}

protected void btnHuythem\_Click(object sender, EventArgs e)

{

lbtenve.Visible = false;

lbmsve.Visible = false;

lbslt.Visible = false;

DropDownListVe.Visible = false;

lbmsv.Visible = false;

txtslt.Visible = false;

btnHuythem.Visible = false;

btnThem.Text = "Thêm";

}

protected void DropDownListVe\_SelectedIndexChanged(object sender, EventArgs e)

{

lbmsv.Text = DropDownListVe.SelectedValue;

}

}

## Trang quản lý khách hàng (**FrCustumer.aspx**)

Đối với quản lý khách hàng chỉ có thao tác sửa và thao tác xóa tuyệt đối không có thao tác thêm (bên quản lý) vì nhân viên bên bán hàng không có quyền thêm khách hàng mới.

Các câu lệnh SQL:

SELECT \* FROM KhachHang

Xóa tức là hủy khách hàng

DELETE FROM KhachHang WHERE (masokhachhang = @masokhachhang)

Sửa tức là điều chỉnh thông tin của khách hàng theo yêu cầu của khách hàng hoặc yêu cầu của công ty, ví dụ khách hàng đề nghị sửa chữa địa chỉ.

UPDATE KhachHang SET tenkhachhang = @tenkhachhang, diachi = @diachi, sodienthoai = @sodienthoai WHERE (masokhachhang = @masokhachhang)

## Trang quản lý đơn mua vé (**FrPurchaseOrder.aspx**)

Đối với quản lý đơn mua vé chỉ có thao tác sửa và thao tác xóa tuyệt đối không có thao tác thêm (bên quản lý) vì nhân viên bên bán hàng không có quyền thêm đơn mới.

Lý do: bởi vì nếu thêm được tức là thêm đơn hàng ảo

Các câu lệnh SQL:

SELECT \* FROM DonMuaVe

Xóa tức là hủy đơn tại vì đơn khống, giả hoặc đơn lỗi kĩ thuật.

DELETE FROM DonMuaVe WHERE (masodonmuave = @masodonmuave)

Sửa tức là điều chỉnh thông tin của đơn hàng theo yêu cầu của khách hàng hoặc yêu cầu của công ty.

UPDATE DonMuaVe SET ngaylap = @ngaylap, masokhachhang = @masokhachhang, masonhanvien = @masonhanvien, diachigiao = @diachigiao, nguoinhan = @nguoinhan, sodienthoai = @sodienthoai, ghichu = @ghichu, masokhuvuc = @masokhuvuc WHERE (masodonmuave = @masodonmuave)

## Trang quản lý chi tiết đơn mua vé (**FrDetailedPurchaseOrder.aspx)**

Bảng chi tiết đơn đơn mua vé thì có đầy đủ chức năng thêm, xóa, sửa (bên quản lý)

SELECT \* FROM ChiTietDonMuaVe

Sửa tức là điều chỉnh thông tin của đơn hàng theo yêu cầu của khách hàng hoặc yêu cầu của công ty.

UPDATE ChiTietDonMuaVe SET soluong = @soluong, masove = @masove WHERE (masodonmuave = @masodonmuave)

Xóa tức là hủy khách hàng, hủy đơn tại vì đơn khống, giả hoặc đơn lỗi kĩ thuật.

DELETE FROM ChiTietDonMuaVe WHERE (masodonmuave = @masodonmuave)

Có thêm bởi vì nhân viên bên quản lý có thể thêm 1 chi tiết cho đơn hàng được ví dụ hàng khuyến mãi, hoặc là hàng cho biếu tặng.

