Implementation business-to-business electronic commerce website using active server pages

Sumuscha Teesri

Follow this and additional works at: https://scholarworks.lib.csusb.edu/etd-project

Part of the E-Commerce Commons

Recommended Citation
https://scholarworks.lib.csusb.edu/etd-project/3189

This Project is brought to you for free and open access by the John M. Pfau Library at CSUSB ScholarWorks. It has been accepted for inclusion in Theses Digitization Project by an authorized administrator of CSUSB ScholarWorks. For more information, please contact scholarworks@csusb.edu.
IMPLEMENTATION BUSINESS-TO-BUSINESS ELECTRONIC COMMERCE
WEBSITE USING ACTIVE SERVER PAGES

A Project
Presented to the
Faculty of
California State University,
San Bernardino

In Partial Fulfillment
of the Requirements for the Degree
Master of Business Administration:
Information Management

by
Sumuscha Teesri
December 2000
IMPLEMENTATION BUSINESS-TO-BUSINESS ELECTRONIC COMMERCE WEBSITE USING ACTIVE SERVER PAGES

A Project
Presented to the
Faculty of
California State University,
San Bernardino

by
Sumuscha Teesri
December 2000

Approved by:

Frank M. Lin, Ph. D., Chair

Harold Dyck, Ph.D.

Walter T. Stewart, Jr., Ph.D.

Date
10/19/00
ABSTRACT

There are three major methods used in this project. First, background research was performed to obtain information for a complete understanding of the toy wholesaler industry. Second, a simulated toy wholesaler company was created, incorporating its business strategies based on the business assumptions. Finally, an electronic business-to-business website application was developed to ensuring that the business is accessible by its customers.

The final application can be operated in three different network environments. In the real-world business situation, the completed application must be operated on the N-tiered network environment to increase security, reliability, and performance of the application. Nevertheless, to make it easier, this business application was created in the two-tiered client/server network environment during the development process. The three-tiered network architecture environment, however, was exploited during the implementation and presentation process to imitate the N-tiered network architecture.
The completed project delivers a business-to-business electronic commerce simulation application running on Windows NT and Windows 2000 systems. The application is built in such a manner that its functions are successfully implemented in the business environment. However, as it is certain that there is no bug-free program in the real world, likewise this application is not bug-free also, since it has been used to merely “simulate” business functions, the completed program might not work perfectly in the actual business environment.
ACKNOWLEDGMENTS

I would like to thank my project committee professors; Professors Frank M. Lin, Ph.D., Committee Chair, Professors Harold Dyck, Ph.D., Second Faculty Reader, and Professors Walter T. Stewart, Jr., Ph.D., Department Chair, for their support in developing the project. Their valuable guidance and suggestions contributed substantially to the project.

I like to express my gratitude to Miss Suwanna Visavapattamawon, who takes care of me during my stay in Southern California, and to Mr. Teerapong Buawatana, my project mate, who suggested the project option to me instead of taking the comprehensive exam.

Finally, I would express my appreciation to my father, whose patience made my education possible, and to my mother, who supported me in everything I did whether I failed or succeed. Big, my older bother, always told me to believe in myself, and my younger bother, Boss, reminded me to do the right thing.

Sumuscha Teesri
To Mom, Big and Boss
TABLE OF CONTENTS

ABSTRACT ........................................................................... iii

ACKNOWLEDGMENTS ............................................................ v

LIST OF TABLES

TABLES .................................................................................. x

FIGURES ............................................................................... xii

CHAPTER ONE: INTRODUCTION

1.1 Preview ............................................................... 1

1.2 Purpose of the Project ............................................. 4

CHAPTER TWO: BACKGROUND OF ELECTRONIC COMMERCE

2.1 E-Commerce Introduction ................................. 9

2.1.1 E-Commerce Definition ................................. 9

2.1.2 Types of E-Commerce ............................... 10

2.1.2.1 Electronic Tailing or The Virtual Storefront and The Virtual Mall 11

2.1.2.2 Market Research ........................... 12

2.1.2.3 Electronic Data Interchange (EDI) ....... 12

2.1.2.4 Electronic Mail, Fax and Internet Telephony .............. 13

2.1.2.5 Business-to-Business Buying and Selling .......................... 13
2.2 E-Commerce in Today Business .................. 14
  2.2.1 Direct Marketing, Selling, and Service 14
  2.2.2 Online Banking and Financial Services . 17
  2.2.3 Corporate Purchasing ...................... 19
  2.2.4 Value Chain Integration .................... 20
2.3 The E-Commerce and Common Commerce .......... 23
2.4 Benefits of E-Commerce .......................... 25
2.5 Issues in Implementing E-Commerce ............. 27
2.6 Requirements of an E-Commerce Site .......... 29

CHAPTER THREE: TOY WHOLESALER INDUSTRY

3.1 Five-force Analysis .............................. 33
  3.1.1 Entry Barriers ............................. 33
  3.1.2 Determinants of Supplier Power .......... 37
  3.1.3 Rivalry Determinants ..................... 41
  3.1.4 Determinants of Buyer Power ............. 44
  3.1.5 Determinants of Substitution Threat .. 47
  3.1.6 Conclusion ............................... 48
3.2 Macrotoys ...................................... 50
  3.2.1 Competitive Advantage:
      The Differentiation .......................... 50
  3.2.2 Business Assumptions ..................... 52
  3.2.3 Business Scopes ........................... 53
  3.2.4 Strategy ................................. 55
CHAPTER FOUR: ACTIVE SERVER PAGES TECHNOLOGY

4.1 Overview .................................................. 59
4.2 What’s ASP? .................................................. 60
4.3 How ASP Works .............................................. 62
4.4 Benefits of ASP ............................................. 64

CHAPTER FIVE: DEVELOPMENT PROCESS

5.1 Planning ...................................................... 69
5.2 Designing .................................................... 74
  5.2.1 Data Flow Diagram ..................................... 74
  5.2.2 Database Design ....................................... 83
     5.2.2.1 E-R Diagram ..................................... 83
     5.2.2.2 Data Dictionary .................................. 85
  5.2.3 Network Design ........................................ 96
  5.2.4 Web Design ............................................ 100
     5.2.4.1 Main Website ..................................... 100
     5.2.4.2 Admin Website ................................... 110

CHAPTER SIX: IMPLEMENTATION

6.1 Overall Implementation Process .......................... 121
6.2 Database Implementation .................................. 124
6.3 Environment ............................................... 126
  6.3.1 Development Environment ............................. 127
  6.3.2 Operating Environment ............................... 130
6.4 Requirements .............................................. 133
6.5 The B2BECA Features .................................. 136
6.6 How to Setup an E-commerce Site ................. 146
  6.6.1 Domain Name Registration .................... 146
  6.6.2 Hosting the E-commerce Website .......... 147
    6.6.2.1 Outsourcing Web Hosting .......... 147
    6.6.2.2 In House Hosting .................. 148

CHAPTER SEVEN: CONCLUSION

  7.1 What's Next? ...................................... 150
  7.2 Conclusion ...................................... 152

APPENDIX A. ASP SOURCE CODE

  Main Website Programming code .................... 154
  Admin Website Programming code ................. 222

APPENDIX B. DATABASE SOURCE CODE .................. 292

REFERENCES ........................................... 305
LIST OF TABLES

Table 3.1 Entry Barriers of the Toy Wholesaler Industry .................................. 37
Table 3.2 Determinants of Supplier Power of the Toy Wholesaler Industry ................ 40
Table 3.3 Rivalry Determinants of the Toy Wholesaler Industry ................................ 44
Table 3.4 Determinants of Buyer Power of the Toy Wholesaler Industry ...................... 47
Table 3.5 Determinants of Substitution Threat of the Toy Wholesaler Industry ............... 48
Table 3.6 Five-forces Analysis Conclusion ......................................................... 49
Table 3.7 Three Generic Strategies ................................................................. 51
Table 5.1 Planning Gantt Chart ................................................................. 70
Table 5.2 Customers Table ................................................................. 87
Table 5.3 Employees Table ................................................................. 88
Table 5.4 Orders Table ................................................................. 89
Table 5.5 Transact Table ................................................................. 89
Table 5.6 Products Table ................................................................. 90
Table 5.7 Suppliers Table ................................................................. 91
Table 5.8 CustomerTypes Table .............................................................. 91
Table 5.9 OrderStatus Table .............................................................. 92
Table 5.10 PaymentTypes Table .............................................................. 92
Table 5.11 Carriers Table .............................................92
Table 5.12 ShippingRates Table .................................93
Table 5.13 WeightZoneID Table .................................93
Table 5.14 States Table .............................................93
Table 5.15 StateZones Table .......................................94
Table 5.16 SizeZones Table .......................................94
Table 5.17 SubCategories Table .................................94
Table 5.18 ProductCats Table .................................95
Table 5.19 Categories Table .......................................95
Table 5.20 WebStats Table .................................96
Table 5.21 Main Website File Detail .................101 – 110
Table 5.22 Admin Website File Detail .................112 – 120
Table 6.1 Implementation Gantt Chart .................123
LIST OF FIGURES

Figure 2.1 Dell Computer Corporation on the Direct Sales Market .................................................. 12
Figure 2.2 Starbucks Coffee Using E-commerce Technology to Build an Online Store ...................... 16
Figure 2.3 Mobil Oil Corporation with TransPoint, Online Bill Delivery and Payment System .......... 18
Figure 2.4 LA County Using Online Purchasing Process to Order Goods and Services ................. 20
Figure 2.5 Sainsbury’s Supermarkets ......................... 21
Figure 2.6 Electronic Commerce Process ....................... 30
Figure 4.1 ASP Process .............................................. 63
Figure 5.1 DFD: Conceptual Level .................................. 75
Figure 5.2 DFD Level 1 .............................................. 76
Figure 5.3 DFD Level 1.0: Search Information .............. 77
Figure 5.4 DFD Level 2.0: Registration ......................... 78
Figure 5.5 DFD Level 3.0: Order ................................. 79
Figure 5.6 DFD Level 4.0: Bill and Receipt .................. 80
Figure 5.7 DFD Level 5.0: Packaging .......................... 81
Figure 5.8 DFD Level 6.0: Shipping ........................... 82
Figure 5.9 Entry-Relationship Design Diagram .................. 84
Figure 5.10 Network Design Diagram ........................... 97
Figure 5.11 Main Website Design Diagram ................. 101
Figure 5.12 Admin Website Design Diagram ...........111
Figure 6.1 SQL Server Database Diagram ...............125
Figure 6.2 Development Environment ...................127
Figure 6.3 Three-tier Network Architecture ............130
Figure 6.4 Default Page ..................................137
Figure 6.5 Browser Page ..................................138
Figure 6.6 Product Detail Page .........................139
Figure 6.7 Shopping Cart Page ..........................140
Figure 6.8 Registration Page ...........................141
Figure 6.9 Member Login Page ..........................142
Figure 6.10 Checkout Page ...............................143
Figure 6.11 Payment Information Page .................144
Figure 6.12 Order Status Page ..........................145
CHAPTER ONE: INTRODUCTION

1.1 Preview

The driving force of this project comes from the hottest issues in today's business: electronic commerce. With current information technology, there are many methods to implement an electronic commerce website for a business such as the Apache with CGI script on a Linux operating system via MSQQL or MYSQL database management system. Other methods include the Microsoft Internet Information Server 4.0 with ASP, Cold Fusion, or even CGI on Windows NT operating system via Microsoft SQL Server, or a Novell IntranetWare with Oracle 8 database management system, and etc.

Nevertheless, the ASP approach offers more advantages than other approaches: the total cost of ownership is cheaper\(^1\); the Internet Information Server and Windows NT also provide higher performance than Apache and Linux or

\(^1\) According to PC Week lab's tests: See reference [1]
other UNIX systems\(^2\); and there are many third-party tools supporting Microsoft technology. These advantages allow many companies to develop scalable, reliable, easy to use, and faster performance commerce websites than before.

In this chapter, the whole picture of the project will be discussed. In the next chapter, the background of the electronic commerce, which includes types, benefits, features, and requirements of electronic commerce, will be provided. In chapter three, the concepts learned in the Information Management System and the Information System Planning, Strategy and Policy classes will be applied. The five-forces analysis of the toy wholesaler industry, which is the simulated business, will be developed. The business scope, including vision, mission, business goal, and business strategy, will be discussed in this chapter as well.

\(^2\) According to Microsoft Lab’s tests: see reference [8]
The project also applies the web-based solution technology taught in the World Wide Web strategy and the Electronic Commerce classes in chapter four. Then, the general ideas in the Information Base System and Information Networking System classes, such as the database design and the network design, will be utilized in chapters five and six. The development process and implementation of the project, which will apply the concept of software development life cycle, will be explained in these chapters as well. After that, the project concludes with a chapter which summaries the benefits of the project.

The project documents, such as ASP, and Database Generating code are also included in the appendices. At the end of this document, all of the books used during the study phase and the document used as reference for the entire project are listed.
1.2 Purpose of the Project

To be successful in any highly contested globalized business today, companies must distinguish their businesses from other challengers by increasing their competitive advantages. Those who can apply the information technology infrastructure in their businesses appear to reach higher accomplishment than other players do. The hottest information technology now is business-to-business electronic commerce or B2B, which assists any corporation in achieving its goal quickly.

E-commerce is the current approach for doing any type of business online, which uses the superior power of digital information to understand the requirements and preferences of each client and each partner, to adapt products and services for them, and then to distribute the products and services as swiftly as possible. Automated services offer businesses the ability to increase revenues, decrease costs, and generate and support customers' and partners' relationships.
To accomplish these benefits, many companies today employ e-commerce for direct marketing, selling, and customer service such as online banking and billing, secure distribution of information, value chain trading, and corporate purchasing. However, while the benefits of e-commerce systems are enticing, to develop, deploy, and manage this system is not always straightforward.

To implement modern technology, many corporations will need to reengineer their business processes to enhance the benefits of e-commerce. There are also many different tools used to establish an e-commerce website. Since information technology moves extremely fast, every organization must make good decisions when selecting these tools, otherwise they will be wasted, and the company will only increase trouble instead of competitive advantage.

This project explores the features of contemporary Internet and information technology such as Hypertext Makeup Language, Active Server Pages, Database Management System, Operating System, Client Server Architecture, and Internet Server to establish an electronic commerce site. This paper will simulate a toy wholesaler to illustrate how information
business management knowledge can be used to implement a business-to-business commerce website, and to accomplish a corporation's goal. HTML and ASP will be used as the major programming techniques for the entire project.

The major purpose of this project is to apply all knowledge learning in the Management Information System major of the MBA program at CSUSB. This knowledge is used to create a business-to-business e-commerce application (B2B ECA) by using the available Internet and information management technology. The project also applies business administration and information management theory into the illustrated business (Toy Wholesaler). A simulated company is set up and business rules are applied to the company similar to the real world organization.

The simulated company, Macrotoys, is operating in the toy wholesale industry. Macrotoys has a vision, mission and goals like any other actual company. By using B2B ECA, Macrotoys can take orders from toy retailers online through the Internet via the World Wide Web. Macrotoys' employees are also able to handle, monitor, and fulfill orders using B2B ECA.
In this project, an electronic commerce business strategy will be established to assist in delivering a technology platform, a gateway for online services, and a professional know-how that companies can leverage to implement new ways of doing today's business. Each electronic commerce business requires development of low-cost, high-value commercial platforms that are easy to grow as business grows.

The platforms (which are the foundation of technologies and products that enable and support electronic commerce) deployed in the project range from operating systems, such as Microsoft Windows NT® Server, to application servers, such as Microsoft SQL Server® and Oracle 8® on Windows NT®, to applications infrastructure and development tools, such as the Microsoft Visual InterDev®, Microsoft FrontPage®, and Active Server Pages programming script language. All of these platforms help businesses implement commerce systems quickly and with low risk. The simulated business, a toy wholesaler, will be established based on these robust and well-known platforms.
The final step of the project will be to assure the full functionality of business-to-business e-commerce application is ready to be deployed. This includes debugging and testing the application in a computer system environment.
CHAPTER TWO: BACKGROUND OF ELECTRONIC COMMERCE

2.1 E-commerce Introduction

In this section, the background of e-commerce will be explained. This includes the definition and types of e-commerce. Many people usually limit the e-commerce to only buying or selling goods and services online, e-tailing. The section will explain what the e-commerce really entails.

2.1.1 E-Commerce Definition

Electronic commerce, (E-commerce or EC) is any online business transaction in which the parties involved interact electronically. The online business transactions include buying and selling of goods and services on the Internet, especially the World Wide Web. It is often misunderstood to be limited to buying and selling of goods and services over the Internet. Actually, e-commerce solutions are a lot more than just the handling of business transactions and fund transfers over the Internet. It defines new forms of doing the old business. In addition to providing buying and selling services, e-commerce solutions can provide a
complete system of services built into an organization's
digital system so it supports the sales processes and
provides total account management. In practice, this term
and a new term, "e-business," are often used
interchangeably.

2.1.2 Types of E-Commerce

Electronic Commerce can be categorized into the following
categories:

- Electronic tailing or "virtual storefronts" on Websites
  with online catalogs, sometimes gathered into a
  "virtual mall"
- The gathering and use of demographic data through Web
  contacts
- Electronic Data Interchange (EDI), the business-to-
  business exchange of data
- Electronic mail and fax and their use as media for
  reaching prospects and established customers (for
  example, with newsletters)
- Business-to-business buying and selling
Each of the above categories is detailed in the following sections.

2.1.2.1 Electronic Tailing or The Virtual Storefront and The Virtual Mall: As a place for direct retail shopping, with its 24-hour availability, a global reach, the ability to interact and provide custom information and ordering, and multimedia prospects, the web is rapidly becoming a multibillion-dollar source of revenue for the world's businesses. As early as the middle of 1997, Dell Computers reported orders of a million dollars a day (see figure 2.1). By early 1999, projected e-commerce revenues for business were in the billions of dollars and the stocks of companies deemed most adept at e-commerce were skyrocketing. Apart from computer and network products, books (Amazon.com), gardening products (Garden.com), music on compact disks (CDNow), and office supplies (SuppliesOnline) were a few of the better-known e-commerce sites. By early 1999, even businesses were planning e-commerce websites and many businesses were planning how to coordinate in-store and web store retail approaches. Meanwhile, new businesses based entirely on web sales were being invented daily.
Dell Computer Corporation is one of the world’s largest computer manufacturers, generating $16.8 billion in sales in its last four quarters. The Internet has provided a great opportunity for Dell’s direct marketing and selling approach. The company sells more than $10 million worth of computers a day over the World Wide Web through an e-commerce system. Dell has also leveraged this system to improve customer service while reducing costs.

Figure 2.1.3. Dell Computer Corporation on the direct sells market.

2.1.2.2 Market Research: In early 1999, it was widely recognized that because of the interactive nature of the Internet, companies could gather data about prospects and customers in unprecedented amounts -through site registration, questionnaires, and as part of taking orders. The issue of whether data was being collected with the knowledge and permission of market subjects had been raised.