INSERT INTO ChiTietDonMuaVe(masodonmuave, masove, soluong) VALUES (@masodonmuave, @masove, @soluong)

Codes xử lý thêm

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace G322DinhVanThanhVeOnline

{

public partial class FrDetailedPurchaseOrder : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

}

protected void btnThem\_Click(object sender, EventArgs e)

{

lbmsdmv.Visible = !lbmsdmv.Visible;

lbmsv.Visible = !lbmsv.Visible;

lbsl.Visible = !lbsl.Visible;

txtdmv.Visible = !txtdmv.Visible;

txtmsv.Visible = !txtmsv.Visible;

txtsl.Visible = !txtsl.Visible;

btnHuyThem.Visible = !btnHuyThem.Visible;

if (btnThem.Text == "Thêm chi tiết")

{

txtdmv.Text = "";

txtmsv.Text = "";

txtsl.Text = "";

btnThem.Text = "Lưu thông tin";

}

else

{

try

{

SqlDataSourceCTD.Insert();

GridViewCTD.DataBind();

this.Title = "Thêm thành công";

}

catch (System.Exception ex)

{

this.Title = "Lỗi thêm mới" + ex.Message;

};

btnThem.Text = "Thêm chi tiết";

}

}

protected void btnHuyThem\_Click(object sender, EventArgs e)

{

lbmsdmv.Visible = false;

lbmsv.Visible = false;

lbsl.Visible = false;

txtdmv.Visible = false;

txtmsv.Visible = false;

txtdmv.Visible = false;

btnThem.Text = "Thêm chi tiết";

}

}

}

## Chuyển nhóm vé : Chuyển vé từ nhóm này sang nhóm khác

Managements.aspx : Gọi từ trang quản lý

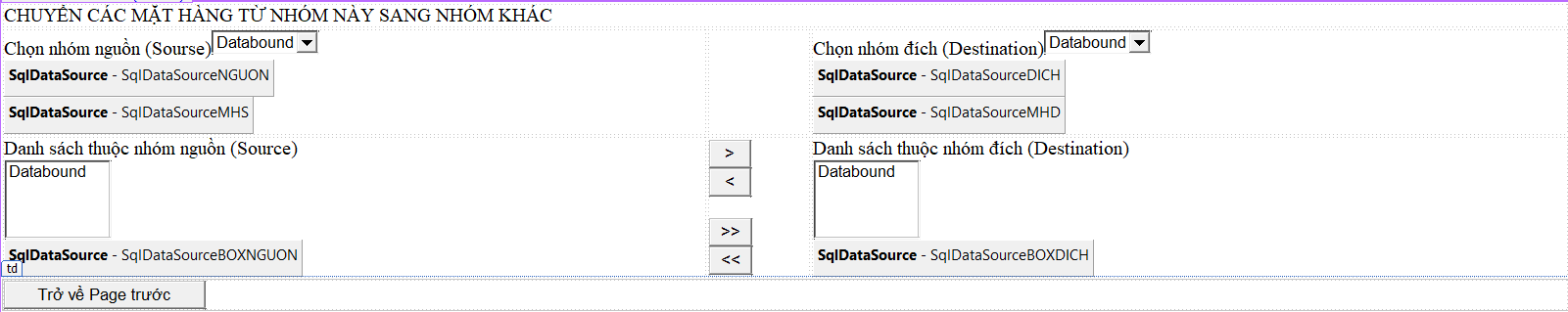
   protected void btnCnv\_Click(object sender, EventArgs e)

{

Response.Redirect("~\\FrChuyenNhomVe.aspx");

}

Thiết kế WebForm chuyển nhóm vé như sau:



Dropdownlist Nguồn: DropDownListS

SELECT \* FROM [DanhMucLoaiVe]

Dropdownlist Đích: DropDownListD

SELECT \* FROM [DanhMucLoaiVe]

ListBox Nguồn: ListBoxS

SELECT \* FROM [DanhMucVe] WHERE ([masoloaive] = @masoloaive)

ListBox Đích: ListBoxD

SELECT \* FROM [DanhMucVe] WHERE ([masoloaive] = @masoloaive)