2.1.2.3 Electronic Data Interchange (EDI): EDI is the exchange of business data using an understood data format. It predates today’s Internet. EDI involves data exchange among parties that know each other well and arrange for one-to-one (or point-to-point) connection.

\[3\] See reference [9]
2.1.2.4 Electronic Mail, Fax, and Internet Telephony: E-commerce is also conducted through the more limited electronic forms of communication called e-mail, facsimile or fax, and the emerging use of telephone calls over the Internet. Most of this is business-to-business, with some companies attempting to use e-mail and fax for unsolicited ads (usually viewed as online junk mail or Spam) to consumers and other business prospects. An increasing number of business Websites offer e-mail newsletters for subscribers. A new trend is opt-in e-mail in which Web users voluntarily sign up to receive e-mail, usually sponsored or containing ads, about product categories or other subjects they are interested in.

2.1.2.5 Business-to-Business Buying and Selling: Thousands of companies that sell products to other companies have discovered that the Web provides not only a 24-hour-a-day showcase for their products but a quick way to reach the right people in a company for more information.
2.2 E-Commerce in Today Business

Businesses communicate with customers and partners through channels. The Internet and electronic commerce are the newest and, for many purposes, the best business communication channels. They are fast, reasonably reliable, inexpensive, and universally accessible—they reach virtually every business and more than 100 million consumers. Electronic commerce provides four main benefit areas where companies can take advantages of business online today: direct marketing, selling, and service; online banking and billing with a secure distribution of information; corporate purchasing; and value chain trading.

2.2.1 Direct Marketing, Selling, and Service

Today, more Websites focus on direct marketing, selling, and service than on any other type of e-commerce. Direct selling was the earliest type of e-commerce, and has proven to be a stepping-stone to more complex commerce operations for many companies. Successes such as Amazon.com, Barnes & Noble, Dell Computer, Starbucks (see figure 2.2) and the introduction of e-tickets by major airlines have
catalyzed the growth of this segment, and proving the reach and customer acceptance of the Internet. Across consumer-targeted commerce sites, there are several keys to success:

➤ Marketing that creates site visibility and demand; targets customer segments with personalized offers; and generates qualified sales leads through observation and analysis of customer behaviors.

➤ Sales-enhancing site design that allows personalized content and adaptive selling processes that do more than just list catalog items.

➤ Integrated sales-processing capabilities that provide secure credit card authorization and payment, automated tax calculation, flexible fulfillment, and tight integration with existing back-end systems such as inventory, billing, and distribution.

➤ Automated customer service features that generate responsive feedback to consumer inquiries; capture and track information about consumer requests; and automatically provide customized services based on personal needs and interests.
This business-to-consumer electronic commerce increases the revenue by reaching the right customers more often. Targeted and automated up selling and cross selling are the new fundamentals of online retailing. Sites that most frequently provide the best and most appropriate products and services are rewarded with stronger customer relationships, resulting in improved loyalty and increased value.

Starbucks, one of the world's premier purveyors of fine coffee, has gone online with a Web site and virtual store. Built by the San Francisco-based Web-development firm Organic, the site was developed using e-commerce technologies. The site will allow people to have the "Starbucks Experience" while helping Starbucks heighten brand awareness, widen its customer base, and increase sales. Through the online store, visitors can purchase coffee, coffee samplers, coffee-related gifts, CDs, and more. A store locator with mapping functionality helps users find the Starbucks location nearest them, while the Coffee "Taste Matcher" interactively recommends coffees based on a user's preferences.

Figure 2.2 Starbucks Coffee uses e-commerce technology to build an online store

---

4 See reference [14]
2.2.2 Online Banking and Financial Services

A broad range of financial and information services are performed over the Internet today, and sites that offer them are enjoying rapid growth. These sites are popular because they help consumers, businesses of all sizes, and financial institutions distribute some of their most important information over the Internet with greater convenience and richness than is available using other channels. For example:

- Online Banking: Consumers and small businesses can save time and money by doing their banking on the Internet. Paying bills, making transfers between accounts, and trading stocks, bonds, and mutual funds can all be performed electronically by using the Internet to connect consumers and small businesses with their financial institutions.

- Online Billing: Companies that bill can achieve significant cost savings and marketing benefits with Internet-based bill-delivery and receiving systems. Today, consumers receive an average of 12 bills a month by mail from retailers, credit card companies, and utilities. If these bills can be sent online, the
companies will save costs of bill-delivery. For example, Mobil Oil Company is using an online billing system called TransPoint to send the bills to its members online (see figure 2.3).

Secure Information Distribution: To many businesses, information is their most valuable asset. While the Internet can enable businesses to reach huge new markets for that information, businesses must also safeguard that information to protect their assets. Digital Rights Management provides protection for intellectual and information property, and is a key technology for secure information distribution.

Mobil Oil Corporation is not only a leading oil and gas company but also a massive creditor - the company sends out 36 million credit card statements each year. To lower billing costs and to take advantage of the Internet's ability to market to individual customers through customized billing statements, Mobil is piloting the TransPoint Internet-based bill delivery and payment system.

Figure 2.3 Mobil Oil Corporation with TransPoint, online bill delivery and payment system.

5 See reference [12]
2.2.3 Corporate Purchasing

The Internet offers tremendous time and cost savings for corporate purchasing of low-cost, high-volume goods for maintenance, repair, and operations (MRO) activities. The good example of corporate purchasing is at LA county (see figure 2.4). Typical MRO goods include office supplies such as pens and paper, office equipment and furniture, computers, and replacement parts. The Internet can transform corporate purchasing from a labor- and paperwork-intensive process into a self-service application. Company employees can order equipment on Websites; company officials can automatically enforce purchase approval through automated business rules; and suppliers can keep their catalog information centralized and up-to-date. Purchase order applications can then use the Internet to transfer the order to suppliers. In response, suppliers can ship the requested goods and invoice the company over the Internet. In addition to reduced administrative costs, Internet-based corporate purchasing can improve order-tracking accuracy; better enforce purchasing policies; provide better customer and supplier service; reduce inventories; and give companies more power in negotiating exclusive or volume-discount contracts.
The largest local government in the United States, Los Angeles County buys $650 million of goods and services—everything from hard drives to helicopters—each year from over 25,000 bidders. Until recently, all of those purchases were made using paper forms. Now, a Web-based solution from a leading software vendor automates the county’s purchasing process. Buyers shop, order and pay for goods over the Internet, helping the county to save $6 million per year.

Figure 2.4 LA County uses an online purchasing process to order goods and services.

2.2.4 Value Chain Integration

No other business model highlights the need for tight integration across suppliers, manufacturers, and distributors quite like the value chain. Delays in inventory tracking and management can ripple from the cash register all the way back to raw material production, creating inventory shortages at any stage of the value chain. The resulting out-of-stock events can mean lost business. The Internet promises to increase business efficiency by reducing reporting delays and increasing reporting accuracy. Speed is clearly the imperative for the value chain.

See reference [10]
Sainsbury's, the second-largest retailer in the United Kingdom, recently launched a value chain initiative to provide tighter collaboration with suppliers, including Proctor & Gamble, Nestle', and Oscar-Mayer. Other benefits include higher profits, and better customer service. The new system will eventually allow thousands of Sainsbury's suppliers to work with buyers in planning, executing, and managing successful product promotions.

Figure 2.5 Sainsbury's supermarkets

Unfortunately, speed can be costly. Today, approximately 50,000 businesses exchange business documents such as orders and invoices with their trading partners through a standard communication and a content protocol called Electronic Data Interchange (EDI). Most EDI implementations use leased lines or Value Added Networks that require significant integration for each trading partner. Network design, installation, and administration can be costly in terms of hardware, software, and staff. In fact, these costs are the key reason that EDI is most widely deployed only in larger companies.

---

7 See reference [13]
Moving forward, all companies will be able to take advantage of value chain integration through the low cost of the Internet. Open standards for electronic document exchange will allow all companies to become Internet trading partners and function as suppliers, consumers, or both in this business-to-business electronic commerce. This integrated trading will tighten relationships between businesses while offering them greater choices in supplier selection. For example, Sainsbury's supermarkets launch a value chain initiative to collaborate with suppliers (see figure 2.5).
2.3 E-Commerce and Common Commerce

Although electronic-commerce is a very new technology, it still employs the same features of ordinary commercial solution such as marketing, sales and order fulfillment functions. However, the degree of value chain integration is improved in all of these functions.

➤ Universal Connectivity: Providing ubiquitous access to the system through a common interface. A business can be contacted by more customers than in the common business commerce.

➤ Marketing: Internet increases marketing channel thus publicizing the products and services to all customers around the world.

➤ Sales: Electronic commerce generates more orders for the products. This means the sales can be increased.

➤ Payment: Electronic commerce enables credit card and other payments along with electronic fund transfers, which provide faster and easier financial service for business and its partners.
➤ Fulfillment: The process of the order and delivering of the product are faster and smoother. After taking an order, the business can contact suppliers using electronic process and allow suppliers to send the products directly to customers, which will reduce the time of the transaction processing.

➤ Support: Businesses’ websites provide pre and post sale assistance to generate more sales, and reduce end-consumers’ problems.

➤ Inventory Management: Reporting and maintaining inventory status to the minimum level, which helps the business to develop the just-in-time inventory system.

➤ Secure Communications: Electronic mail makes a company faster, more efficient, and provides communication that is more reliable with customers and partners.
2.4 Benefits of E-Commerce

Electronic commerce seems to be the most powerful tool for doing business today. Every organization, large or small, acknowledges the advantage of using electronic commerce. Without e-commerce tools, a business will be separated from its partners and its customers. The following is some goods reasons why a business should apply an electronic commerce strategy now.

➢ Lowers entry costs: Compared to the ordinary commercial business, electronic commerce provides lower administration costs. By using an online and automatic system, each organization can reduce the number of entry-level employees, because customers can connect to the company’s system directly without any trouble.

➢ Reduces transaction costs: Electronic commerce makes it possible to do the business without the second or the third party companies. Any customer can order a product quickly through a manufacturer’s website, which will reduce the price and the shipping cost of the product.
Provides access to global market: Because Internet and the World Wide Web provides a whole picture of globalization, when a business presents its products through the Internet it means that it also exhibits them to the world as well. Customers around the world can visit and order the products and services without geographic boundaries anymore.

Provides online distribution: The electronic commerce offers a new marketing distribution channel to every business. A company can use this new marketing channel as a backup channel or a major channel for doing marketing strategy in today's business.

Secures market share: Customers will find it easier to contact e-business companies than companies without an e-business support. The Internet also changes the narrower local market to the broader global market. Therefore, the opportunities to increase market share and revenue of those companies will rise.
2.5 Issues in Implementing E-Commerce

Although it is simple to describe electronic commerce's benefits, it is not nearly as easy to develop and deploy commerce systems. Companies can face significant implementation issues:

➤ Cost: Electronic commerce requires significant investments in new technologies that can touch many of a company's core business processes. As with all major business systems, electronic commerce systems require significant investment in hardware, software, staffing, and training. Businesses need comprehensive solutions with greater ease of use to help foster cost-effective deployment.

➤ Value: Businesses want to know that their investments in electronic commerce systems will produce a return. Business objectives such as lead generation, business-process automation, and cost reduction must be met. Systems used to reach these goals need to be flexible enough to change when the business changes.
Security: The Internet provides universal access, but companies must protect their assets against accidental or malicious misuse. System security, however, must not create prohibitive complexity or reduce flexibility. Customer information also needs to be protected from internal and external misuse. Privacy systems should safeguard the personal information critical to building sites that satisfy customer and business needs.

Leverage Existing Systems: Most companies already use information technology to conduct business in non-Internet environments, such as order management, billing, inventory, distribution, and customer service. The Internet represents an alternative and complementary way to do business, but it is imperative that electronic commerce systems integrate existing systems in a manner that avoids duplicating functionality and maintains usability, performance, and reliability.

Interoperability: When systems from two or more businesses are able to exchange documents without manual intervention, businesses achieve cost reduction, improved performance, and more dynamic value chains.
Failing to address any of these issues can spell failure for a system's implementation effort. The business commerce strategy must be designed to address all of these issues to help customers achieve the benefits of electronic commerce.

2.6 Requirements of an E-Commerce Site

The e-commerce process starts with a customer link to the merchant website via the Internet through the Internet Service Provider (ISP). He or she, then, selects favored items, and puts them in an electronic shopping cart, similar to shopping at any superstore in the real world. After completing the shopping process, the customer goes to an available cashier to begin the checkout process, which includes payment, packaging, and shipping processes. Figure 2.6 shows the steps of e-commerce processes.
To complete these processes perfectly, each electronic commerce website needs to implement sub-procedures like in the real-world situation. The following are the requirements before setting up an e-commerce website.

- Promotional and Searchable Content: In the real-world situation, when a company wants to promote special products, they will put these products where customers can see them easily. Sometimes, the company also puts a banner in front of the store as well. On merchant
websites, usually an e-commerce business will put the special promotion products on the first page, which is called a homepage. In addition, to help the customer locate the specific product, some businesses hang a little sign to show where a specific product is. In the e-business website, the searcher needs to be placed on the website to do this job.

- **Shopping Cart:** The concept of a shopping cart makes the shopping more convenient to the customer. He or she will put selected products in a shopping cart then continue shopping for the other products instead of checking them out immediately. Customers will use more time to shop, which increases the opportunity to sell more products. The sales and revenue of a company will be increased.

- **Shopper Management:** The shopper management is a tool that permits a customer to handle the product in a shopping cart simply. This tool allows a customer to increase, reduce, or remove the quantity of each individual product. It also permits a customer to select payment, and shipping methods.
Flexible Support for Payment: Online banking and financial service makes it possible for electronic commerce businesses to receive the payments of the orders from the Internet customers online. They will have many choices of payment, such as using a major credit card, or an online checking account. The merchant companies, then, can send products to customers without waiting for customers' checks to be sent and cleared.

Secure Transactions: The most important feature for online business transactions is security. Without a protection system, businesses will be challenged with many troubles. The process of decoding and encoding, or encryption process, is a set of techniques to ensure that data and messages can be stored and transmitted securely. This process needs to be implemented on every critical transaction, such as credit approval, ordering, and payment processes to ensure that customers' information will not be intercepted or stolen.
3.1 Five-force Analysis

To start a business, a good manager needs to analyze every factor affected his/her business. Otherwise, the business will encounter many troubles. The five-force analysis model by Michael Porter\(^8\) is a good tool used for examining the five major factors affected a business. This tool will help the manager to implement organization strategic planning.

3.1.1 Entry Barriers

> Economies of Scale: Economies of scale arise from the ability to perform activities differently and more efficiently at large volumes. In the toy wholesaler industry, the profit of intangible costs such as

\(^8\) See reference [17]
advertising and tangible costs such as the storage of a great sales volume is high. The scale sensitivity of activities for the industry is also high. Value activities, such as national advertising, and firm infrastructure are typically more scale-sensitive than other activities, such as sales force operations, because these costs are heavily fixed no matter what the firm's scale is.

- Proprietary product differences: Since wholesalers do not produce any product, but resell it to retailers, wholesalers do not control the differences of the products. Therefore, the proprietary product difference is low in this industry. However, in this industry, many manufacturers sell their products directly to customers. In this situation, manufacturers can take advantages of product differences.

- Brand identity: Usually, the brand identity in this industry is very low. Retailers are free to purchase any product from any wholesaler and any manufacturer. However, in a licensed toy, the brand identity is very high.
Switching costs: Because toy wholesalers do not need to manufacturer products, it is not important to buy any machinery, just a warehouse, which is easy to switch to other products. The switching costs for this industry is low. When a wholesaler owns a warehouse, it can be changed to sell other wholesale products easily.

Capital requirements: Owners of a toy wholesaler do not need much capital, since they can rent a warehouse, and do not need any machines to produce toys. Therefore, the capital requirement is low.

Access to distribution: There are many distribution channels for the toy wholesalers such as participating in national fairs, advertising in national wide areas, and using direct sales. Hence, the access to distribution is quite high.

Absolute cost advantages: There are three minor subcategories in cost advantages.

- Proprietary learning curve: Since there are many different products, a new player will have very high cost of proprietary learning curve.
o Access to necessary inputs: To compete with other competitors, a toy wholesaler needs to contact many manufacturers. Therefore, the costs of access to necessary inputs are high.

o Proprietary low-cost product design: Since wholesalers do not need to design any product at all, the cost of product design is zero.

➢ Government policy: There is a moderate effect of government policy to toy wholesalers. Nevertheless, for the manufacturers, the government policy might have a high affect. If a product is recalled, the manufacturers will pay for the cost of the recallation, not the wholesalers.

➢ Expected retaliation: The payback per unit of wholesale business is low, but the total units sold of wholesalers are very high. Consequently, the expected retaliation for this business is high.

➢ Conclusion: The entry barriers for this business are quite low. New players may find it is easy to enter into this business. Moreover, they may find it is not difficult to switch to other wholesale businesses. Table 3.1 shows the scale of each factor of entry barriers.
<table>
<thead>
<tr>
<th>Factor</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Economic of Scale</td>
<td>Economic ▲ Uneconomic</td>
</tr>
<tr>
<td>2. Proprietary product</td>
<td>Different ▲ Indifferent</td>
</tr>
<tr>
<td>differences</td>
<td></td>
</tr>
<tr>
<td>3. Brand identity</td>
<td>High ▲ Low</td>
</tr>
<tr>
<td>4. Switching Costs</td>
<td>Low ▲ High</td>
</tr>
<tr>
<td>5. Capital requirements</td>
<td>Low ▲ High</td>
</tr>
<tr>
<td>6. Access to distribution</td>
<td>Open ▲ Close</td>
</tr>
<tr>
<td>7. Absolute cost advantages</td>
<td>Low ▲ High</td>
</tr>
<tr>
<td>8. Government policy</td>
<td>Open ▲ Close</td>
</tr>
<tr>
<td>9. Expected retaliation</td>
<td>High ▲ Low</td>
</tr>
<tr>
<td>Conclusion</td>
<td>Weak ▲ Strong</td>
</tr>
</tbody>
</table>

Table 3.1 Entry Barriers of the Toy Wholesaler Industry.

3.1.2 *Determinants of Supplier Power*

- Differentiation of inputs: There are many kinds of toys in the market. Usually a manufacturer will not produce only one major product, but will produce about three to fifteen different types of toys. All of these products might use many different inputs such as wood, metal, paper and plastic. This means there are many differentiations of inputs in the toy industry.
Switching costs of suppliers and firms in the industry:
The switching costs of suppliers in the toy industry are quite high since manufacturers need to invest in factories and machines. When they need to switch to other products, they might need more capital to change the production line. Note also that, since toy businesses change often, manufacturers also need to have a new product design and minor changes in the production line to produce new kinds of toys.

Presence of substitute inputs: In general, toy manufacturers will set up their branches in third-world countries such as China, Taiwan and Mexico, where the costs of production, such as labor and material, are low and controllable. Therefore, the presence of substitute input is high in this business.

Supplier concentration: Because of the baby boom echo, it is estimated that the growth rates of toy industry will expand quickly in the next few years, which will attract a lot more suppliers. The current suppliers will increase a concentration to this industry.
➢ Importance of volume to supplier: For the toy industry, manufacturers require volume quantities of a purchase order. The larger the volume is, the lower the costs of production for the manufacturers are.

➢ Cost relative to total purchases in the industry: Considering that there are many competitors in the toy market, every manufacturer needs to keep the costs of production low. Therefore, the relation of cost to total purchases is very high.

➢ Impact of inputs on cost or differentiation: The impact of inputs cost and differentiation are high because every manufacturer needs to redesign its product frequently. Customers will discard uninteresting or identical products.

➢ Threat of forward integration relative to threat of backward integration by firms in the industry: In general, the manufacturer attempts to contact retailers directly. Many manufacturers have their own marketing departments and salesperson teams. Because the popularity of a toy changes fast and often, manufacturers require contact directory to resellers when they have a new product, which will increase the speed of distribution of their products.
Conclusion: The power of suppliers in the toy industry is relatively weak. Therefore, several manufacturers have tried to develop their own marketing channels, which have increased their power. Table 3.1 shows the scale of the supplier power.

<table>
<thead>
<tr>
<th>Factor</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Differentiation of inputs</td>
<td>Different ▲</td>
</tr>
<tr>
<td>2. Switching costs of suppliers and firms in the industry</td>
<td>Low ▲ High</td>
</tr>
<tr>
<td>3. Presence of substitute inputs</td>
<td>Many ▲ Few</td>
</tr>
<tr>
<td>4. Supplier concentration</td>
<td>Low ▲ High</td>
</tr>
<tr>
<td>5. Importance of volume to supplier</td>
<td>Unimportant ▲ Important</td>
</tr>
<tr>
<td>6. Cost relative to total purchases in the industry</td>
<td>Low ▲ High</td>
</tr>
<tr>
<td>7. Impact of inputs on cost or differentiation</td>
<td>Low ▲ High</td>
</tr>
<tr>
<td>8. Threat of forward integration relative to threat of backward integration by firms in the industry</td>
<td>Easy ▲ Difficult</td>
</tr>
<tr>
<td>Conclusion</td>
<td>Strong ▲ Weak</td>
</tr>
</tbody>
</table>

Table 3.2 Determinants of Supplier Power of the Toy Wholesaler Industry.
3.1.3 Rivalry Determinants

- Industry growth: According to the economic specialists, it is predicted that the baby boom echo effect will influence many industries, including the toy industry. Therefore, the expected overall growth in the industry is high.

- Fixed (or storage) costs/value added: Since a warehouse can be rented, it is not necessary to invest in building a warehouse for the wholesale business. Thus, the fixed cost relative to value added in the toy wholesaler business is reasonably low.

- Intermittent overcapacity: Because the fashionable toy will change frequently, manufacturing might be disrupted; usually, wholesalers or resellers will not stock too many products. They will stock only what they can sell.

- Product differences: Because toy wholesalers do not produce the products per se, the products depend on how a wholesaler corresponds with its manufacturers. If a wholesaler can make good arrangements with many manufacturers, it will have many different products.
Brand identity: Ordinarily, the toy market can be classified as two minor markets. The first one is the famous toy, the most well-known toy in the market, such as Pokémon. The second one is the general toy, the commonly accepted toy such as radio-controlled toys. There will be only a few famous toys accepted in each period, but there will be many general toys. For the popular toy, the brand identity is very high.

Switching costs: The switching costs of wholesalers are moderately low since only a warehouse is needed for a wholesaler's business. The warehouse can be used to store other products instead of toys.

Concentration and balance: Usually manufacturers will establish their own market channels. This makes the concentration of toy wholesalers low.

Informational complexity: Because many manufacturers produce many different products, each wholesaler needs to find a lot of information about this business. Therefore, the information is exceptionally complex.

Diversity of competitors: There are not many players in the toy wholesaler industry. However, the major manufacturers typically create their own marketing paths.
Corporate stakes: Normally toy wholesalers do not need to take the risk of the business. When a toy cannot be sold, the manufacturer will carry the risk of lost, not the wholesalers.

Exit barriers: It is very easy for toy wholesalers to exit the business. They can change to trade other products instead of toys. Therefore, the exit barrier is low.

Conclusion: The rivalry of this industry is robust. Although there are not many competitors, manufacturers may play the role of wholesalers as well. This increases competition in the current toy wholesaler industry. Table 3.3 shows the scale for each factor in the rivalry determinant.
<table>
<thead>
<tr>
<th>Factor</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Industry growth</td>
<td>High ▲ Low</td>
</tr>
<tr>
<td>2. Fix (or storage) costs/value added</td>
<td>Low ▲ High</td>
</tr>
<tr>
<td>3. Intermittent overcapacity</td>
<td>High ▲ Low</td>
</tr>
<tr>
<td>4. Product differences</td>
<td>Different ▲ Indifferent</td>
</tr>
<tr>
<td>5. Brand identity</td>
<td>High ▲ Low</td>
</tr>
<tr>
<td>6. Switching Costs</td>
<td>Low ▲ High</td>
</tr>
<tr>
<td>7. Concentration and balance</td>
<td>Unbalance ▲ Balance</td>
</tr>
<tr>
<td>8. Informational complexity</td>
<td>Simple ▲ Complex</td>
</tr>
<tr>
<td>9. Diversity of competitors</td>
<td>Few ▲ Many</td>
</tr>
<tr>
<td>10. Corporate stakes</td>
<td>Low ▲ High</td>
</tr>
<tr>
<td>11. Exit barriers</td>
<td>Easy ▲ Difficult</td>
</tr>
<tr>
<td>Conclusion</td>
<td>Strong ▲ Weak</td>
</tr>
</tbody>
</table>

Table 3.3 Rivalry Determinants of the Toy Wholesaler Industry.

3.1.4 Determinants of Buyer Power

- Bargaining Leverage
  - Buyer concentration versus firm concentration: The firm concentration is higher than buyer concentration since buyers are free to purchase from many manufacturers.
o Buyer volume: The buyer volume is average when compared with the firm volume. A retailer might not need to stock as many products as a wholesaler.

o Buyer switching costs relative to firm switching costs: When comparing the switching costs of the wholesaler with the buyer, the switching costs for a wholesaler is a little bit higher than a retailer because a wholesaler needs more capital than a retailer does.

o Buyer information: There is a lot of information distributed to buyers since manufacturers usually pass the information directly to retailers and customers.

o Ability to backward integrate: In the toy industry, each buyer is free to contract directly with manufacturers.

o Substitute products: Since there are many types of toys, there are many substitute products for any retailer.
➢ Price Sensitivity

  o Price/total purchases: The price of an order for a retailer is high.
  o Product differences: There is a very low difference for retailer products in the customer point of view. The final customers are free to purchase the products from wherever they want.
  o Brand identity: The brand identity for retailer is intermediate.
  o Impact on quality/performance: Since toys are usually sold in retail stores, where there are many consumer products, the impact on quality/performance is low.
  o Buyer profits: The buyer’s profits depend on the types of products. On average, the profit is medium.
  o Decision makers’ incentives: The decision makers’ incentive depends on the popularity of the toy.

➢ Conclusion: The retailer power when compared with wholesaler power is strong (see table 3.4). Retailers, such as Wal-Mart and Toy R Us, can bypass wholesalers and contact manufacturers directly to purchase any product they want.
<table>
<thead>
<tr>
<th>Factor</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Bargaining leverage</td>
<td>High ▲</td>
</tr>
<tr>
<td>2. Price Sensitivity</td>
<td>High ▲</td>
</tr>
<tr>
<td>Conclusion</td>
<td>Strong ▲</td>
</tr>
</tbody>
</table>

Table 3.4 Determinants of Buyer Power of the Toy Wholesaler Industry.

3.1.5 Determinants of Substitution Threat

- Relative price performance of substitutes: The substitute products of toys are, for example, the entertainment and leisure time products such as amusement parks. The relative price performance of substitutes is low. However, it depends on an individual’s point of view.

- Switching costs: The switching costs of substitute product to the customers are low. They can change from buy a toy to go to see a movie or visit amusement parks easily.

- Buyer propensity to substitute: Buyers’ interest in substitution is high.
Conclusion: There are some threats of substitute products for the toy industry (see table 3.5). However, this threat is not, since toys are sometime considered as one of the consumer products. Parents still need to buy a toy for their children birthday.

<table>
<thead>
<tr>
<th>Factor</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Relative price performance of substitutes</td>
<td>Low ▲ High</td>
</tr>
<tr>
<td>2. Switching costs</td>
<td>Low ▲ High</td>
</tr>
<tr>
<td>3. Buyer propensity to substitute</td>
<td>High ▲ Low</td>
</tr>
<tr>
<td>Conclusion</td>
<td>Strong ▲ Weak</td>
</tr>
</tbody>
</table>

Table 3.5 Determinants of Substitution Threat of the Toy Wholesaler Industry.

3.1.6 Conclusion

After analyzing the five forces, there are only one advantage, table 3.6, to the toy wholesalers in the five-force analysis: the supplier power is low. The others are negative to the toy wholesaler business. Since buyer power and rivalry of existing competitors for toy wholesaler industry are moderately high, toy wholesalers have quite low
power to control the market. The barriers to entry in this business are low, which means it is easy for a new player to enter into this industry. In addition, the substitute threat is also strong.

<table>
<thead>
<tr>
<th>Factor</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Entry barriers</td>
<td>Weak ▲ Strong</td>
</tr>
<tr>
<td>2. Determinants of supplier power</td>
<td>Strong ▲ Weak</td>
</tr>
<tr>
<td>3. Rivalry determinants</td>
<td>Strong ▲ Weak</td>
</tr>
<tr>
<td>4. Determinants of buyer power</td>
<td>Strong ▲ Weak</td>
</tr>
<tr>
<td>5. Determinants of substitution threat</td>
<td>Strong ▲ Weak</td>
</tr>
<tr>
<td>Conclusion</td>
<td>Con. ▲ Pro.</td>
</tr>
</tbody>
</table>

Table 3.6 Five-Force Analysis Conclusion

To survive, grow, and succeed in this industry, a firm needs to find a good long-term strategic plan and a major competitive advantage. E-business seems to be a good answer for this question. E-commerce provides direct marketing, online banking, and value chain trading to a business. It also offers access to global market and online distribution to a firm. Therefore, a toy wholesaler should include e-commerce solution to its strategic plan to succeed in this industry.
3.2 Macrotoys

In this session, a simulated company will be demonstrated. The company will carry out every feature similar to a real business company. This includes the company name, which is a "Macrotoys", the vision, mission and business strategy to succeed in this industry. However, before moving into that section, the major competitive advantage should be discussed to implement the best strategy for this company.

3.2.1 Competitive Advantage: The Differentiation

Before settling any long-term strategy, Macrotoys should consider which three generic strategies it should use to find a major competitive advantage. The toy wholesale business is in a highly competitive industry. Although, there are not many direct competitors, toy manufacturers usually sell the products directly to resellers; this increases concentration of competition. Therefore, the competitive scope of toy wholesale is in a broad target.
Customers typically think that the toy wholesalers are identical. Customers are free to order products from wholesalers, or they can contact manufacturers directly. Since manufacturers make the products, they have better control of cost so they can take advantage of cost leadership. Therefore, the only way to succeed in this industry is using the differentiation strategy.

<table>
<thead>
<tr>
<th>Competitive Scope</th>
<th>Competitive Advantage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Broad Target</td>
<td>Lower Cost</td>
</tr>
<tr>
<td></td>
<td>1. Cost Leadership</td>
</tr>
<tr>
<td>Narrow Target</td>
<td>Differentiation</td>
</tr>
<tr>
<td></td>
<td>2. Differentiation</td>
</tr>
<tr>
<td></td>
<td>3 A. Cost Focus</td>
</tr>
<tr>
<td></td>
<td>3 B. Differentiation</td>
</tr>
<tr>
<td></td>
<td>on Focus</td>
</tr>
</tbody>
</table>

Table 3.7 Three Generic Strategies

In conclusion, Macrotoys should use differentiation in its strategy to accomplish the business goal. Differentiation allows the firm to control a premium price, and sell more of its product at that price. It leads to superior performance if the price premium achieved exceeds any added costs of being unique. If Macrotoys can be unique
at something that is valuable to retailers, it will differentiate itself from competitors and create higher revenues.

3.2.2 Business Assumptions

Since Macrotoys is not a real company, it is hard to tell what the company strategy should be without creating some business boundary assumptions. Therefore, in this section the Macrotoys' initial assumptions will be declared. The following are some preliminary assumptions of Macrotoys.

- Macrotoys is a toy wholesaler limited corporation operating in Southern California. It is operated by a group of managers with an MBA degree.
- The initial budget is approximately $1 million, which does not include any financial support from outside companies.
- The major products are non-licensed toys, about 80% of the total sales. The licensed toy sales are about 20% of the total sales.
The major market at present is in North America—United State and Canada. The company has the potential growth to other countries around the world.

The company has not developed any websites at this time, and does not sell the products through an e-commerce site. The information system infrastructure is currently used only to support generic commercial functions. They have not acquired any hardware or software to support the e-commerce site.

Human resources of the company is sufficient to operate the business functions.

3.2.3 Business Scope

Business scope defines the essential characteristics of a business. Each organizational unit must realize the business scope. In addition, every employee should clearly understand the nature of the business. Realizing the business scope of a company will require collaboration across each job function in the company. The following is the business scope of Macrotoys.

53
➤ Vision: To lead in shaping collaborative, co-managed and cross-functional relationships with partners and customers by using information technology and by moving this whole industry ahead quickly.

➤ Mission: Leverage the Internet and electronic commerce technology to reach as many new potential customers as possible, and to be a national wholesaler of quality toys.

➤ Value: Coordinate business partners, toy manufacturers and toy retailers, together to respond to the end-consumers’ need.

➤ Customers and Markets: The major customers are toy retailers and mini-distributors in the United States and Canada. However, the Internet and electronic commerce provide an opportunity to grow into other markets around the world.

➤ Product and Service: Macrotoys sells a variety of toy products, both licensed and non-licensed, from average to high quality.
Strategic Intent:

- Maintain relationships with business partners to preserve a superior image and increase market share.
- Be a leader in market share and technology and to gain creditability and customer loyalty.

Driving Force: Exploit information technology including Internet and electronic commerce expertise to succeed in a highly competitive market with superior satisfactory collaboration with business associates.

3.2.4 Strategy

A company usually separates business functions into financial, operating, marketing, human-resource and information management functions. Each function might have a different mission and responsibility. Therefore, in this section, the strategy of Macrotoys will be aligned into these job functions.

- Financial strategy: Macrotoys has startup capital of $1 million. This might not be enough in the highly competitive business of the toy wholesaler industry.
Macroteys must gain outside capital sources. There are lots of outside financial sources. For the short-term financial strategy, Macroteys can borrow needed funds from banks, financial institutions, or venture capital. For the long-term financial plan, the company can issue bonds and other debt. Once the company meets the requirement to go public, Macroteys should issue common stock to have more outside investors. After the company goes public, there will be more capital resources and more people will know about the company, which increases the company's value.

Operation management strategy: Since the toy business depends on popularity of a fashionable toy, wholesalers need not over-stock old-fashioned toys. Therefore, the concept of the Just-in-Time (JIT) system should be implemented by stock only popular toys, and maintained a low-level of the stock, except during holidays such as between Thanksgiving and New Years. This strategy also reduces the significance of short-term financing by limiting the variable costs of company.
Marketing strategy:

- Product: Both licensed and non-licensed toys are the major products of Macrotoys. The lifecycle of a toy is very short. However, since Macrotoys is a wholesaler, not a manufacturer, Macrotoys need not to be concerned for the lifecycle of any toy. The branding policy also has no effect the toy wholesaler industry as well. The variety of the products seems to be a major focus for product strategy in this industry.

- Price: Because Macrotoys' major competitive strategy is differentiation. Macrotoys cannot be a price leader. Therefore, the profit goals are selling high-quality products and obtaining premium prices.

- Promotion: Macrotoys should participate in the major toy industry fair. The national media should be used. However, the television or radio, national broadcasting might not increase brand awareness for this industry. Macrotoys should advertise in any other media, such as the Internet. In addition, Macrotoys will
create a good image by establishing a donation fund for nonprofit public organization.

- Place: The major distribution channel is direct distribution to customers. Macrotoys will sell toys on the Internet. Salespeople and representatives are the other distribution channel. Macrotoys should employ the selective member distribution channels, which authorize some salespeople to take care of a marketing area.

➢ Human-resource management strategy: Employee loyalty is the key strategy in human-resource management. Managers must know what their employees' needs are. Macrotoys' organizational structure should be a flat organization that will provide good communication across levels of managers and employees.

➢ Information management strategy: Macrotoys will use Internet and information technology as a lethal weapon. Information technology will be developed to link the organization throughout with outside partners. This includes developing infrastructure for each organization level, and creating informational interchange channels for partners.
CHAPTER FOUR: ACTIVE SERVER PAGES TECHNOLOGY

4.1 Overview

The major technology used in the implementation process of this project is Active Server Pages technology, or ASP. ASP is simply a way to create a dynamic web application provided by Microsoft. ASP technology is usually implemented in the Microsoft environment.

Web programmers or web administrators generally develop, test, and debug ASP source programs using an Integrated Development Environment (IDE) such as Microsoft Visual InterDev or other third-party IDE. The ASP scripts are, usually, linked to a Database Management System (DBMS), such as Microsoft Access, Microsoft SQL Server, or Oracle using older database connection technology, such as Data Source Name (DSN) through Open Database Connectivity (ODBC), or newer database connection technology, ActiveX Data Object (ADO).
Finally, a web server, normally Microsoft Internet Information Server (IIS) that is running on Microsoft Windows NT or Microsoft Windows 2000 operating system, is where the ASP scripts are hosted and run before a client web browser requesting an ASP website.

4.2 What is ASP?

Microsoft® Active Server Pages or ASP is an open, compile-free server-side script (small-embedded programs) technology used to create dynamic, interactive, and powerful Web applications. These applications are processed on a Microsoft Web server, Microsoft Internet Information Server (IIS), before the pages are sent to the user's browser. The result, which is the HTML stream, can be delivered to almost any browser. An ASP is somewhat similar to a server-side include (SSI), or a common gateway interface (CGI) application in that all involve programs that run on the server, usually tailoring a page for the user.
ASP is similar in concept to Allaire's Cold Fusion and Netscape's server-side JavaScript, and is designed as a convenient alternative to conventional CGI scripting using Perl or C scripts. It is a great fit for a wide array of workgroup applications, including online human resources and accounting applications.

An ASP combines Extensible Markup Language (XML), Component Object Model (COM), and Hypertext Markup Language (HTML), scripts, and reusable ActiveX server components to create powerful interactive Websites. An ASP script can be written in VBScript or JScript in an HTML file or by using ActiveX Data Objects (ADO) program statements in the HTML file. An ASP is recommended by Microsoft to use the server-side ASP rather than a client-side script, where there is actually a choice because the server-side script will result in an easily displayable HTML page. Client-side scripts, for example Java Scripts, may not work as intended on older browsers.
4.3 How ASP works

The process (as shown in figure 4.1) of requesting an ASP starts with a user bringing up a Website where the default page has the extension .asp. The user's browser requests the ASP file from the Web server. Typically, server-side scripts run in the Web server when a browser requests an .asp file. The ASP script is called by the Web server, which processes the requested file from top to bottom and executes any script commands.