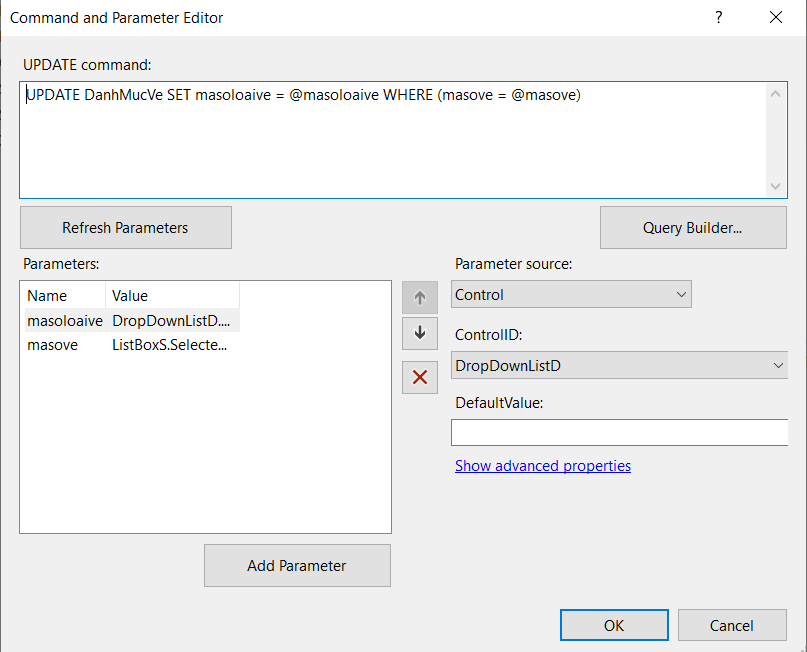
**Viết lệnh Update của SqlDataSource Nguồn và Đích FrChuyenNhomVe.aspx**

CẢ Nguồn và Đích đều viết lại câu lệnh SQL Update:

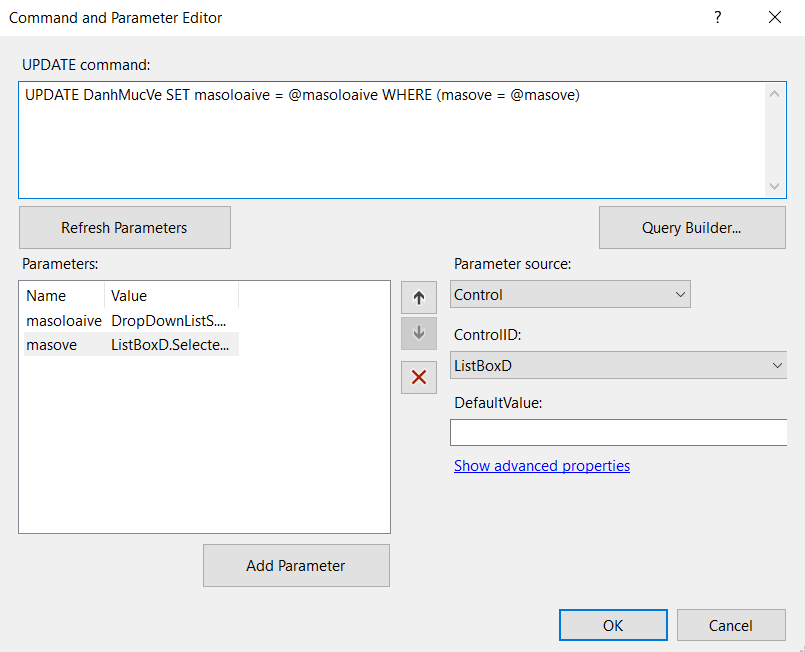
SqlDataSourceMHS  và SqlDataSourceMHD

UPDATE DanhMucVe SET masoloaive = @masoloaive WHERE (masove = @masove)

Tham số bên Nguồn:



Tham số bên Đích:



**FULL CODES**

public partial class FrChuyenNhomVe : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

DropDownListS\_SelectedIndexChanged(sender, e);

}

protected void btnBack\_Click(object sender, EventArgs e)

{

Response.Redirect("~\\Managements.aspx");

}

protected void LR\_Click(object sender, EventArgs e)

{

SqlDataSourceMHS.Update(); // gọi Update() bên Nguồn (S) đề chuyển

ListBoxS.DataBind();//Tải DL sau Update lên Source

ListBoxD.DataBind();//Tải DL sau Update lên Destinatio

}

protected void RL\_Click(object sender, EventArgs e)

{

// CHUYỂN 1 MH TỪ NHÓM "ĐÍCH" SANG NHÓM "NGUỒN" : RIGHT to LEFT

SqlDataSourceMHD.Update(); // gọi Update() bên Đích (D) đề chuyển

ListBoxS.DataBind();//Tải DL sau Update lên Source

ListBoxD.DataBind();//Tải DL sau Update lên Destination

}

protected void ButtonLAR\_Click(object sender, EventArgs e)

{

SqlDataSourceMHS.UpdateCommand = "UPDATE DanhMucVe SET masoloaive = @masoloaive WHERE masoloaive = @masoloaive";

//Viết lại lệnh Update....không Where..