The ASP may use input received as the result of the user's request from the previous page to access data in a database and then builds or customizes the page on the fly before sending it to the requestor's browser.
It is possible to extend ASP scripts using COM components and Extensible Markup Language (XML). COM is a way to manage transactional components extended scripting capabilities by providing a compact, reusable, and secure means of gaining access to information. XML is a meta-markup language that provides a format to actually describe the structured data content of the element by using a set of tags.
4.4 Benefits of ASP

Because the ASP script runs on the server, the Web server does all of the processing and standard HTML pages can be generated and sent to the browser. This means that Web pages are limited only by what the web server supports. It does not care what web browser is being used. Another benefit of having ASP script reside on the server is that the user cannot "view source" on the original script and code. Instead, the user sees only the generated HTML as well as non-HTML content, such as XML, on the pages that are being viewed.

The following are some benefits of using ASP script:

> Language Independence: Two of the most common scripting languages are supported right out of the box: VBScript and JScript™, which may not be supported by a client's browser. For example, Netscape Navigator supports only Java script language. Support for other scripting languages, such as PERL, is also available with third-party add-in software. Programmers can write a web application in any preferred language without concerning for compatibility of the clients' browsers.
Dynamic Application: ASP provides the new way of dynamic web application programming. There are many client-side dynamic web application programs such as Java applets, Dynamic HTML, ActiveX controls, all of which are browser specific. Since the ASP script runs at the web server, it enables the programmers to program dynamic web applications browser-independently, without recourse to client-side programming features.

Real-time information: Businesses may need to provide up-to-date information that does not reside on their partners' browsers, such as the current price of a product, or the current exchange rate. ASP can provide this information to the client's browser with the most recent data that resides on the web or other database servers.

Reduce loading time: It often makes for quicker loading times than with client-side dynamic web technologies such as Java applets or ActiveX controls because client-side dynamic web technologies are downloaded in specific application program formats which are bigger than the HTML file format and then run at the client's computer. For the ASP, however, users are actually only downloading a page of HTML at the end.
Secure source code: ASP provides improved security measures since the programming code can never be viewed from the client browser. Users will see the source file in HTML format form. None of the ASP programming code will be shown to the client's browser.

Data-Driven Web Pages: IIS 4.0 and ASP make it easy to access data and put it on a Web page. Programmer can simply display data from an ODBC-compliant database, or use ASP to make decisions about what to display on Web pages.

Transactions on the Web: Both Client and server computers have recently become much more powerful, making multitier client/server computing possible. By arranging functionality into neat packages called "transactions," it is possible to involve more than just the client and server computers in the process. Thus, the key to using the Internet for major application development is to support these "transactions." This is not easy to do because "transactions" typically involve processes on multiple computers and a method is needed to coordinate all those computers safely and reliably.
- **Simplified coding**: ASP pages and scripts enable the company's existing developers to create the bulk of the functionality required. At the same time, interfaces to specialized software are built using conventional Windows technology. IIS ties all the pieces together.

- **Reduced interconnection hardware**: The Internet supplies ready-made connectivity.

- **Simplified maintenance**: ASP and components work cooperatively, enabling Webmaster to script or program each portion of the solution using the appropriate tool.

- **Quicker prototyping**: The combination of scripting and Web pages enables the development team to deliver prototypes to the management team on a regular basis. Changes are integrated almost immediately.

- **Isolated Processes**: An isolated application runs in its own memory space on the server. This memory is available only to that specific application; no other application can access that memory space. If a problem develops in the application, there is no way for it to "invade" the memory space of other applications, or of any other software running on the server. Isolation prevents one application from dragging down other
applications, or the server, if it crashes. It is a little like having multiple computers—virtual machines—running on one server.
CHAPTER FIVE: DEVELOPMENT PROCESS

5.1 Planning

Before any project begins, a good project management planning should be considered. This project uses Microsoft Project 2000 Beta version, which is project management software. The project is divided into five planning parts, which are the study phase, systems analysis, coding, documentation, and presentation. Table 5.1 shows the Gantt chart of this project, which includes a timeline, start dates, and end dates of each task. In this section, each task and its subtasks will be discussed.

Study Phase: This phase is separated into two subtasks. Before every project can be launched, important data must be gathered. This information will be used to study the feasibility of the project. If the project is impossible, the process will be stopped here before any more resources are put in the project. Otherwise, all of those resources will be wasted since the project cannot continue. Therefore, this phase is very important.
<table>
<thead>
<tr>
<th>ID</th>
<th>Task Name</th>
<th>Duration</th>
<th>Start</th>
<th>Finish</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>B2B Project</td>
<td>118.63 days</td>
<td>Mon 1/17/00</td>
<td>Thu 5/18/00</td>
</tr>
<tr>
<td>2</td>
<td>Study Phase</td>
<td>20 days</td>
<td>Mon 1/17/00</td>
<td>Mon 2/14/00</td>
</tr>
<tr>
<td>3</td>
<td>Gathering Data</td>
<td>10 days</td>
<td>Mon 1/17/00</td>
<td>Mon 1/31/00</td>
</tr>
<tr>
<td>4</td>
<td>Feasibility Study</td>
<td>10 days</td>
<td>Mon 1/31/00</td>
<td>Mon 2/14/00</td>
</tr>
<tr>
<td>5</td>
<td>System Analysis</td>
<td>23 days</td>
<td>Tue 2/15/00</td>
<td>Fri 3/17/00</td>
</tr>
<tr>
<td>6</td>
<td>DFD</td>
<td>7 days</td>
<td>Tue 2/15/00</td>
<td>Thu 2/24/00</td>
</tr>
<tr>
<td>7</td>
<td>Design Database</td>
<td>10 days</td>
<td>Thu 2/24/00</td>
<td>Thu 3/9/00</td>
</tr>
<tr>
<td>8</td>
<td>Network Design</td>
<td>9 days</td>
<td>Thu 3/9/00</td>
<td>Tue 3/14/00</td>
</tr>
<tr>
<td>9</td>
<td>Web Diagram</td>
<td>3 days</td>
<td>Tue 3/14/00</td>
<td>Fri 3/17/00</td>
</tr>
<tr>
<td>10</td>
<td>Coding</td>
<td>52 days</td>
<td>Mon 3/20/00</td>
<td>Mon 5/1/00</td>
</tr>
<tr>
<td>11</td>
<td>ASP coding</td>
<td>35 days</td>
<td>Mon 3/20/00</td>
<td>Wed 4/19/00</td>
</tr>
<tr>
<td>12</td>
<td>Debugging</td>
<td>10 days</td>
<td>Wed 4/19/00</td>
<td>Wed 3/26/00</td>
</tr>
<tr>
<td>13</td>
<td>Testing</td>
<td>7 days</td>
<td>Wed 4/26/00</td>
<td>Mon 5/1/00</td>
</tr>
<tr>
<td>14</td>
<td>Documentation</td>
<td>76 days</td>
<td>Tue 2/29/00</td>
<td>Wed 5/10/00</td>
</tr>
<tr>
<td>15</td>
<td>EC Intro</td>
<td>16 days</td>
<td>Tue 2/29/00</td>
<td>Mon 3/13/00</td>
</tr>
<tr>
<td>16</td>
<td>Toy industry</td>
<td>14 days</td>
<td>Tue 3/14/00</td>
<td>Fri 3/31/00</td>
</tr>
<tr>
<td>17</td>
<td>Technology--ASP</td>
<td>12 days</td>
<td>Sat 4/1/00</td>
<td>Sun 4/9/00</td>
</tr>
<tr>
<td>18</td>
<td>Development Process</td>
<td>10 days</td>
<td>Sun 4/9/00</td>
<td>Sun 4/16/00</td>
</tr>
<tr>
<td>19</td>
<td>Implementation</td>
<td>10 days</td>
<td>Mon 4/17/00</td>
<td>Mon 4/24/00</td>
</tr>
<tr>
<td>20</td>
<td>Conclusion</td>
<td>10 days</td>
<td>Mon 4/24/00</td>
<td>Mon 5/1/00</td>
</tr>
<tr>
<td>21</td>
<td>Appendices</td>
<td>10 days</td>
<td>Mon 5/1/00</td>
<td>Wed 5/10/00</td>
</tr>
<tr>
<td>22</td>
<td>Presentation</td>
<td>11 days</td>
<td>Thu 5/11/00</td>
<td>Thu 5/18/00</td>
</tr>
<tr>
<td>23</td>
<td>Power Point</td>
<td>7 days</td>
<td>Thu 5/11/00</td>
<td>Tue 5/16/00</td>
</tr>
<tr>
<td>24</td>
<td>Prepare for Present</td>
<td>3 days</td>
<td>Tue 5/16/00</td>
<td>Thu 5/18/00</td>
</tr>
<tr>
<td>25</td>
<td>Present</td>
<td>1 day</td>
<td>Thu 5/18/00</td>
<td>Thu 5/18/00</td>
</tr>
</tbody>
</table>
Systems Analysis: For any information management project, the system analysis is the most important phase. Usually system analysts will cooperate with many people from each job function of the project. They must maintain contact with the project manager to acknowledge the project schedule, talk with end users to realize the problem of the system, and must communicate with the development team to create the system. In this project the system analysis phase include the following tasks:

- DFD Design: Data Flow Diagram, or DFD, shows the flow of data passing into or out of the system. The DFD also allows the developer to know which attributes are passing into or out of a specific process.

- Database Design: The database is the most important part of any computer program. In this phase, the entry-relationship diagram will be drawn. The data dictionary, which will provide detail information of each attribute in the table, will also be created. All of this information will be used to design the system.
o Network Design: Since the B2B ECA will be operated on the Internet, which is the networking environment, the network design will show a small picture of the computer and communication connection.

o Web Diagram: The B2B ECA is a web-based application. Therefore, the web diagram will show the connection among all of the pages that are kept in the web server. The web template, which is a prototype of the user interface, will also be designed in this step.

Coding or Programming Process: In this phase, the real application program is generated by using software development tools. For this project, there are three subtasks in this phase. First, the ASP and HTML code will create the B2B ECA based on information from previous phases. Second, after the programming code has been generated, the application will be debugging to find any error in the programming code. Finally, the whole application will be tested in a simulated environment to make sure that the program will operate without any problems in the real situation. All of the subtasks
in this phase are time consuming, especially the debugging process. Since this major step of the project is dealing with programming, testing, and debugging, this document will not present the detail of this process. However, the codes of every web page are provided in Appendix A.

- **Documentation:** This phase can be started simultaneously with the coding process, since all tasks from both phases are independent. In general project, the documentation means creating help files and a user manual. However, for this project, the documentation means creating the project document, which is this document. There are seven chapters and three appendices in this project. Hence, the tasks are separated by all of these chapters.

- **Presentation:** The final task of the project is the presentation. The project is planed to be presented in the middle of May 2000. The Microsoft PowerPoint will be used as the major presentation tools for this project.

However, the project was idled for three months due to the author personal reason (see chapter 6.1).
5.2 Design Phase

Generally, in the designing phase, system analysis team will start by create many diagrams, such as system flow diagram, data flow diagram. These diagrams are used to show the overall picture of the system and are used as the major tools for developing the system. However, in this project, users will run the system through the Internet. This reduces the significance of developing the system flow diagram, since the diagram will look like network diagram, which will be discussed in the following part.

5.2.1 Data Flow Diagram

The first step of the design phase in this project is to create the data flow diagram, or DFD, which will show the flow of data inputs through the system. Usually a system analysis will create the first DFD, which will illustrate the whole picture of the process. This first DFD is called the conceptual level. Then, he or she will divide the conceptual level DFD into many sub diagrams, which demonstrate the sub system in more detail, until each individual sub systems cannot be divided further.
The figure 5.1 illustrates the conceptual level of the B2BECA system. First, customers acquire detailed information about the products via the Internet. After customers are satisfied, they will make the order online. The order transaction will be received by the e-commerce system. Then, it will pass the transaction to the inventory system, which in turn is connected to the supplier’s systems. The inventory system will check the stock and certify that the product is ready to be shipped before it sends a “confirn” transaction back to the e-commerce system. Once the e-commerce system collects the confirm transaction, it will send the product to customers by using the outside carriers, such as UPS or USPS. The process is completed when the customer receives the product from the carrier.

Note: In this project, the inventory system will not be developed; only the e-commerce system will be developed.
Figure 5.2 shows the breakdown of the DFD. In this level, which is called level one, the conceptual level is broken into six sub systems, which are the search information system, the registration system, the order system, the bill and receipt system, the packing system, and the shipping system. All of these systems have the same significance. Although they will deal with different job functions, they will work together to create the whole system. Without any of the subsystems, the application cannot operate.

The following part will discuss each job function of those subsystems. They will also be divided into many smaller subsystems as well.
Figure 5.3 shows the order system in more detail. There are two sub-systems in the order system. The user interface is the system that contacts directly to customers. To make the website more attractive, the user interface generally will be designed by a profession graphic designer before any programming code by a programmer.

The fetch information system is a search engine used to provide information about the product to customers. The good search engine should allow customers to create any ad hoc query depending upon individual need. The speed of the search is also an important issue for the Internet application as well.
Figure 5.4 demonstrates the registration system. Before customers can place any order, first their necessary information, such as company name, address, phone number, and fax number, must be collected. Then the system ought to validate the integrity of this information. The integrity rule and the business rule, such as uniqueness of company name, and user name, will be verified before the information will be sent to the next process. When the information is confirmed, the record information system will add the information to the database, which will be used in the subsequent processes.
Figure 5.5 explains the process of order system. After customers register and create their accounts, they will be allowed to purchase the products. The process starts when customers select a satisfactory product and put them in shopping carts. They will also provide essential information, such as quantity of the product, payment, and shipping information. The order transaction will be verified to make sure that customers provide the correct information. After the information is validated, the record information subsystem will write the information to the company's database. This information will also be used in the inventory system later.
The next system, as shown in figure 5.6, is the bill and receipt system. The system will retrieve required information from the database. It will calculate the price of the order, and add tax and shipping cost before sending this information to the next system. The debit customer’s account system will send the total amount of the order to the customer account and credit this amount to company’s account. These subsystems need to connect with outside companies, such as a credit card company, or a company’s bank. After the bank confirms the credit transaction, the system will print the receipt to customers, and to company’s accounting department. For the e-commerce website these processes will look a little bit different. However, the result will be similar.
The figure 5.7 illustrates the packaging process of the system. This subsystem is, sometimes, included in the inventory system. After customers make the order and the order transaction is verified, an employee, usually from order department, will remove the sold products from the company’s warehouse, put them into a box and wrap up the package. The package, then, will be assigned a tracking number, and a label will be printed. In order to make the application run correctly, the employee must record this information to the database as well. Therefore, the employee’s user interface will be developed and added into the application, as well.
The last process, as demonstrated in figure 5.8, is the shipping system. Once the sold products are packed, the application will send the tracking number to customers, and the package will be shipped to the carrier.

For the e-commerce system, some processes, such as order and receipt processes, or packaging and shipping processes, will be combined together to increase the speed of the order. Sometimes, the subsystems are combined till the user feels like he or she is using a single unique system, not separated subsystems.
5.2.2 Database Design

The database is the heart of every business application. Without a database, it cannot track customer information, calculate sale revenue, and estimate profits. In this part, the entry-relationship, or E-R, diagram will be developed. System analysts and developers must design a normalized database to avoid trouble in the subsequent steps. Then the data dictionary, which shows the detail, such as key fields, variable types, and length of each field in each table, will be created. The E-R diagram and data dictionary will be used as a guideline during the subsequent processes.

5.2.2.1 E-R Diagram: The E-R diagram of this project, as illustrated in figure 5.9, provides the detail information of every important table with its fields. The key fields are stated by using the underscore character. The relationship functions, such as one-to-one and one-to-many relationships, are also provided.
There are five major tables and one many-to-many relationship, which will be converted to a table. The Customers table maintains the customers' information including company name, address, and contract information. The Employees table keeps employees' information, such as social security number, and hire date. However, because the system does not provide salary function, the employee wages
will not be kept. The Orders table keeps each order transaction detail including type of the payment, date of the order, and order status. The Products table provides detail information of each individual product. This information includes stock quantity, and reorder point. The Suppliers table maintains all suppliers information. In the future, this table will be needed to implement the fully function business-to-business application. The last table is the Transact table, which is converted from many-to-many relationship between the orders table and the products table.

5.2.2.2 Data Dictionary: The data dictionary presents each table in the detail. In this project, there are two types of tables in the database. The first one is the basic table, which is shown in Figure 5.9. The basic table is the requirement of the application. Without any of this table, the system will not operate. The second one is the system table. The system tables are created to provide better performance of the application. There are many ways to implement the application without all of these tables.
Therefore, the system tables will not be shown in the E-R diagram, since they are not necessary.

The subsequent tables show the data type of each field in the Microsoft T-SQL format, since the Microsoft SQL server will be used as the major database server for this project. The size, length, of each field will be presented when the data type is VARCHAR only. The primary, foreign and composite keys of each table are shown in the remark field. The T-SQL codes to generate the whole database, including the steps to run each file, the add user command, and the commands used to create all tables, will be provided in Appendix B.
Basic Tables:

CUSTOMERS: Used to keep customers' information.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>CusID</td>
<td>INT</td>
<td></td>
<td>Customer ID</td>
<td>Key Field</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- System generate unique number</td>
<td></td>
</tr>
<tr>
<td>EmpID</td>
<td>INT</td>
<td></td>
<td>Employee ID</td>
<td>Foreign Key</td>
</tr>
<tr>
<td>CompanyName</td>
<td>VARCHAR</td>
<td>40</td>
<td>Customer’s company name</td>
<td></td>
</tr>
<tr>
<td>Address</td>
<td></td>
<td></td>
<td>Composite Field:</td>
<td></td>
</tr>
<tr>
<td>Address1</td>
<td>VARCHAR</td>
<td>30</td>
<td>- Street address</td>
<td></td>
</tr>
<tr>
<td>Address2</td>
<td>VARCHAR</td>
<td>30</td>
<td></td>
<td></td>
</tr>
<tr>
<td>City</td>
<td>VARCHAR</td>
<td>20</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Region</td>
<td>VARCHAR</td>
<td>20</td>
<td>- Region or State</td>
<td>Null</td>
</tr>
<tr>
<td>PostalCode</td>
<td>VARCHAR</td>
<td>10</td>
<td>- Postal or zip code</td>
<td>Null</td>
</tr>
<tr>
<td>Country</td>
<td>VARCHAR</td>
<td>20</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Phone</td>
<td>VARCHAR</td>
<td>15</td>
<td>Customer’s phone number</td>
<td></td>
</tr>
<tr>
<td>Fax</td>
<td>VARCHAR</td>
<td>15</td>
<td>Customer’s fax number</td>
<td></td>
</tr>
<tr>
<td>LogInName</td>
<td>VARCHAR</td>
<td>15</td>
<td>Customer’s login name</td>
<td></td>
</tr>
<tr>
<td>Password</td>
<td>VARCHAR</td>
<td>15</td>
<td>Password</td>
<td></td>
</tr>
<tr>
<td>Website</td>
<td>VARCHAR</td>
<td>40</td>
<td>Customer’s Website</td>
<td></td>
</tr>
<tr>
<td>Contact</td>
<td></td>
<td></td>
<td>Composite Field</td>
<td></td>
</tr>
<tr>
<td>Title</td>
<td>VARCHAR</td>
<td>35</td>
<td>- Contractor title</td>
<td></td>
</tr>
<tr>
<td>FirstName</td>
<td>VARCHAR</td>
<td>30</td>
<td>- Contractor first name</td>
<td></td>
</tr>
<tr>
<td>LastName</td>
<td>VARCHAR</td>
<td>30</td>
<td>- Contractor last name</td>
<td></td>
</tr>
<tr>
<td>Email</td>
<td>VARCHAR</td>
<td>35</td>
<td>- Contractor email</td>
<td></td>
</tr>
<tr>
<td>Direct</td>
<td>VARCHAR</td>
<td>15</td>
<td>- Direct phone line</td>
<td></td>
</tr>
<tr>
<td>Since</td>
<td>DATETIME</td>
<td></td>
<td>The stated date</td>
<td></td>
</tr>
<tr>
<td>Type</td>
<td>VARCHAR</td>
<td>2</td>
<td>Type of customer</td>
<td>Discount %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- GC General Customer</td>
<td>0%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- PC Premium Customer</td>
<td>15%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- BP Business Partner</td>
<td>30%</td>
</tr>
</tbody>
</table>

Table 5.2 Customers Table
EMPLOYEES: Used to keep employees' information.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>EmpID</td>
<td>INT</td>
<td></td>
<td>Employee ID</td>
<td>Key Field</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- System generate unique number</td>
<td></td>
</tr>
<tr>
<td>SSN</td>
<td>VARCHAR</td>
<td>11</td>
<td>Social Security Number</td>
<td></td>
</tr>
<tr>
<td>FirstName</td>
<td>VARCHAR</td>
<td>30</td>
<td>Employee's name</td>
<td></td>
</tr>
<tr>
<td>LastName</td>
<td>VARCHAR</td>
<td>30</td>
<td>Employee's last name</td>
<td></td>
</tr>
<tr>
<td>Address</td>
<td></td>
<td></td>
<td>Composition Field:</td>
<td></td>
</tr>
<tr>
<td>Address1</td>
<td>VARCHAR</td>
<td>30</td>
<td>- Address of employee</td>
<td></td>
</tr>
<tr>
<td>Address2</td>
<td>VARCHAR</td>
<td>30</td>
<td></td>
<td></td>
</tr>
<tr>
<td>City</td>
<td>VARCHAR</td>
<td>20</td>
<td></td>
<td></td>
</tr>
<tr>
<td>State</td>
<td>VARCHAR</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PostalCode</td>
<td>VARCHAR</td>
<td>10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Phone</td>
<td>VARCHAR</td>
<td>15</td>
<td>Employee's phone number</td>
<td></td>
</tr>
<tr>
<td>Email</td>
<td>VARCHAR</td>
<td>35</td>
<td>Employee's email address</td>
<td></td>
</tr>
<tr>
<td>Password</td>
<td>VARCHAR</td>
<td>15</td>
<td>Employee's password</td>
<td></td>
</tr>
<tr>
<td>HireDate</td>
<td>SMALL</td>
<td>DATETIME</td>
<td>Employee's hired date</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.3 Employees Table
ORDERS: Used to keep order information.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>OrderID</td>
<td>INT</td>
<td></td>
<td>Order ID</td>
<td>Key Field</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- System generate unique number</td>
<td></td>
</tr>
<tr>
<td>CusID</td>
<td>INT</td>
<td></td>
<td>Customer ID</td>
<td>Foreign Key</td>
</tr>
<tr>
<td>OrderDate</td>
<td>SMALL</td>
<td>DATETIME</td>
<td>Date of order</td>
<td></td>
</tr>
<tr>
<td>Status</td>
<td>VARCHAR</td>
<td>2</td>
<td>Status of the order</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- IP In Process</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- IT In Transit</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- PS Package is Shipped</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- OS Out of Stock</td>
<td></td>
</tr>
<tr>
<td>Payment</td>
<td>VARCHAR</td>
<td>2</td>
<td>Type of payment</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- AX American Express</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- DS Discover Card</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- DT Direct Transfer</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- MO Money Order</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- MS Master Card</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- VS Visa Card</td>
<td></td>
</tr>
<tr>
<td>AccNum</td>
<td>VARCHAR</td>
<td>16</td>
<td>The account number of customer's payment.</td>
<td></td>
</tr>
<tr>
<td>CarrierID</td>
<td>INT</td>
<td></td>
<td>The carrier ID</td>
<td></td>
</tr>
<tr>
<td>Tracking</td>
<td>VARCHAR</td>
<td>20</td>
<td>Tracking number</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.4 Orders Table

Transact: Links Products and Orders table together.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>OrderID</td>
<td>INT</td>
<td></td>
<td>Order ID</td>
<td>Composite Key</td>
</tr>
<tr>
<td>ProdID</td>
<td>INT</td>
<td></td>
<td>Product ID</td>
<td>Composite Key</td>
</tr>
<tr>
<td>Quantity</td>
<td>INTEGER</td>
<td></td>
<td>Quantity of the product in the order</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.5 Transact Table

89
PRODUCTS: Used to keep all products information.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>ProdID</td>
<td>INT</td>
<td></td>
<td>Product ID</td>
<td>Key Field</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- System generate unique number</td>
<td></td>
</tr>
<tr>
<td>SupID</td>
<td>INT</td>
<td></td>
<td>Supplier ID</td>
<td>Foreign Key</td>
</tr>
<tr>
<td>ProdName</td>
<td>VARCHAR</td>
<td>50</td>
<td>Product name</td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>TEXT</td>
<td></td>
<td>Description of the Product</td>
<td>Memo Field</td>
</tr>
<tr>
<td>PurchasePrice</td>
<td>MONEY</td>
<td></td>
<td>Purchase price per unit</td>
<td></td>
</tr>
<tr>
<td>SalePrice</td>
<td>MONEY</td>
<td></td>
<td>Sale Price Per Unit</td>
<td></td>
</tr>
<tr>
<td>StockQuantity</td>
<td>INTEGER</td>
<td></td>
<td>Current quantity in Stock</td>
<td></td>
</tr>
<tr>
<td>ReorderPoint</td>
<td>INTEGER</td>
<td></td>
<td>Minimum quantity of the Stock before reorder</td>
<td></td>
</tr>
<tr>
<td>SaleUnit</td>
<td>VARCHAR</td>
<td>20</td>
<td>Unit used when sale the product</td>
<td></td>
</tr>
<tr>
<td>PurchaseUnit</td>
<td>VARCHAR</td>
<td>20</td>
<td>Unit used when purchase the product</td>
<td></td>
</tr>
<tr>
<td>Type</td>
<td>VARCHAR</td>
<td>10</td>
<td>Composite Key:</td>
<td>Classify kinds of the product</td>
</tr>
<tr>
<td>Age</td>
<td>VARCHAR</td>
<td>10</td>
<td>(M)ale (F)emale (U)nisex</td>
<td></td>
</tr>
<tr>
<td>Gender</td>
<td>VARCHAR</td>
<td>1</td>
<td>(M)ale (F)emale (U)nisex</td>
<td></td>
</tr>
<tr>
<td>Picture</td>
<td>VARCHAR</td>
<td>20</td>
<td>The file's name of product's picture</td>
<td></td>
</tr>
<tr>
<td>Dimension</td>
<td></td>
<td></td>
<td>Composite Key:</td>
<td>Used for shipping purpose</td>
</tr>
<tr>
<td>Measure</td>
<td></td>
<td></td>
<td>- Measure</td>
<td></td>
</tr>
<tr>
<td>SizeZoneID</td>
<td>INT</td>
<td></td>
<td>- Link to size zone</td>
<td></td>
</tr>
<tr>
<td>Weight</td>
<td>REAL</td>
<td></td>
<td>- Weight in pounds</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.6 Products Table
SUPPLIERS: Used to keep suppliers' information

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>SupID</td>
<td>INT</td>
<td></td>
<td>Supplier ID</td>
<td>Key Field</td>
</tr>
<tr>
<td>CompanyName</td>
<td>VARCHAR</td>
<td>40</td>
<td>Supplier's company name</td>
<td></td>
</tr>
<tr>
<td>Address</td>
<td>VARCHAR</td>
<td>30</td>
<td>Composition Field:</td>
<td></td>
</tr>
<tr>
<td>Address1</td>
<td>VARCHAR</td>
<td>30</td>
<td>- Address of supplier</td>
<td></td>
</tr>
<tr>
<td>Address2</td>
<td>VARCHAR</td>
<td>30</td>
<td></td>
<td></td>
</tr>
<tr>
<td>City</td>
<td>VARCHAR</td>
<td>20</td>
<td>- Region or State</td>
<td></td>
</tr>
<tr>
<td>Region</td>
<td>VARCHAR</td>
<td>20</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PostalCode</td>
<td>VARCHAR</td>
<td>10</td>
<td>- Postal or Zip code</td>
<td></td>
</tr>
<tr>
<td>Country</td>
<td>VARCHAR</td>
<td>20</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Phone</td>
<td>VARCHAR</td>
<td>15</td>
<td>Supplier's phone number</td>
<td></td>
</tr>
<tr>
<td>Fax</td>
<td>VARCHAR</td>
<td>15</td>
<td>Supplier's fax number</td>
<td></td>
</tr>
<tr>
<td>Website</td>
<td>VARCHAR</td>
<td>40</td>
<td>Supplier's Website</td>
<td></td>
</tr>
<tr>
<td>Contact</td>
<td>VARCHAR</td>
<td></td>
<td>Composite Field</td>
<td></td>
</tr>
<tr>
<td>Title</td>
<td>VARCHAR</td>
<td>35</td>
<td>- Contractor title</td>
<td></td>
</tr>
<tr>
<td>FirstName</td>
<td>VARCHAR</td>
<td>30</td>
<td>- Contractor first name</td>
<td></td>
</tr>
<tr>
<td>LastName</td>
<td>VARCHAR</td>
<td>30</td>
<td>- Contractor last name</td>
<td></td>
</tr>
<tr>
<td>Email</td>
<td>VARCHAR</td>
<td>35</td>
<td>- Contractor email</td>
<td></td>
</tr>
<tr>
<td>Direct</td>
<td>VARCHAR</td>
<td>15</td>
<td>- Direct phone line</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.7 Suppliers Table

System Table:

CustomerTypes: Provides types of customers

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>VARCHAR</td>
<td>2</td>
<td>Customer type id</td>
<td>Key Field</td>
</tr>
<tr>
<td>Description</td>
<td>VARCHAR</td>
<td>50</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>Discount</td>
<td>NUMBER</td>
<td></td>
<td>Discount rate (in percentage)</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.8 CustomerTypes Table

91
OrderStatus: Shows detail of order status.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>VARCHAR</td>
<td>2</td>
<td>Order status type id</td>
<td>Key Field</td>
</tr>
<tr>
<td>Description</td>
<td>VARCHAR</td>
<td>30</td>
<td>Description</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.9 OrderStatus Table

PaymentTypes: Gives detail of payment types.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>VARCHAR</td>
<td>2</td>
<td>Payment type id</td>
<td>Key Field</td>
</tr>
<tr>
<td>Description</td>
<td>VARCHAR</td>
<td>30</td>
<td>Description</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.10 PaymentTypes Table

Carriers: Provides carriers information, such as name.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>CarrierID</td>
<td>INT</td>
<td></td>
<td>Carrier ID</td>
<td>Key Field</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- System generate</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>unique number</td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>VARCHAR</td>
<td>50</td>
<td>Description</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.11 Carriers Table
ShippingRates: Provides basic shipping cost.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>CarrierID</td>
<td>INT</td>
<td></td>
<td>Carrier ID</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- System generate unique number</td>
<td>Composite Key</td>
</tr>
<tr>
<td>WeightZoneID</td>
<td>INT</td>
<td></td>
<td>Weight Zone ID</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- System generate unique number</td>
<td>Composite Key</td>
</tr>
<tr>
<td>ShippingCost</td>
<td>REAL</td>
<td></td>
<td>The base price for shipping.</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.12 ShippingRates Table

WeightZoneID: Tells maximum and minimum weight for zones.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>WeightZoneID</td>
<td>INT</td>
<td></td>
<td>Weight Zone ID</td>
<td></td>
</tr>
<tr>
<td>MinWeight</td>
<td>REAL</td>
<td></td>
<td>The minimum weight</td>
<td></td>
</tr>
<tr>
<td>MaxWeight</td>
<td>REAL</td>
<td></td>
<td>The maximum weight</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.13 WeightZoneID Table

States: Provides state name and two characters.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>VARCHAR</td>
<td>2</td>
<td>Order status type id</td>
<td>Key Field</td>
</tr>
<tr>
<td>StateName</td>
<td>VARCHAR</td>
<td>30</td>
<td>Description</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.14 States Table
**StateZones:** Provides charge rate depended on zone.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>StateZoneID</td>
<td>INT</td>
<td></td>
<td>State Zone ID</td>
<td>Key Field</td>
</tr>
<tr>
<td>ChargeRate</td>
<td>REAL</td>
<td></td>
<td>The charge rate in %</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.15 StateZones Table

**SizeZones:** Provides charge rate depended on size.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>SizeZoneID</td>
<td>INT</td>
<td></td>
<td>State Zone ID</td>
<td>Key Field</td>
</tr>
<tr>
<td>Height</td>
<td>REAL</td>
<td></td>
<td>Height of the package</td>
<td></td>
</tr>
<tr>
<td>Width</td>
<td>REAL</td>
<td></td>
<td>Width of the package</td>
<td></td>
</tr>
<tr>
<td>Length</td>
<td>REAL</td>
<td></td>
<td>Length of the package</td>
<td></td>
</tr>
<tr>
<td>ChargeRate</td>
<td>REAL</td>
<td></td>
<td>The charge rate in %</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.16 SizeZones Table

**Subcategories:** Provides subcategories detail.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>SubCatID</td>
<td>INT</td>
<td></td>
<td>Subcategory ID</td>
<td>Key Field</td>
</tr>
<tr>
<td>CatID</td>
<td>INT</td>
<td></td>
<td>Categories ID</td>
<td>Foreign Key</td>
</tr>
<tr>
<td>SubCatName</td>
<td>VARCHAR</td>
<td>30</td>
<td>Name of subcategory</td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>VARCHAR</td>
<td>50</td>
<td>Description</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.17 Subcategories Table
ProductCat: Shows which product belongs to what category.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>ProdID</td>
<td>INT</td>
<td></td>
<td>Product ID</td>
<td>Composite key</td>
</tr>
<tr>
<td>SubCatID</td>
<td>INT</td>
<td></td>
<td>Subcategories ID</td>
<td>Composite key</td>
</tr>
<tr>
<td>StartDate</td>
<td>SMALL DATETIME</td>
<td></td>
<td>The first date to be put in this subcategory</td>
<td></td>
</tr>
<tr>
<td>EndDate</td>
<td>SMALL DATETIME</td>
<td></td>
<td>The last date to be put in this subcategory</td>
<td></td>
</tr>
<tr>
<td>Promotion</td>
<td>Money</td>
<td></td>
<td>Promotion price used during promotion a product</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.18 ProductCat Table

Categories: Presents categories detail.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>CatID</td>
<td>INT</td>
<td></td>
<td>Category ID</td>
<td>Key Field</td>
</tr>
<tr>
<td>CatName</td>
<td>VARCHAR</td>
<td>30</td>
<td>Name of the Category</td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>VARCHAR</td>
<td>50</td>
<td>Description</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.19 Categories Table
WebStats: Used to keep website statistic.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>WebStatsID</td>
<td>INT</td>
<td></td>
<td>Web statistic ID</td>
<td>Key Field</td>
</tr>
<tr>
<td>Browser</td>
<td>VARCHAR</td>
<td>20</td>
<td>Browser type</td>
<td>Allow null</td>
</tr>
<tr>
<td>Version</td>
<td>VARCHAR</td>
<td>20</td>
<td>Browser' version</td>
<td>Allow null</td>
</tr>
<tr>
<td>RemoteAddress</td>
<td>VARCHAR</td>
<td>20</td>
<td>IP address of client</td>
<td>Allow null</td>
</tr>
<tr>
<td>RemoteHost</td>
<td>VARCHAR</td>
<td>50</td>
<td>Host name of client</td>
<td>Allow null</td>
</tr>
<tr>
<td>HttRef</td>
<td>VARCHAR</td>
<td>50</td>
<td>Reference page</td>
<td>Allow null</td>
</tr>
<tr>
<td>SessionID</td>
<td>INT</td>
<td></td>
<td>Session ID of client</td>
<td>Allow null</td>
</tr>
<tr>
<td>ActiveDate</td>
<td>SMALL</td>
<td>DATE</td>
<td>The date when client visit the site</td>
<td>Allow null</td>
</tr>
</tbody>
</table>

Table 5.20 WebStats Table

5.2.3 Network Design

Typically, there are two network architectures in every Internet application. The first one is inside a company network. The Local Area Network, or LAN, is used in today's business company. The LAN provides collaboration of employees and increase productivity of each individual. While the Wide Area Network, or WAN, links the company to its customers and partners.
Figure 5.10 demonstrates both network architectures used in this project. Customers using their own computer connect to the company by using the Internet, which is a WAN connection. They can access information related to making and tracking their order. Customers and outsiders will not be allowed to access critical information. They will be blocked from this access by a firewall. Employees, in
contrast, connect to the system using the inside computer. They might be able to access information, such as employees' salaries, sales revenue, and order transactions. However, they must be authorized from the system before they log in to those critical data.

In a real e-business company, there are at least three servers. The first one is the logon server. The logon server will validate the user login name and password, and keep information of available resources. Each user, including both employees and customers, need to logon to this server first in order to access other resources. The second is the databases server. This server keeps the database of the company and might provide backend applications for data access. Normally these two servers will be suited to the company without using an e-commerce business solution. The web server is only needed when implementing an e-commerce solution.

The backup/recovery and fault-tolerant systems need to be implemented in a real business situation as well. There are many methods to implement the backup and recovery systems. Usually a company needs a tape backup system. The
tape backup device offers the ability to back up and recover data to and from tape medias. Experts also recommend keeping a copy of the important data outside the company as well. Another expensive backup method includes a redundant server. The redundant server will connect directly to the system server. It will keep the same data as in the system server. However, it will not be connected to other computers until the system server is down. In a critical system, a redundant server might be needed when the down time period of the server is unacceptable.