SqlDataSourceMHS.UpdateParameters.Clear(); //Xóa các tham số cũ

SqlDataSourceMHS.UpdateParameters.Add("masoloaive", DropDownListD.SelectedValue.Trim());

//Khai báo lại tham số cho Update...

SqlDataSourceMHS.UpdateParameters.Add("masoloaive", DropDownListS.SelectedValue.Trim());

SqlDataSourceMHS.Update(); // gọi Update() bên Nguồn (S) đề chuyển

ListBoxS.DataBind();//Tải DL sau Update lên Source

ListBoxD.DataBind();//Tải DL sau Update lên Destination

}

protected void ButtonRAL\_Click(object sender, EventArgs e)

{

SqlDataSourceMHD.UpdateCommand = "UPDATE DanhMucVe SET masoloaive = @masoloaive WHERE masoloaive = @masoloaive";

//Viết lại lệnh Update....không Where..

SqlDataSourceMHD.UpdateParameters.Clear(); //Xóa các tham số cũ

SqlDataSourceMHD.UpdateParameters.Add("masoloaive", DropDownListD.SelectedValue.Trim());

//Khai báo lại tham số cho Update...

SqlDataSourceMHD.UpdateParameters.Add("masoloaive", DropDownListS.SelectedValue.Trim());

SqlDataSourceMHD.Update(); // gọi Update() bên Nguồn (S) đề chuyển

ListBoxD.DataBind();//Tải DL sau Update lên Source

ListBoxS.DataBind();//Tải DL sau Update lên Destination

}

protected void DropDownListS\_SelectedIndexChanged(object sender, EventArgs e)

{

if (DropDownListS.SelectedValue.Trim() == DropDownListD.SelectedValue.Trim())

// 2 Bên đã chọn giống nhau

{

LR.Enabled = false;

RL.Enabled = false;

ButtonLAR.Enabled = false;

ButtonRAL.Enabled = false;

DropDownListS.ToolTip = DropDownListD.ToolTip = "HAI BÊN PHẢI KHÁC NHAU THÌ MỚI CHUYỂN ĐƯỢC";

}

else // 2 bên đã chọn KHÁC nhau

{

LR.Enabled = true;

RL.Enabled = true;

ButtonLAR.Enabled = true;

ButtonRAL.Enabled = true;

DropDownListS.ToolTip = DropDownListD.ToolTip = "";

}

}

protected void DropDownListD\_SelectedIndexChanged(object sender, EventArgs e)

{

if (DropDownListS.SelectedValue.Trim() == DropDownListD.SelectedValue.Trim())

// 2 Bên đã chọn giống nhau

{

LR.Enabled = false;

RL.Enabled = false;

ButtonLAR.Enabled = false;

ButtonRAL.Enabled = false;

DropDownListS.ToolTip = DropDownListD.ToolTip = "HAI BÊN PHẢI KHÁC NHAU THÌ MỚI CHUYỂN ĐƯỢC";

}

else // 2 bên đã chọn KHÁC nhau

{

LR.Enabled = true;

RL.Enabled = true;

ButtonLAR.Enabled = true;

ButtonRAL.Enabled = true;

DropDownListS.ToolTip = DropDownListD.ToolTip = "";