Fault-tolerant describes a computer system or component designed so that in the event that a component fails a backup component or procedure can immediately take its place with no loss of service. The well-known fault-tolerant system is Redundancy Array of Inexpensive Disks, or RAID. RAID can be provided with software, embedded in hardware, or provided by some combination. However, fault-tolerant is not regarded as a replacement of a backup and recovery system.
5.2.4 Web Design

There are two types of websites in this project. The first one is the main website, where customers and outsiders visit. This website is less secure, and will permit everyone to visit. He or she can obtain product information, register in order to buy the products, purchase the products online, track the order they make, and send a comment or email to the company. The second website is the admin website. This site is protected from outsiders and customers because all users who can access this site will be able to modify the significant information. Therefore, only employees are allowed to access this site. Employees can change customer, order, supplier, and product information. They are also permitted to check email online from this website.

5.2.4.1 Main Website: There are twenty-three main pages in the main website. Figure 5.11 shows all of these pages linked together. However, during implementation ten more pages and a directory will be created to make it easier for programming.
Figure 5.11 Main Website Design Diagram

<table>
<thead>
<tr>
<th>Page</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adrot.txt</td>
<td>Advertising rotator: This file is called from Bottom.asp. It includes information such as advertising picture, frequency and time of the picture shown, and the link to a website for each advertising</td>
</tr>
<tr>
<td>See page 157</td>
<td></td>
</tr>
<tr>
<td>Animate.js</td>
<td>Java script created by Microsoft FrontPage used to show animate button on the menu.</td>
</tr>
<tr>
<td>Page</td>
<td>Purpose</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>About.asp</td>
<td>This file contains Macrotoys' information. It will be activated by clicking at &quot;About Us&quot; button on the main menu.</td>
</tr>
<tr>
<td>See page 153</td>
<td></td>
</tr>
<tr>
<td>AddItem.asp</td>
<td>ProductDetail.asp will call this file after a customer clicks the &quot;Buy&quot; button on that page. The item will be added to the shopping cart using this file only.</td>
</tr>
<tr>
<td>See page 155</td>
<td></td>
</tr>
<tr>
<td>AddUser.asp</td>
<td>After CheckRegisterInfo.asp verifies customer information, this file will be called to add a new customer into the Customers Table in Macrotoys' database.</td>
</tr>
<tr>
<td>See page 156</td>
<td></td>
</tr>
<tr>
<td>AdRotatorRedirector.asp</td>
<td>This page will redirect each click on the advertising to the sponsor's website and record statistics of every visit into Application variable.</td>
</tr>
<tr>
<td>See page 159</td>
<td></td>
</tr>
<tr>
<td>Bottom.asp</td>
<td>This is a bottom template. It will be added into every page in this site to show contact information of the developer, advertising, and links to other major pages.</td>
</tr>
<tr>
<td>See page 159</td>
<td></td>
</tr>
<tr>
<td>Page</td>
<td>Purpose</td>
</tr>
<tr>
<td>--------------------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>BrowseBar.asp</td>
<td>A navigation template included into every page. It shows a welcome message and links to product categories.</td>
</tr>
<tr>
<td>See page 166</td>
<td></td>
</tr>
<tr>
<td>Browser.asp</td>
<td>This is the main page after users connect to the system. It will show lists of the products, three products on each list.</td>
</tr>
<tr>
<td>See page 162</td>
<td></td>
</tr>
<tr>
<td>CheckLogIn.asp</td>
<td>After user provide login information, <em>Login Name</em>, and <em>Password</em>, this page will verify this information before allowing the user to login. <em>LogIn.asp</em> will call it.</td>
</tr>
<tr>
<td>See page 168</td>
<td></td>
</tr>
<tr>
<td>CheckOrder.asp</td>
<td>Once users order products, they might want to know the process of their order. This page will provide order information of each customer.</td>
</tr>
<tr>
<td>See page 170</td>
<td></td>
</tr>
<tr>
<td>CheckOut.asp</td>
<td>The checkout process begin when user click checkout on the shopping cart page. This page is the last step of the checkout process. After necessary information is collected, this page will write down this information into the database. The <em>Payment.asp</em> will call this page.</td>
</tr>
<tr>
<td>See page 172</td>
<td></td>
</tr>
<tr>
<td>Page</td>
<td>Purpose</td>
</tr>
<tr>
<td>----------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>CheckRegisterInfo.asp</td>
<td>The major function of this page is to verify the registration information. It will apply the integrity and business rules to the information provide by customers. If the information violates the rules, customers will require reentering that information again. Register.asp will call this page. In addition, it will call Adduser.asp after the data is validated.</td>
</tr>
<tr>
<td>Comment.asp</td>
<td>When a user clicks the &quot;Comment&quot; button on the main menu, he or she will connect to a mail system, which will send the mail to Macrotoys mail server. This page will collect customer's data, such as name, email, and the comment, before sending all information to SendMail.asp.</td>
</tr>
<tr>
<td>DataConn.asp</td>
<td>Whenever a page on this site connects to the database, this file will be called. It contains database connection information. The major purpose of this file is to make it easier to change the database server.</td>
</tr>
<tr>
<td>Page</td>
<td>Purpose</td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Default.asp</td>
<td>The default page is the first page shown to the user. This page provides information about the project, such as project committees, before redirecting each connect to <em>Browser.asp</em>.</td>
</tr>
<tr>
<td>FAQ.asp</td>
<td>Frequently Asked Question about Macrotoys, such as order process, contact information, and so on, is presented on this page. It also provides links that allow customers to send mail about order problems.</td>
</tr>
<tr>
<td>ForgetPassword.asp</td>
<td>Once a customer created an account, he or she might forget his or her own password. This page will ask customers to provide their email address before sending the password and login name back to them again. In addition, after customers fail to login three times, they will be forwarded to this page, as well.</td>
</tr>
<tr>
<td>Global.asp</td>
<td>The Internet Information Server will call this file every time it starts the application. The file contains necessary command to set up Application and Session values.</td>
</tr>
<tr>
<td>Page</td>
<td>Purpose</td>
</tr>
<tr>
<td>--------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Login.asp</td>
<td>If customers have already registered to the Macrotoys’ system, they will click the Login button on the main menu when they revisit the site again. This page will call CheckLogIn.asp to validate and authorize customers.</td>
</tr>
<tr>
<td>Message.asp</td>
<td>This file is a template file that other pages can call to provide active messages back to customers. It needs two variables. The first one is &quot;MSG,&quot; which is the message presented to the user. The second one is &quot;Redirect,&quot; which is the page it will send the user to after he or she clicks on “Click here to continue” button.</td>
</tr>
<tr>
<td>ModifyItem.asp</td>
<td>After a customer puts any product in a shopping cart, he or she is free to modify the product. He or she might want to increase the quantity, or remove any product out of the shopping cart. This page lets customers do this job. The ShoppingCart.asp will call this file when customers want to modify their shopping cart.</td>
</tr>
<tr>
<td>Page</td>
<td>Purpose</td>
</tr>
<tr>
<td>---------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>NonBrowseBar.asp</td>
<td>This is a template file called by any page that does not need Navigator button. This page will work with Top, and Bottom to create a frame for every page.</td>
</tr>
<tr>
<td>See page 192</td>
<td>See page 192</td>
</tr>
<tr>
<td>OrderDetail.asp</td>
<td>CheckOrder.asp will call this file to present the product detail information, such as product ID, name, and quantity, of each order.</td>
</tr>
<tr>
<td>See page 192</td>
<td>See page 192</td>
</tr>
<tr>
<td>Payment.asp</td>
<td>Before a customer can checkout, he or she must provide credit card or banking account information. This information will be use to debit customers account and credit the company account. The Shipping.asp will called this file after receive shipping method information. Payment page then forwards the control to CheckOut.asp to complete the checkout process.</td>
</tr>
<tr>
<td>See page 193</td>
<td>See page 193</td>
</tr>
<tr>
<td>ProductDetail.asp</td>
<td>After customers browse the product category or search for the product, they are able to see the details of each product by clicking on the interesting product. By doing that, they will be forward to this page.</td>
</tr>
<tr>
<td>See page 196</td>
<td>See page 196</td>
</tr>
<tr>
<td>Page</td>
<td>Purpose</td>
</tr>
<tr>
<td>---------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Register.asp</td>
<td>Each visitor will be asked to register before being allowed to checkout. This page will collect required information, such as the company name, address, login name, and password, and then it will verify and record this information by calling CheckRegisterInfo.asp.</td>
</tr>
<tr>
<td>Search.asp</td>
<td>This is a search engine page, which will make it easier for customers to find a product they want. It will find the keyword in the product description field and show all products found to the customer. In case the keyword is not found, it will permit the customer to make a search again for other keywords.</td>
</tr>
<tr>
<td>SendMail.asp</td>
<td>When customers want to send comments, or question about their order, this file will be called to take care this job. After it sends the mail to Macrotoys' mail server, It will call Message.asp to show the visitor that the mail was sent.</td>
</tr>
<tr>
<td>Page</td>
<td>Purpose</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SendPassword.asp</td>
<td>The ForgetPassword.asp will call this file after validating the customer’s email address. This file sends the user’s login name and password back to the customer again.</td>
</tr>
<tr>
<td>See page 209</td>
<td></td>
</tr>
<tr>
<td>Shipping.asp</td>
<td>This is the first step of the checking out process. The page will let customers select a shipping method and calculate the total price of each order before forwarding the control to Payment.asp. Customers are free to select any carrier for their shipping.</td>
</tr>
<tr>
<td>See page 210</td>
<td></td>
</tr>
<tr>
<td>ShoppingCart.asp</td>
<td>The Shopping cart shows all item the customer add in it. This page also permits customers to modify any item in the cart by calling ModifyItem.asp. After, customers complete the orders, they will be allowed to checkout by clicking at the “Check Out” button.</td>
</tr>
<tr>
<td>See page 215</td>
<td></td>
</tr>
<tr>
<td>Template.asp</td>
<td>This file is used to create other pages during the development process. However, it will not be used during the runtime.</td>
</tr>
<tr>
<td>See page 216</td>
<td></td>
</tr>
<tr>
<td>Page</td>
<td>Purpose</td>
</tr>
<tr>
<td>--------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Top.asp</td>
<td>A top template works together with BrowseBar.asp, and Bottom.asp to create a web frame for any other web page. This page also provides necessary buttons and linking tools.</td>
</tr>
<tr>
<td>Image&lt;DIR&gt;</td>
<td>This directory contains every picture used in the project. It also holds a subdirectory called Product, where all product pictures are kept.</td>
</tr>
</tbody>
</table>

Table 5.21 Main Website File Detail

5.2.4.2 Admin Website: For security purposes, this site should not be put in the same directory or as a subdirectory of the main website. It is also suggested that the admin website should use an IP port number higher then 1024. The port number 80 is not recommended. A virtual directory called "image," which is mapped to "image" directory in the main website, is also suggested. These requirements are out of the scope of this project. Only the administrator can do all of these requirements on the web server.
There are twenty-eight main pages on this website as shown in figure 5.12. However ten more pages and a virtual directory will be created to increase the admin website performance. Some of these pages, which will be preceded with "Emp," might operate in the same job function as in the main website. However, the codes are a little bit different. To reduce the hard drive space, a virtual directory, which is mapped to the "image" directory in the main website, should be created. Otherwise, an actual directory must be created, and all image files must be copied into this directory.

Figure 5.12 Admin Website Design Diagram
<table>
<thead>
<tr>
<th>Page</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddCat.asp</td>
<td>This page will add category to the database. Users will not see it.</td>
</tr>
<tr>
<td>See page 222</td>
<td></td>
</tr>
<tr>
<td>AddProd.asp</td>
<td>This page will add product into the database. Users will not see it.</td>
</tr>
<tr>
<td>See page 222</td>
<td></td>
</tr>
<tr>
<td>AddProdCat.asp</td>
<td>Employees will used this file to put a product into any subcategory.</td>
</tr>
<tr>
<td>See page 226</td>
<td></td>
</tr>
<tr>
<td>AddSubCat.asp</td>
<td>This file will add subcategory into the database. Users will not see it.</td>
</tr>
<tr>
<td>See page 227</td>
<td></td>
</tr>
<tr>
<td>AddSup.asp</td>
<td>This file will add supplier information into the database. Users will not see it.</td>
</tr>
<tr>
<td>See page 228</td>
<td></td>
</tr>
<tr>
<td>Adrot.txt</td>
<td>Advertising rotator: This file is called from Bottom.asp. It includes information such as advertising picture, frequency and time of the picture shown, and the link to a website for each advertising</td>
</tr>
<tr>
<td>See page 231</td>
<td></td>
</tr>
<tr>
<td>Category.asp</td>
<td>This page presents all categories, and lets employees view detail of each category by selecting on an individual category.</td>
</tr>
<tr>
<td>See page 232</td>
<td></td>
</tr>
<tr>
<td>Page</td>
<td>Purpose</td>
</tr>
<tr>
<td>---------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>CatDetail.asp</td>
<td>After employees select a specific category, this page will be called to show the category detail. It presents all subcategories in the selected category, and allows the employee to select individual subcategories. It will permit an employee to add more subcategories into this specific category, as well.</td>
</tr>
<tr>
<td>ChangePassword.asp</td>
<td>When employees snap on the Change Password link, this page will be presented. It will ask the employee to provide the old password, new password and a verification of the new password.</td>
</tr>
<tr>
<td>CheckEmp.asp</td>
<td>The EmpLogIn.asp page will call this file to validate the employees’ login. If the password, or employee email address does not match the information in the Employees Table, they will not be allowed to login and be asked to provide new email and password again.</td>
</tr>
<tr>
<td>Page</td>
<td>Purpose</td>
</tr>
<tr>
<td>-------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>CustDetail.asp</td>
<td>After employees snap on the link in Customer.asp page, they will be leaded to this page, which will show the customer detail. It also allows employees to change a customer's representative.</td>
</tr>
<tr>
<td>Customer.asp</td>
<td>This file will show all customers that relate to a current employee. A search engine, which will be enable employees to find specific customers, is also shown in this page.</td>
</tr>
<tr>
<td>DataConn.asp</td>
<td>This file contains database connection information. The major purpose of this file is to make it easier to change the database server.</td>
</tr>
<tr>
<td>Default.asp</td>
<td>This page provides information about the project, such as project committees, before redirect each connect to EmpLogIn.asp.</td>
</tr>
</tbody>
</table>

See page 238
See page 240
See page 243
See page 244
<table>
<thead>
<tr>
<th>Page</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>EmpLogIn.asp</td>
<td>This page is used to log employees into the system. For security purposes, there is a code used to forward each connect to this page at the heard of any other page in this site in case an unauthorized person tries to reach those pages directly.</td>
</tr>
<tr>
<td>EmpMessage.asp</td>
<td>This file has the same function as the Message.asp in the main website.</td>
</tr>
<tr>
<td>EmpOrderDetail.asp</td>
<td>Order.asp will call this file to present the customer and product detail information, such as customer company name, contact information, product ID, product name, and quantity, of each order.</td>
</tr>
<tr>
<td>EmpProductDetail.asp</td>
<td>After an employee clicks the link on Product.asp, this page will show the detail of the selected product. It also authorizes the employee to change some product fields.</td>
</tr>
<tr>
<td>Page</td>
<td>Purpose</td>
</tr>
<tr>
<td>-----------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>EmpSendMail.asp</td>
<td>This file will be called in order to send employee's mail to any other mail server.</td>
</tr>
<tr>
<td>See page 256</td>
<td></td>
</tr>
<tr>
<td>Footer.asp</td>
<td>The function of this file is the same as Bottom.asp in the main website.</td>
</tr>
<tr>
<td>See page 256</td>
<td></td>
</tr>
<tr>
<td>Global.asp</td>
<td>The Internet Information Server will call this file every time it starts the application. The file contains necessary command to set up Application and Session values.</td>
</tr>
<tr>
<td>See page 259</td>
<td></td>
</tr>
<tr>
<td>Head.asp</td>
<td>The function of this page is the same as Top.asp in the main website.</td>
</tr>
<tr>
<td>See page 260</td>
<td></td>
</tr>
<tr>
<td>MailConn.asp</td>
<td>A header file, which will create mail server connection. This file must be put on every page that calls mail server service.</td>
</tr>
<tr>
<td>See page 262</td>
<td></td>
</tr>
<tr>
<td>MailMan.asp</td>
<td>The MainMenu.asp and ViewMessage.asp will request this page in order to compose a new mail or reply the old one. This page, then, sends the control to NewMail.asp</td>
</tr>
<tr>
<td>See page 263</td>
<td></td>
</tr>
<tr>
<td>Page</td>
<td>Purpose</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>MainMenu.asp</td>
<td>At the main menu, each employee will be permitted to check his or her emails. Then, he or she is able to reply or compose messages.</td>
</tr>
<tr>
<td>NavigatorBar.asp</td>
<td>This page takes care of the same job function as BrowserBar.asp in the main website.</td>
</tr>
<tr>
<td>NewMail.asp</td>
<td>When an employee wants to reply or compose mail, he or she will be led to this file in order to write the mail. After he or she finishes, this file will call the EmpSendMail.asp to send the mail to the mail server.</td>
</tr>
<tr>
<td>NonNavigatorBar.asp</td>
<td>This page has the same function as NonBrowseBar.asp in the main website.</td>
</tr>
<tr>
<td>Order.asp</td>
<td>Each employee will be able to search specific order by using this page. It, then, calls the EmpOrderDetail.asp, when the employee clicks on the link on this page.</td>
</tr>
<tr>
<td>Page</td>
<td>Purpose</td>
</tr>
<tr>
<td>------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Product.asp</td>
<td>When an employee wants to know that what a specific supplier sells, this page will be called. It shows the list of all product of the specific supplier. The employee then able to select a specific product again to view the product detail. To do that, the EmpProductDetail.asp will be called.</td>
</tr>
<tr>
<td>RecProd.asp</td>
<td>After product information is validated, this program will record it to the database.</td>
</tr>
<tr>
<td>RecSup.asp</td>
<td>After supplier information is validated, this program will record it to the database.</td>
</tr>
<tr>
<td>Stats.asp</td>
<td>This program will show statistics of the website. Employee will know how many customers connect to the site.</td>
</tr>
<tr>
<td>SubCat.asp</td>
<td>This file requested by CategoryDetail.asp shows the list of all subcategories in a specific category. It will request SubCatDetail.asp when an employee clicks on the link on this page.</td>
</tr>
<tr>
<td>Page</td>
<td>Purpose</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SubCatDetail.asp</td>
<td>This program will search the database and show the category detail to users.</td>
</tr>
<tr>
<td>See page 279</td>
<td></td>
</tr>
<tr>
<td>SubCatUpdate.asp</td>
<td>This program will update subcategory information to the database.</td>
</tr>
<tr>
<td>See page 281</td>
<td></td>
</tr>
<tr>
<td>Supplier.asp</td>
<td>This page shows the list of all suppliers. It also provides links to view the suppliers' details. In order to add a product, employees need to connect to this page first.</td>
</tr>
<tr>
<td>See page 285</td>
<td></td>
</tr>
<tr>
<td>SupDetail.asp</td>
<td>Once an employee clicks a link on Supplier.asp, this file will be requested. It shows the selected supplier detail and permits employees to add products.</td>
</tr>
<tr>
<td>See page 282</td>
<td></td>
</tr>
<tr>
<td>Template.asp</td>
<td>This file is used to create other pages during the development process. However, it will not be used during the runtime.</td>
</tr>
<tr>
<td>See page 289</td>
<td></td>
</tr>
</tbody>
</table>
After employees login, their mails will be shown on MainMenu.asp. In order to view the mails, they need to click a link, which is leaded them to this page.