}

}

}

# TRIỂN KHAI PHÁT HÀNH WEBAPP

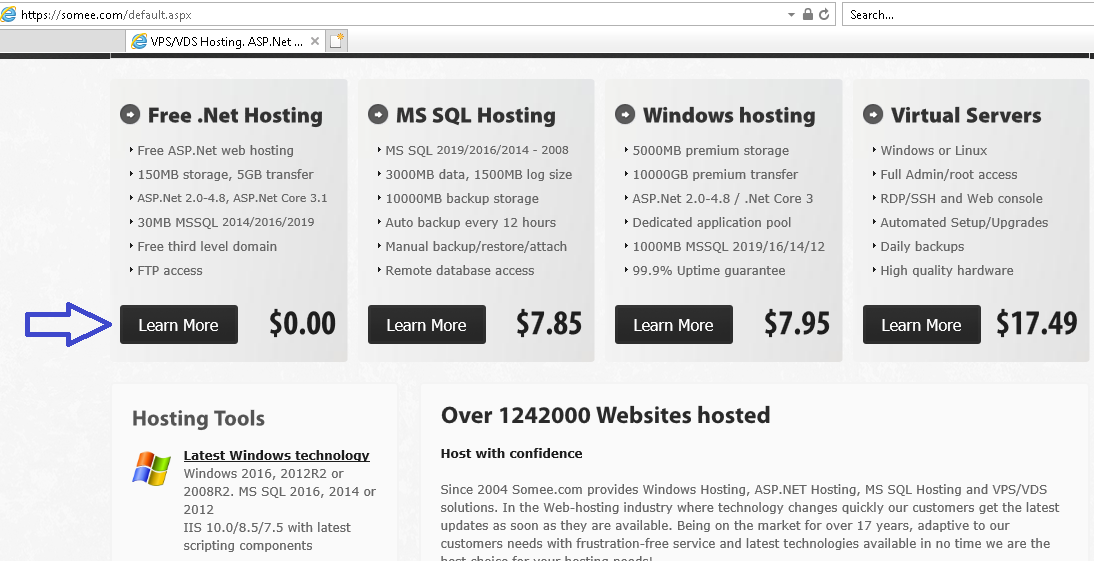
## Giới thiệu về Host miễn phí somee

+ Các host web phải đảm bảo phù hợp với nền tảng công nghệ mà WebApp đang lập trình phát triển sử dụng: Sản phản WebApp trong đề tài này là ASP.NET 4.6.1.

Vì vậy, Host Web được chọn để triển khai phát hành WebApp của đề tài này phải hỗ trợ AS.NET 4.6.1.

+ Có nhiều Host Web miễn phí, trong đề tài này sử dụng Free Host Web: somee như Hình *6*‑*1*

www.somee.com



Hình 6‑1. Trang chủ Host Web somee

+ Đặc điểm cơ bản của Free .Net Hosting:

. Dung lượng 150 MB

. Hỗ trợ ASP.NET từ 2.0 – 4.8 nền sử dụng được cho sản phẩm của đề tài là ASP.NET 4.7.2

. Dung lượng DB 30 MB MS. SQL Server 2014/2016/2019 nên sử dụng cho sản phẩm của đề tài nảy là MS.SQL 2018.

## Đăng ký tài khoản somee

**BƯỚC 1: Đăng ký tài khoản host somee & Đăng nhập vào host đó**

First name: **G3** Last name: **dvthanh**

User ID: **gthreedvthanh** Pass: @Vthanh2611

Email: dvthanh2611@gmail.com

URL: <http://ticketg322dvthanh.somee.com> OR http://www.ticketg322dvthanh.somee.com

Profile:

User ID: **gthreedvthanh**

First name: **G3**

Last name: **dvthanh**

Email address: dvthanh2611@gmail.com

FTP:

Addresses: ftp://ticketg322dvthanh.somee.com/www.ticketg322dvthanh.somee.com

ftp://155.254.244.28/www.ticketg322dvthanh.somee.com

Username: gthreedvthanh

Password: \* Use the password from this control panel \* @Vthanh2611

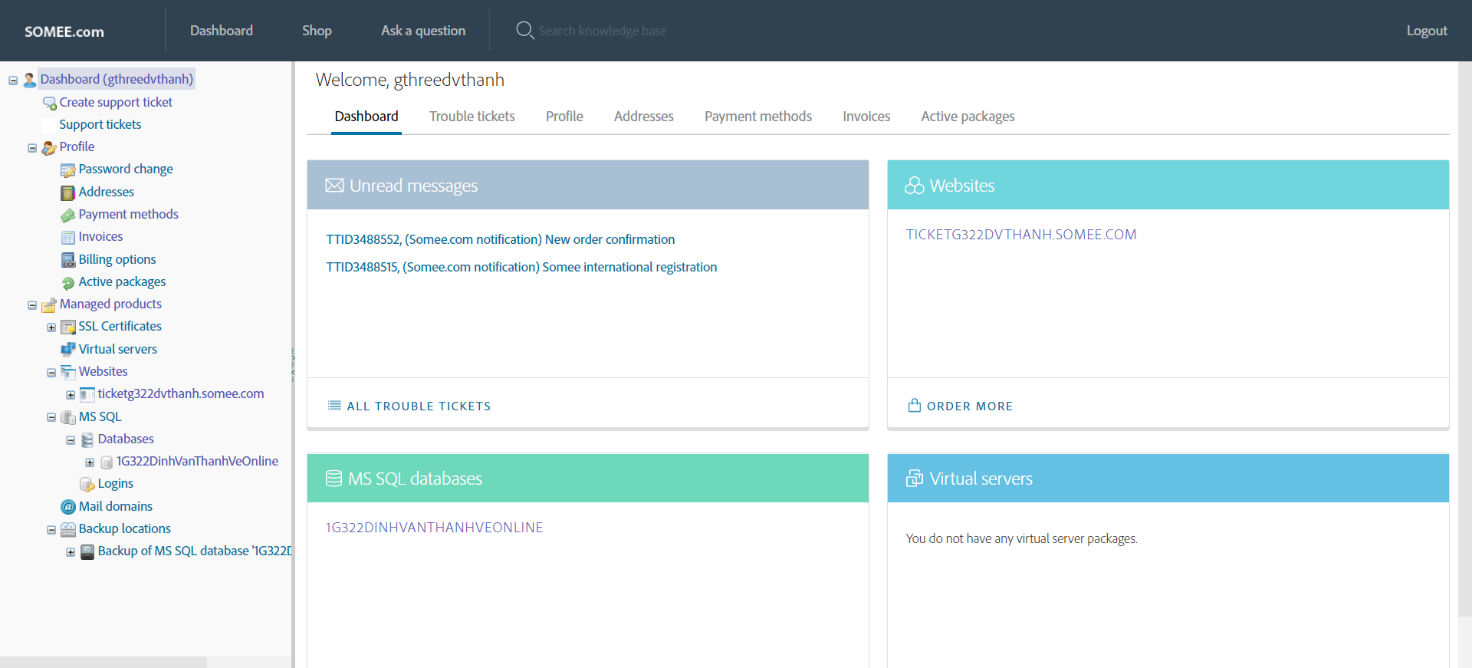
ASP.Net version: 4.0 – 4.8

Thông tin địa chỉ vật lý lưu trữ Web trên Host

Website local path: d:\DZHosts\LocalUser\gthreedvthanh\www.ticketg322dvthanh.somee.com

Protected local path: d:\DZHosts\LocalUser\gthreedvthanh\Protected.ticketg322dvthanh.somee.com

KHÔNG GIAN HOST ĐĂNG KÝ ĐƯỢC:



Hình 6‑2. Trang thông tin Host somee đã đăng ký

## Restore BD lên host somee

**BƯỚC 2: Tạo mới database trùng tên với DataBase của Web; Restore hoặc Attach DB**

Datebase name: 1G322DinhVanThanhVeOnline

THÔNG TIN KẾT NỐI DB

MS SQL Server version: MS SQL 2018 Express

MS SQL Server address: 1G322DinhVanThanhVeOnline.mssql.somee.com

Login name: gthreedvthanh\_SQLLogin\_1

Login password: hzp3ww338k

Connection string: workstation id=1G322DinhVanThanhVeOnline.mssql.somee.com;packet size=4096;user id=gthreedvthanh\_SQLLogin\_1;pwd=hzp3ww338k;data source=1G322DinhVanThanhVeOnline.mssql.somee.com;persist security info=False;initial catalog=1G322DinhVanThanhVeOnline

BACKUP DB

Address:   
ftp://1G322DinhVanThanhVeOnline.backup.somee.com/1G322DinhVanThanhVeOnline\_MSSql\_Database\_Backup