This is a virtual directory connected to an image directory in the main website.

Table 5.21 Admin Website File Detail
CHAPTER SIX: IMPLEMENTATION

6.1 Overall Implementation Process

In this section, the implementation process problems encountered will be discussed. Although, a good project management plan was developed, many unexpected problems might happen. These problems will delay the project. Once a delay occurs in a previous phase, the following-phase starting date will be postponed.

In general, the project manager will monitor the whole process of the project. If a problem happens, he or she needs to respond quickly to solve this problem. In case of the delay, more resources, such as human resource, and hardware will be required to make the project finish on the schedule. However, the more resources it uses, the more expensive it costs to finish the project. In many cases, the cost to finish the project on time might exceed the final cost for the delay per se. Therefore, the project manager needs to weigh these costs carefully.
For this project, the author developed a project management plan (see table 5.1). However, there was too much information to gather in the study phase. This major problem caused the delay in the succeed phases (see table 6.1). After the study phase was finished, the author also found that there were more requirements than first estimated. This meant the e-commerce application would use more time for coding, debugging and testing in order to meet those requirements. In addition, the author had to take care of the entire project. Therefore, it was not possible to add more resources to finish the project within the planned schedule.

The author also went back to visit his home for three months. During this time the project was halted. This was the major cause of delay during documentation and presentation process. However, if this unusual delay was not count, the project will delay about two weeks.
6.2 Database Implementation

By changing only one file, DataConn.asp, in each website, the B2BECA can access at least three different database systems: Microsoft Access, Microsoft SQL Server and Oracle. However, the default database system used, and suggested by the author is Microsoft SQL Server. Because when we compare the price and performance of these three databases, the SQL Server is suitable for this project. The MS Access is not secure when putting on the Internet, and its performance is insufficient. Although the performance of MS SQL server is inferior to the Oracle 8, the total cost of ownership of Oracle is higher than that of MS SQL server. In addition, there is not much web hosting that supports Oracle database.

In order to connect to SQL Server database, a user account name IUSR with read and write permission must be added to the database. This user will be used as a login user for those who connect to the system as well. Figure 6.1 shows the real SQL server database diagram of the project.
All of the MS SQL Server codes used to generate the database, and necessary tables are included in Appendix B. The setup requirements are also mentioned in the following section. These requirements must be done before the application can be implemented, otherwise, the application will not operate, and unexpected errors may occur.

Figure 6.1 SQL Server Database Diagram.


6.3 Environment

In computer fields, the term environment and platform regularly refers to the aggregate of hardware and software used in a computer or a computer system. In this project, the application can be operated in two major different environments. The first one is the environment that is used for creating, testing, and debugging the application. Since only the developers use the system, security is not a concern for this environment. Nevertheless, the security issue becomes more important in the operating environment because the system is connected to the Internet, where information is transferred fast and unsecured.

In this part, the detail of hardware and software used during the development will be discussed, and then, the actual environment that should be used to operate the application will be suggested. Although users are free to implement the application in any other environment, this suggestion is a minimum requirement.
6.3.1 Development Environment

Figure 6.2 demonstrates the environment used during development process. All of the server applications, such as database server, and Internet server, are running on the server computer. In addition, all of the development tools are installed on the client computer. Although, the application is able to operate in this environment, it is not advisable to do that. The application should be implemented in the system recommended in the operating environment part.

Figure 6.2 Development Environment
The application is developed and tested using the following hardware and software configuration on a client and a server computer.

**Hardware:**

<table>
<thead>
<tr>
<th>Component</th>
<th>Server</th>
<th>Client</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CPU:</strong></td>
<td>AMD K2 450 With 3D Now</td>
<td>Pentium II 266</td>
</tr>
<tr>
<td><strong>Memory:</strong></td>
<td>256 MB.</td>
<td>96 MB.</td>
</tr>
<tr>
<td><strong>Hard Drive:</strong></td>
<td>Primary 8 GB</td>
<td>4 GB.</td>
</tr>
<tr>
<td><strong>LAN Card</strong></td>
<td>Ethernet Card 10/100 Mbps</td>
<td>PCMCIA LAN Card</td>
</tr>
</tbody>
</table>

**HUB**

| Component | Ethernet HUB | 10 Mbps |

**Software**

<table>
<thead>
<tr>
<th>Component</th>
<th>Server</th>
<th>Client</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Operating System:</strong></td>
<td>Windows 2000 Advance Server</td>
<td>Windows 2000 Professional</td>
</tr>
<tr>
<td><strong>Database Server:</strong></td>
<td>MS SQL Server 7.0</td>
<td>MS SQL Server 7.0:</td>
</tr>
<tr>
<td><strong>Web Server:</strong></td>
<td>Internet Information Server 5.0</td>
<td></td>
</tr>
<tr>
<td>Component</td>
<td>Software Information</td>
<td></td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>Application:</td>
<td>Desktop version</td>
<td></td>
</tr>
<tr>
<td>Web Server</td>
<td>Internet Information Server 5.0: Client version</td>
<td></td>
</tr>
<tr>
<td>Development Tools</td>
<td>- Microsoft Visual InterDev. Enterprise Edition</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Microsoft FrontPage 2000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Microsoft Access 2000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Notepad</td>
<td></td>
</tr>
<tr>
<td>Browser</td>
<td>Internet Explorer 5.0</td>
<td></td>
</tr>
</tbody>
</table>
6.3.2 Operating Environment

The B2BECA is recommended to implement in, at least, the three-tier network architecture as shown in figure 6.3 for the security purpose. In this architecture, the client hardware and software are not the major focus because users connect to the system through the Internet. For the server computers, however, a minimum specification will be suggested.

Figure 6.3 Three-tier Network Architecture
There are two different servers in this three-tier architecture. The first one is the Internet, or web, server. This server is used to login all users who connect via the Internet. It hosts all web page files, runs the ASP scripts, and sends the HTML files to client computer. The second one is the database server. This server provides all database service to the web server, which, then, passes this service to the end users, who will not be allowed to connect into this server directly. Each database connection must be done passing the web server. In the user points of view, they will not know about this server. They will think that all database services come from the Internet server. In practice, a firewall should be set up between this two servers to increase security of the system.

The following are the hardware and software configuration recommended for implementating applications on a client and a server computer.

Note: This is only a minimum requirement. A better hardware configuration is recommended for better performance.
**Hardware:**

**Web Server**

<table>
<thead>
<tr>
<th>Component</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>Pentium II class CPU</td>
</tr>
<tr>
<td></td>
<td>- Recommend speed 300 MHz</td>
</tr>
<tr>
<td>Memory</td>
<td>128 MB.</td>
</tr>
<tr>
<td>Hard Drive</td>
<td>4 GB</td>
</tr>
<tr>
<td>LAN Card</td>
<td>Ethernet Card 10/100 Mbps</td>
</tr>
</tbody>
</table>

**Database Server**

<table>
<thead>
<tr>
<th>Component</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>Pentium II class CPU</td>
</tr>
<tr>
<td></td>
<td>- Recommend speed 266 MHz</td>
</tr>
<tr>
<td>Memory</td>
<td>128 MB.</td>
</tr>
<tr>
<td>Hard Drive</td>
<td>8 GB.</td>
</tr>
<tr>
<td></td>
<td>- Hard disk array and type backup also recommended</td>
</tr>
<tr>
<td>LAN Card</td>
<td>Ethernet 10/100 Mbps</td>
</tr>
</tbody>
</table>

**Software**

**Web Server**

<table>
<thead>
<tr>
<th>Component</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System</td>
<td>Windows 2000 Server</td>
</tr>
<tr>
<td>Web Server</td>
<td>Internet Information Server 5.0</td>
</tr>
</tbody>
</table>
6.4 Requirements

In this section, the hardware and software setting up requirements, which are the Internet server, the database server, the mail service, and the client browser, are presented.

Internet Server:

- Windows 2000 Server (recommended)
  - Or Windows NT Server with ASP version 3.0
- A virtual directory for all ASP files
  - Allow only Read permit for users
  - Enable default document: Default.asp
  - Allow anonymous access: IUSR_servername
- A username login to the Server for an Administrator
  - The Administrator level permit
  - A username for the Internet user
  - Normally IUSR_servername
The typelib file for ASP MSADO15.Dll in the following directory
- C:\Program Files\Common Files\System\ado\msado15.dll

A virtual website for Macrotoys employees to admin the website
- Specific a special port number for security
- Virtual directory link to image in the main Macrotoys website

Database Server (SQL Server 7 on Windows 2000 Server):

- A database name MacrotoysSQL
  - The database and table structure is in the Data dictionary

- Username to login to SQL server for an Administrator
- Username IUSR and a blank password for Internet user
  - The administrator can create this account.
  - Allow read and write permit to MacrotoysSQL database

Note: the Macrotoys database can be implemented on MS Access 2000 or Oracle 8 also.
SMTP Service (Windows 2000 SMTP Server Service):

➢ SMTP host service for macrotoys.com
   o To handle any mail forward to Macrotoys.com

➢ A default user to login to SMTP server
   o IUSR user account. This account used for check email for Macrotoys' employees

Client Browser:

➢ Internet Explorer 4.0, Netscape Navigator 4.0 or better
   o Enable Cookies support
   o Enable Java Script support
6.5 The B2BECA Features

In this section the features and some print screen images of the application will be presented. This section will focus only on the main website because most customers and outsiders connect to the application through this website.

For the security purpose, the application is written in such a manner that every connection to the website must be forwarded to the Default page. Every customer must visit this page first in order to visite the other parts of the website.

This page, as shown in figure 6.4, will show information about the project itself, such as project name, university name, department name, degree, and subject. The contact information, such as advisor’s, project commeter’s, and the author’s name with the links to their email addresses, is also provided. Vistors can click on the links to send suggestions via email, as well.
The default page will be shown for five seconds. Then, the customer will be redirected to the Browser page. He or she might click on the link to enter to that page directly.

Figure 6.4 Default Page
After visiting the default page, visitors will be forwarded to the browser page, as demonstrated in figure 6.5. This page allows the user to browse all product categories, login to the system, or create a new user account.

Figure 6.5 Browser Page
When customers are interested in any product they can click on the product to see the product detail. Then, they can make an order by clicking at the buy button as shown in figure 6.6. Customers are also able to select the quantity of the product. However, since Macrotoys is a wholesaler, each item will be sold in a box. This means the quantity shown in the quantity list box will be set to fit in a box size.

Figure 6.6 Product Detail Page
After customers decide to purchase any product, they will be lead to the shopping cart page as seen in figure 6.7. In this page, customers have a chance to increase, decrease quantity of an item and remove any item from the shopping cart. Then, they can select to continue shopping or check out.

![Shopping Cart Page](image)

Figure 6.7 Shopping Cart Page

In case the customer decides to continue shopping, he or she will be forwarded back to the browser page again. This will give him or her an opportunity to add other items into his or her shopping cart.
If customers try to check out without logging in, they will be asked to login or register first. Figure 6.8 shows the registration page. Every new customer must create a new account at this page first, otherwise they cannot checkout.

Figure 6.8 Registration Page
For the return customers, they need to provide a correct user login name and a password in order to login to the system. Figure 6.9 shows user Ana during login process. In case the customer forgets his or her user name, and/or password, a link connected to forget password page is also provided.

![Figure 6.9 Member Login Page](image)

In this case, customers will be asked for a correct email address. Then, the application will send user login name and the password to those customers again.
After customers successfully login, their user login name will be shown here.

Customers are also allowed to select carriers they like.

Total price, tax, shipping cost and grand total are calculated.

Figure 6.10 Checkout Page

Customers are free to select the carriers they want. The prices will be recalculated after they click on a “Recalculate” button as shown in figure 6.10. Then, they may click at “Place Order” button to go to the next step.
Figure 6.11 demonstrates the next step of the check out process. At this point, customers will be able to select the payment they want. Once they provide the correct information, they can click on "checkout" button to finish the process. The order and its detail, then, will be recorded into the database and a confirmation page will be show. Customers, then, will be directed to print the page for confirmation purpose. An email, which confirms the order, will be sent to the customer too.
Once a customer logs in, he or she can view the process of his or her order by clicking on the check order link on the top of each page as seen in figure 6.12. The detail of each order, such as list of the products and their quantity, will be presented after the customer clicks on the order number.

![Order Status Page](image-url)

Figure 6.12 Order Status Page
6.6 How to Setup an E-commerce Site.

This section discusses the processes of setting up an e-commerce website: the domain name registration process and the website hosting process.

6.6.1 Domain Name Registration

When an organization decides to set up a website, a Webmaster needs to find a name for this website. He or she will check whether the name is registered by visiting domain name service provider's websites such as http://www.networksolutions.com, http://www.123domains.net, and http://www.registernames.com. The standard price of registering a domain name is $70 for a two-year period. However, this price varies, since there are many domain name service providers today.

Nevertheless, there are not many dot-com domain names left for registration today. If the business really needs a specific name, it is possible to buy a registered name. There are many dot-com domain names selling on the Internet. A Webmaster should visit at http://www.ebay.com, or http://auctions.yahoo.com. Unfortunately, the price for
purchasing a registered domain name is a lot higher than registering the new one. The other way is using dot-net or dot-org instead of a dot-com domain name.

6.6.2 Hosting the E-commerce Website

There are two methods for hosting a website. First, for a small to medium size business, outsourcing web hosting is a good answer. However, for a large organization, in-house hosting is a better idea for hosting its own e-commerce site.

6.6.2.1 Outsourcing Web Hosting: This is the easiest and cheapest way to host an e-commerce website. The web developer team only creates and uploads the website to a hosting service provider. The hosting service provider will take care of the rest. Using a hosting service lets many companies share the cost of a fast Internet connection for serving files. The cost for hosting a website varies depending on the features needed. For the e-business solution, a company needs some exclusive features such as a database system, as well.
Although outsourcing web hosting is easy and cheap, there are still many disadvantages. First, the business will rely on the hosting service provider system. If this system went down, the company must wait until the provider fixes the problems. Secondly, because all of the website files are kept on the provider system, they are unsecured. The service provider will be able to obtain the sensitive information, such as customers’ information, and sell this information to a competitor. Finally, since a server computer is shared, the performance of the e-commerce website might be reduced if there are too many sites shared on the same server.

6.6.2.2 In-house Hosting: For security and reliability purposes, larger companies usually consider in house hosting instead. This way, the website files will be kept on the company web server. Once customers are connecting to the company’s website, they will get only information the company wants them to see. The company also carries its own risk. Therefore, a good backup plan must be considered, such as a reliable backup system. Typically, an individual business hosting its own site would require a similar connection as offered by hosting service providers. The good thing is that no other companies share the connection.
Nevertheless, there are many drawbacks for this method. First, the company needs to hire more professionals (such as webmasters, web developers, database administrators, and network administrators) to operate the e-commerce website. In the large size company, however, the MIS department already takes care of these job functions. Secondly, the costs of setting up in house hosting are very expensive. These costs include the connection to the Internet, the special hardware, such as a router and a firewall, and the salary of IT professionals.

There are both advantages and disadvantages in these two methods. The IT manager must compare and make a good decision whether the company should host its own website or use a hosting service provider.
CHAPTER SEVEN: CONCLUSION

7.1 What is next?

In this part the author intent to provide some information for the students and/or developers who want to continue developing this project. Although, the B2BECAn has been tested and its functions are successfully implemented, there are still some functions that could be implemented and added into the application.

First, since the application is used for simulation purpose, the connections to outsiders' websites, such as UPS, a merchandise, bank or credit card company, are not really implemented. In order to implement this function, the knowledge of CGI and/or C programming, which is out of the scope of this project, is needed.

Second, in the real world, a development team, who creates an e-commerce website, is a combination of specialists with various knowledge and expertise. To increase the attractiveness of visitors, the user interface
Finally, since security becomes the major problem of Internet applications, the developers should secure any credit card transaction by using Secure Sockets Layer (SSL). SSL provides verification and secure communication between a web browser and server by using public key and secret key encryption. In order to do this, the developer needs to require a key encryption through third party Certificate Authorities (CAs) such as Verisign, for example.
7.2 Conclusion

In conclusion, the Business-to-Business Electronic Commerce Application (B2BECA) was developed for Macrotoys company. The ASP scripts were used for delivering fast, effective, and comprehensive business solutions. In this project, the toy wholesale business was analyzed. The B2BECA was designed based on this analysis. The application was developed, tested, and verified on a client/server network architecture, and then moved into the final physical implementation phase, which was implemented on the Windows NT Server, using the MS SQL Server as the back-end database support. The author also recommends some features that should be added to the application in the previous section.