Username: gthreedvthanh

Password: \* Use the password from this control panel \*

Hosting plan: MS SQL Plan "Novice"

Supplied package ID: SPID1074428

Managed product ID: MPID3921733

Data file: Size, Max/Used/Usage: 30MB / 8.00MB / 26.67%

Log file: Size, Max/Used/Usage: 30MB / 8.00MB / 26.67%

**Backup file Manager => Restore**

Browse for local backup (50MB max): chọn file .bak từ PC : 1G322DinhVanThanhVeOnline.bak

chọn : Upload and Restore

## Upload đăng tải Source WebApp lên host somee để phát hành

**BƯỚC 3: Điều chỉnh nội dung Web.Config:** TRONG Web.config

<system.web>

<customErrors mode="Off" defaultRedirect="Homepage.aspx"/>

</system.web>

<system.webServer>

<defaultDocument>

<files>

<add value="Homepage.aspx" />

</files>

</defaultDocument>

<directoryBrowse enabled="false" />

</system.webServer>

**BƯỚC 4: Điều chỉnh Web.Config:** TRONG Web.config

Connection string: workstation id=1G322DinhVanThanhVeOnline.mssql.somee.com;packet size=4096;user id=gthreedvthanh\_SQLLogin\_1;pwd=hzp3ww338k;data source=1G322DinhVanThanhVeOnline.mssql.somee.com;persist security info=False;initial catalog=1G322DinhVanThanhVeOnline

**BƯỚC 5: Upload và đăng tải phát hành WebApp bán vé trực tuyến trên somee**

1. nén các file trong thư mục Web thành file .zip

2. Upload lên somee

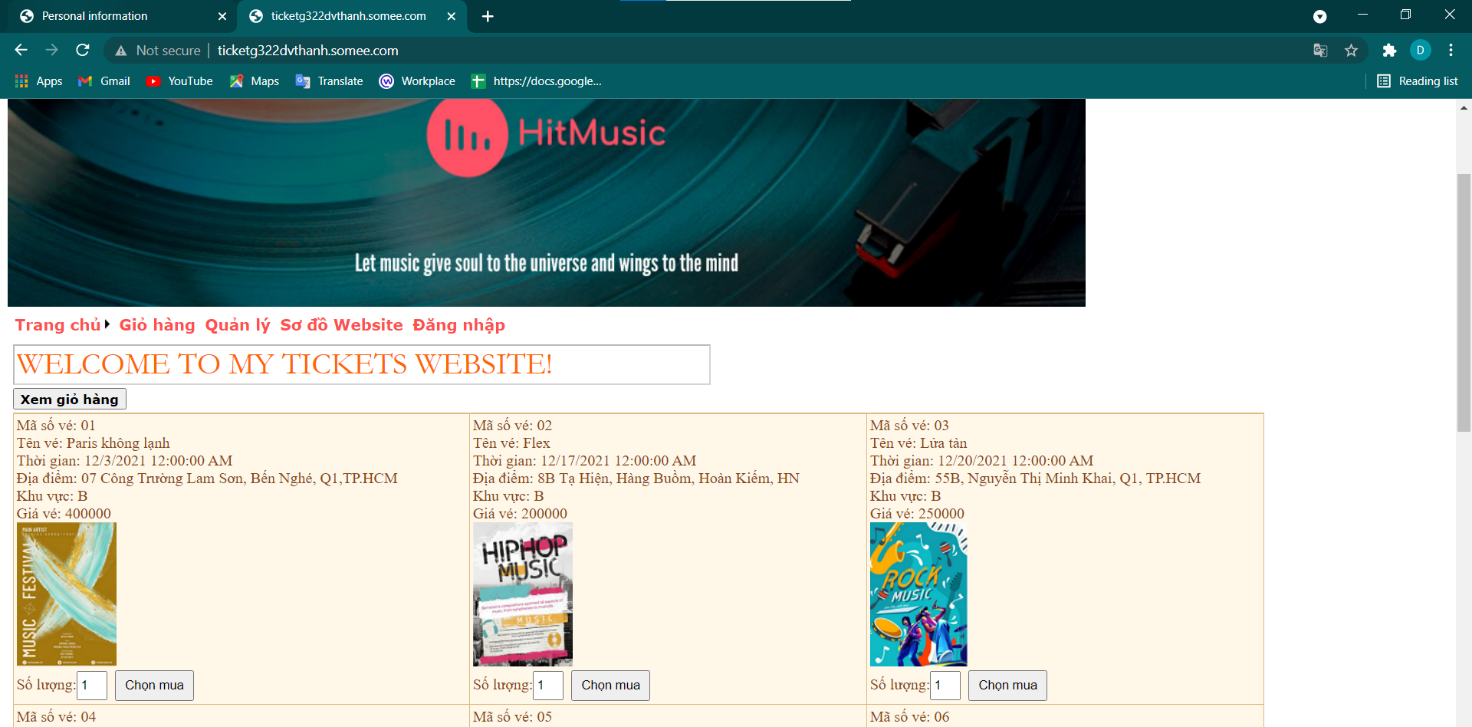
Trong Host: Manage Product->Websites-> ticketg322dvthanh.somee.com ->File Manager

Upload // chọn file .zip nêu trên // Upload and unzip

## Giới thiệu sản phẩm WebApp quản lý bán vé online trên somee

URL: http://ticketg322dvthanh.somee.com OR http://www.ticketg322dvthanh.somee.com

Trang chủ HomePage: HomePage.aspx



Trang chủ Giỏ hàng: Cart.aspx

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN CỦA ĐỒ ÁN

## **Kết luận**

## **Những kết quả đạt được**

Vận dụng các cơ sở lý thuyết lập trình Web, thiết kết và lập trình được:

+ Phần giao dịch:

. MaeterPage dùng chung cho cả giao dịch và quản lý

. HomePage: Trang chủ, trưng bày các loại vé sự kiện âm nhạc

. Giỏ hàng : Giỏ hàng chọn mua vé sự kiện âm nhạc

. Một phần của trang đặt hàng (Order)

+ Phần Quản lý

. Quản lý chung bao gồm cả chức năng đăng nhập (Login) dùng Codes; trang này bao gồm tất cả các chức năng quản lý, cho phép nhân viên chọn các chức năng quản lý cụ thể đến các trang khác.

. Quản lý danh mục các loại vé sự kiện âm nhạc: Thêm, Xóa, Sửa, RBTV

## **Hạn chế**

Do hạn chế thời gian, một số chức năng sau chưa thực hiện hoặc thực hiện chưa đầy đủ:

+ Chuyển nhóm.

+ Trang đặt hàng: chưa thực hiện đầy đủ phần thông tin đơn hàng, chi tiết đơn hàng và lưu trữ vào DB.

+ Chưa hoàn thành trọn vẹn được các chức năng quản lý.

## **Hướng phát triển**

## **Hướng khắc phục các hạn chế**

+ Sẽ tiếp tục hoàn thiện trang đặt hàng với chức năng đơn hàng và chi tiết đơn để lưu vào DB.

+ Thực hiện đầy đủ các trang chuyển nhóm.

+ Tiếp tục hoàn thiện sản phẩm minh họa (Demo) và báo cáo đề tài.

## **Hướng mở rộng ĐỒ ÁN HỌC PHẦN**

+ Tiếp tục phát triển sản phẩm với các công nghệ mới, như: MVC, SP, . . .

+ Thanh toán điện tử

+ Định vị giao hàng

+ Bổ sung chữ ký số

+ Điều tiết vé giữa các điểm bán vé

+ Khai phá dữ liệu: Khai phá số liệu bán vé của các điểm bán vé, đơn hàng, doanh số, doanh thu từ đó chọn tỉ lệ điều tiết khi nhập mới về đến các kho sao cho phù hợp.

# DANH MỤC TÀI LIỆU THAM KHẢO

1. Võ Xuân Thể (2021), *Bài giảng học phần Lập trình Web (4 Tín chỉ)*, Trường Đại học Tài chính – Marketing (lưu hành nội bộ).
2. https://vi.wikipedia.org/wiki/ASP.NET

1. Đây là khái niệm về “Mạng xã hội” [SN] theo Giảng viên học phần: Võ Xuân Thể [↑](#footnote-ref-1)