Nevertheless, the technology in this field is advancing faster than anybody can expect. Although the concepts of business management applied in this project will not much change, the author believes that the appropriate lifecycle of the B2BECA is about two years. Then it needs to be redesigned again, otherwise it will be out-of-date.
Finally, the author has applied almost all of the knowledge in information management learned during the last two years. This knowledge includes the Information Management System, Information System Planning, strategy and Policy, Information Base System, Information Networking System, World Wide Web Strategy, and Electronic Commerce. Nevertheless, while cultivating this project, the author had an opportunity to study other advanced ASP methodologies, which can be used to implement e-commerce or other business websites, such as XML (Extensible Markup Language), COM+ (Component Object Model extension), MSMQ (Microsoft Message Queue), and ADSI (Active Directory Service Interfaces). Unfortunately, due to the restriction on the length of this project, these methodologies could not be covered. Interested readers may refer to entries [5], [6] and [7] of the Reference for more information on how to apply these advanced methodologies.
**Appendix A. ASP SOURCE CODE**

**Main Website Programming Code**

*About.asp*

```html
<!-- Language=VBScript -->
<!---------------------------------------------------------------------
* About.asp
* Program by Sumuscha Teesri
* About the Macrotoys company
* May 4, 2000
---------------------------------------------------------------------
---><!-- #INCLUDE FILE="Top.asp" -->
---><!-- #INCLUDE FILE="BrowseBar.asp" -->
---><!-- -->
---------------------------------------------------------------------
---><!-- -->
----------------------------------------------------------------------->

<META content="MSHTML 5.00.3103.1000" name=GENERATOR>
</HEAD>
</BODY>

<CENTER><FONT COLOR="#22ff22"><H1>About US</H1></FONT></CENTER>

About Macrotoys is a toy wholesaler company operating in Southern California by a group of MBA graduated managers. We sell variety of toy both license and non-license. The major market is in North America, United States and Canada. The company has the potential growth to other countries around the world.

<UL>

<LI><BIG><FONT COLOR="#22ff22">Vision:</BIG>
Leading in shaping collaborative, co-managed and cross-functional relationships with partners and customers by using information technology and moving this whole industry ahead quickly.

<LI><BIG><FONT COLOR="#22ff22">Mission:</BIG>
Leverage the Internet and electronic commercial technology to reach as many new potential customers as possible, and to be a national wholesaler of quality toys.

<LI><BIG><FONT COLOR="#22ff22">Value:</BIG>
Correlate businesses' partners, toy manufacture and toy retailer, together to respond the end-customer's obligation.

<LI><BIG><FONT COLOR="#22ff22">Customer and Markets:</BIG>
The major customers are toy retailers, and mini-distributors in The United States and Canada.

<LI><BIG><FONT COLOR="#22ff22">Product and Service:</BIG>
</UL>

154
Macrotoys sells the variety of toy products, both licensed and non-licensed, from average to high quantity.

**Driving Force:**

Exploit information technology including Internet and e-commerce expertise to achieve in high competitive market with satisfactory collaboration of the businesses' associates.

Feel free to browse our listings - you don't need to register to do this. If you find an item that you'd like to order, we will ask you to register with us.

---

**AddITEM.asp**

```vbscript
<%@ Language=VBScript %>
<%Response.Buffer = True%
'*****************************************************************************
' * AddItem.asp
' ** Program by Sumuscha Teesri
' ** May 7, 2000
'*****************************************************************************

' Declaration and set necessary variable
Dim intCartSize, intCartProd(), intCartQuan()
Dim blnAddQuan, intProd, intQuan
intProd = CInt(Request("ProdID"))
intQuan = CInt(Request("Quantity"))
blAddQuan = False

' If this is the first time user add a product
If Session("Shopping") = "True" Then

  ' Customer change shopping something to the cart
  CartProd = Session("CartProd")
  CartQuan = Session("CartQuan")
  ' Customer add more item
  ' Re-define the shopping cart size
  intCartSize = UBound(CartProd) + 1
  ReDim intCartProd(intCartSize), intCartQuan(intCartSize)
  ' keep old shopping information
  For i=0 To intCartSize-1
    intCartProd(i) = CartProd(i)
    intCartQuan(i) = CartQuan(i)
    If intProd = CartProd(i) Then
      intCartQuan(i) = intCartQuan(i) + intProd
      blnAddQuan = True
    End If
  Next
```
' Add new shopping information
If blnAddQuan Then
    ReDim Preserve intCartProd(intCartSize-1)
    ReDim Preserve intCartQuan(intCartSize-1)
Else
    intCartProd(intCartSize) = intProd
    intCartQuan(intCartSize) = intQuan
End If
Else
    ' This is the first time customer add the product
    ' Set shopping status
    Session("Shopping") = "True"

    ' Re-define the shopping cart array
    ReDim intCartProd(0), intCartQuan(0)

    ' Record product information to shopping cart
    intCartProd(0) = intProd
    intCartQuan(0) = intQuan
    ' Keep the array into session variable
End if

    ' Then update the shopping cart information into session variable
    Session("CartProd") = intCartProd
    Session("CartQuan") = intCartQuan

    ' Redirect to show the suppingcart status
    Response.Clear
    Response.Redirect "ShoppingCart.asp"
%>

AddUser.txt

<%@ Language=VBScript %>
<%Response.Buffer = True%>
<!-- #INCLUDE FILE="DataConn.asp" -->
<%*
****************************************************
* AddUser.asp
* Program by Sumuscha Teesri
* Add Customer user to Macrotoys database
* May 4, 2000
****************************************************

Dim objRSCUS
Set objRSCUS = Server.CreateObject("ADODB.Recordset")
objRSCUS.Open "Customers", objConn, adOpenForwardOnly, adLockOptimistic, adCmdTable

' Currently logged-on user
If Session("UPDATE") Then
    objRSCUS.Filter = "CusID = " & Session("CusID")
Else
    ' New session

156
'Make sure that unique fields are checked
objRSCUS.Filter = "CompanyName = '' & Session("CompanyName") & '''
& _
  "AND LogInName = '' & Session("LogInName") & ''' & _
  "AND Email = '' & Session("Email") & '''

' User not found ...so add a new record
If objRSCUS.EOF Then
  objRSCUS.AddNew
Else
  ' Request fields are not unique register again
  Response.Clear
  Response.Redirect "Register.asp"
End IF
End IF

' Write customer details to record
objRSCUS("CompanyName") = Session("CompanyName")
objRSCUS("Address1") = Session("Address1")
objRSCUS("Address2") = Session("Address2")
objRSCUS("City") = Session("City")
objRSCUS("Region") = Session("Region")
objRSCUS("PostalCode") = Session("PostalCode")
objRSCUS("Country") = Session("Country")
objRSCUS("Phone") = Session("Phone")
objRSCUS("Fax") = Session("Fax")
objRSCUS("LogInName") = Session("LogInName")
objRSCUS("Password") = Session("Password")
objRSCUS("WebSite") = Session("WebSite")
objRSCUS("Title") = Session("Title")
objRSCUS("FirstName") = Session("FirstName")
objRSCUS("LastName") = Session("LastName")
objRSCUS("Email") = Session("Email")
objRSCUS("Direct") = Session("Direct")

' The following fields are not allow to be updated
If Not(Session("Update")) Then
  ' New customer always provide EmpID 1
  objRSCUS("EmpID") = 1
  objRSCUS("Since") = Now
  objRSCUS("Type") = "GC"
End If

' Update the database
objRSCUS.Update

' Create session variables
Dim strName, strValue
For Each strField in objRSCUS.Fields
  strName = strField.Name
  strValue = strField.Value
  Session(strName) = strValue
Next
Session.Contents.Remove("Password")
Session.Contents.Remove("VerifyPassword")
Session.Contents.Remove("WebSite")
Session.Contents.Remove("Since")
Session.Contents.Remove("Phone")
Session.Contents.Remove("FAX")

' Declare that current user is validated
'Session("blnValidUser") = True

' Close database
objRSCUS.Close
Set objRSCUS = Nothing

Dim strMsg
If Session("Update") Then
    strMsg = "Your account was updated in the database."
Else
    strMsg = "Your account was created. You can enjoy shopping now."
End If
If Session("CheckOut") Then
    strMsg = strMsg & "&redirect=shipping.asp"
Else
    strMsg = strMsg & "&redirect=browser.asp"
End If

' Redirection to browser page
Response.Clear

%>

Adrot.txt

Redirect AdRotatorRedirector.asp
width 88
height 31
border 0
*
image\nts_iis.gif
Microsoft Internet Information Services
33
image\iisside.gif
http://www.microsoft.com/backoffice
Microsoft Backoffice
33
image\webandhost.gif
http://www.webandhost.com
Web hosting with Webandhost.com
33
AdRotatorRedirector.asp

<%@ Language=VBScript %>
<%
'***************************************************************
'* AdRotatorRedirector.asp
'* Program by Sumuscha Teesri
'* Record Ad click statistics
'* before redirection
'* May 5, 2000
'***************************************************************
Response.Buffer = True
strURL = Request.QueryString("url")

Select Case lcase(strURL)
        Application.Lock
        Application("IISAd") = Application("IISAd") + 1
        Application.Unlock
    Case "http://www.microsoft.com/backoffice"
        Application.Lock
        Application("BackOfficeAd") = Application("BackOfficeAd") + 1
        Application.Unlock
    Case "http://www.webandhost.com"
        Application.Lock
        Application("WebAndHostAd") = Application("WebAndHostAd") + 1
        Application.Unlock
End Select
Response.Clear
Response.Redirect strURL
%

Bottom.asp

<!********************************************************************************
* Bottom.asp.
* Program by Sumuscha Teesri
* Bottom Template
* May 4, 2000
********************************************************************************>
<!-*------------------------------>
<!-* Begin Close                *-->
<!-*------------------------------>
<** Link bar **)>
<HTML><HEAD>
<META content="text/html; charset=unicode" http-equiv=Content-Type>
<META content="MSHTML 5.00.3103.1000" name=GENERATOR></HEAD>
<BODY>
<HR>
<TABLE BORDER=0 WIDTH=600>
    <TR ALIGN=middle>**** Link To Browser page **>
        <TD WIDTH=200><%If blnBrowserPage = True Then

159
Response.Write "<B>[Browse the Products]<B>"</TD>"

Else
  Response.Write "<A HREF="Browser.asp"" & _
  "onMouseOver=""window.status='Browse Product';return true'"" & _
  "onMouseOut=""window.status='';return true;'"">" & _
  "<B>[Browse the Products]<B>"</A></TD>"
End If

%>

<% Dim objAd
  Set objAd = Server.CreateObject("MSWC.AdRotator")
  Response.Write objAd.GetAdvertisement("adrot.txt")
%>

General Information**>
</HR>
</TABLE></TABLE></TABLE>

<TR>
<table>
<thead>
<tr>
<th>Width</th>
<th>Contact Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>300</td>
<td>Sumuscha Teesri</td>
</tr>
<tr>
<td>300</td>
<td>3512 N. Bond St.; San Bernardino, CA 92405</td>
</tr>
<tr>
<td>300</td>
<td>Phone: 1-530-685-5016 FAX: 1-530-235-7314</td>
</tr>
<tr>
<td>300</td>
<td>XOOM.COM &lt;LI&gt; GEOCITY.COM &lt;LI&gt; WEBANDHOST.COM</td>
</tr>
</tbody>
</table>

Questions or comments about this web site: 

Copyright © 1999-2000 Sumuscha Teesri Last modified: October 16, 2000 Best view on 800x600 Pixels
**Browse.asp**

```vbscript
<%@ Language=VBScript %>
**********************************************************************
* Browser.asp
* Program by Sumuscha Teesri
* User for browse all product(function as main Page)
* May 4, 2000
**********************************************************************
<!— #INCLUDE FILE="Top.asp" -->
<!— #INCLUDE FILE="BrowseBar.asp" -->
</i_**************************>
<!--* Begin main *--><br/>
<font color="#22FF22"><b>Macrotoys Product Categories:</b><br/></font>
binBrowserPage = True

'Connect to the database using SQL statement
**********************************************************************
'Dim objRS : already declare in browserbar
Set objRS = Server.CreateObject("ADODB.Recordset")
strSQL = "SELECT p.ProdID, ProdName, Picture, SalePrice, " & _
"s.SubCatID, SubCatName, c.CatId, CatName " & _
"FROM products p, subcategories s, categories c,
productcats pdc " & _
"WHERE p.prodid = pdc.prodid and pdc.subcatid = s.subcatid
and " & _
"s.catId = c.catid

' Filter only category needed
If Request.QueryString("Cat")<"" Then
strSQL = strSQL & " and c.CatID = " & Request.QueryString("Cat")

' Filter only subcategory needed
If Request.QueryString("SubCat") <> "" Then
strSQL = strSQL + " and s.SubCatID = " & Request.QueryString("SubCat")
End If
End If

'Order by CatID
strSQL = strSQL & " ORDER BY c.CatId; "

' Open Database with query
objRS.Open strSQL, objConn, adOpenStatic , adLockReadOnly, adCmdText
objRS.PageSize = 3

' Write information
If Not objRS.EOF Then
    ' Set Page Number
    Dim PageMove
    PageMove = Request("PageMove")
```

162
If Request("PageNo") = "" Then
    PageNo=1
Else
    PageNo = Request("PageNo")
End If

' Set Navigation button
If PageMove = "Previous" or PageMove = "Next" Then
    Select Case PageMove
    Case "Previous"
        If PageNo > 1 Then
            PageNo = PageNo -1
        Else
            PageNo = 1
        End If
    Case "Next"
        If objRS.AbsolutePage < objRS.PageCount Then
            PageNo = PageNo + 1
        Else
            PageNo = objRS.PageCount
        End If
    Case Else
        PageNo = 1
    End Select
End If

objRS.AbsolutePage = PageNo

Dim strMessage, intItemCount
strMessage = "<TABLE BORDER=0 WIDTH=550><TR><TD WIDTH=450>" & _
    "<SMALL>"
' Write category name
If Request.QueryString("Cat") <> "" Then
    strMessage = strMessage & objRS("CatName")
' Write subcategory name
If Request.QueryString("SubCat") <> "" Then
    strMessage = strMessage & ": " & objRS("SubCatName")
Else
    strMessage = strMessage & ": All subcategories"
End If
Else
    strMessage = strMessage & "Show all product in all categories"
End If

' Write page number
strMessage = strMessage & "</TD></SMALL><TD ALIGN=RIGHT WIDTH=100>" & _
    "<SMALL>" & PageNo & ": " & objRS.PageCount & _
    "</SMALL></TD></TR></TABLE>"
Response.Write strMessage

' Table head
Response.Write "<TABLE BORDER=1 WIDTH=600><TR>"
Loop 3 record per page
For intItemCount = 1 to objRS.PageSize
  ' Write table body
  Response.Write "<TR><TD WIDTH=30 VALIGN=TOP ALIGN= CENTER>" &
  HREF=ProductDetail.asp?ProdID=" & _
  objRS("ProdID") & ">
  objRS("ProdID") & ">
  "<TD VALIGN=TOP WIDTH="452">" & _
  "<A HREF=ProductDetail.asp?ProdID=" &
  objRS("ProdID") & _
  ">" & objRS("ProdName") & "</TD>" & _
  "<TD VALIGN=TOP ALIGN=RIGHT
  HREF=ProductDetail.asp?ProdID=" & _
  objRS("SalePrice") & "</TD>" & _
  "<TD WIDTH="58">"<A
  HREF=ProductDetail.asp?ProdID=" & _
  objRS("ProdID") & ">
  BORDER=0 SRC=image\product\" & _
  objRS("Picture") & " WIDTH=58
  HEIGHT=72</A></TD></TR>"
  objRS.MoveNext
  ' No more record
  If objRS.BOF Then
    Exit for
  End If
Next
Response.Write "</TABLE>"
strMessage = "PageNo=" & PageNo & _
"&Cat=" & Request.QueryString("Cat") & _
"&SubCat=" & Request.QueryString("SubCat")
If PageNo > 1 or PageNo < objRS.PageCount Then
  Response.Write "<TABLE BORDER=0><TR><TD WIDTH=100>&nbsp;</TD><TD>" & _
  "<TABLE BORDER=2 WIDTH=400><TR><TD WIDTH=100 HEIGHT=30>"
  End If
If PageNo > 1 Then
  ' Write Animation button for the Previour record navigator
  <A HREF="browser.asp?%=" strMessage %>&PageMove=Previous"
onmouseover="document["fpAnimswapImgPre"].imgRolln=document["fpAnimswapImgPre"].src;
document["fpAnimswapImgPre"].src=document["fpAnimswapImgPre"].imgRolln;
window.status='Previous Record';return true;"
onmouseout="document["fpAnimswapImgPre"].src=document["fpAnimswapImgPre"].lowsrc;
window.status='';return true";
<img border="0" src="image/Back.gif" width="100" height="30"
id="fpAnimswapImgPre" name="fpAnimswapImgPre" dynamicanimation="fpAnimswapImgPre"
lowsrc="image/BackOV.gif">
</A><
%>
End IF
Response.Write "</TD><TD WIDTH=100 HEIGHT=30>"
If PageNo < objRS.PageCount Then
' Write Animation button of Next record navigator
%>
<A HREF="browser.asp?f= strMessage & PageMove=Next"
onmouseover="document["fpAnimswapImgNext"].imgRolln=document["fpAnimswapImgNext"].src;
document["fpAnimswapImgNext"].src=document["fpAnimswapImgNext"].lowsrc;
window.status='Next Record';return true;"
onmouseout="document["fpAnimswapImgNext"].src=document["fpAnimswapImgNext"].lowsrc;
window.status='';return true">
<img border="0" src="image/Next.gif" width="100" height="30"
id="fpAnimswapImgNext" name="fpAnimswapImgNext" dynamicanimation="fpAnimswapImgNext"
lowsrc="image/NextOV.gif">
</A>

<% End If
Response.Write "</TD><TD WIDTH=100 HEIGHT=30></TD></TR></TABLE></TD></TR></TABLE>"
'Close record set
objRS.Close
Set objRS = Nothing
Else
' This is no information in subcategory
Response.Write "<BR><BR><CENTER>" & _
"<P>Sorry We are currently do not have any
product in this category." & _
"<BR>Please try visit other category.<BR>" & _
"You may find the product you want</CENTER>"&<BR>"
End If

*******************************************************************
* The following code used for debug and testing only.
*
*******************************************************************
'For Each objltem In Session.Contents

165
If Not IsArray(Session(objItem)) Then
  Response.Write objItem & " = " & Session(objItem) & "<BR>"
Else
  Response.Write objItem & " Is an Array<BR>"
End If
'Next

'If Session("Shopping") <> "" Then
  Dim intCartSize, i
  CartProd = Session("CartProd")
  CartQuan = Session("CartQuan")
  intCartSize = UBound(CartProd)
  For i = 0 To intCartSize
    Response.Write "ProdID : " & CartProd(i) & _
      " Quantity : " & CartQuan(i) & "<BR>"
    Next
'End If
%>

</TABLE BORDER="0" WIDTH=150 CELLSspacing="0" CELLPadding="0" ALIGN="LEFT">
  <TR>
    <TD ALIGN="CENTER" WIDTH="150" HEIGHT="30"><IMG SRC="image\welcom.gif"></TD>
  </TR>
  <TR>
    <TR>
      <!--* Begin Navigation *-->
      <!---************************************************************************-->
      <TABLE BORDER="0" WIDTH=150 CELLSspacing="0" CELLPadding="0" ALIGN="LEFT">
        <TR>
          <TD ALIGN="CENTER" WIDTH="150" HEIGHT="30"><IMG SRC="image\welcom.gif"></TD>
        </TR>
        <TR>
          <TR>
            <!--* If login user show user name or show Guess***>
            <TD ALIGN="CENTER" WIDTH="150"><FONT COLOR="#22FF22"><STRONG><If Session("CusID")<>"" Then
              Response.Write Session("LogInName")
            Else
              Response.Write "Guess"
            End If
            </STRONG></FONT></TD>
          </TR>
          <TR>
            <TD ALIGN="CENTER" WIDTH="150" HEIGHT="30"><IMG SRC="image\product.gif"></TD>
          </TR>
          <TR>
            <!--* Begin Navigation *-->
            <!---************************************************************************-->
          </TR>
        </TR>
      </TABLE>
    </TR>
  </TR>
' Create Categories and subcategories active bar
'********************************************************************

' Connect to Categories table
Dim objRS
Set objRS = Server.CreateObject("ADODB.Recordset")
objRS.Open "Categories", objConn, adOpenForwardOnly, adLockReadOnly, adCmdTable

' Request CatID from Previous page
Dim strCat, strCatID
strCatID = Request.QueryString("Cat")
strCat = "<B>

' Loop until end categories table
While Not objRS.EOF
    ' Show Categories active bar
    ' Hide Status bar
    strCat = strCat & "<TR><TD><A HREF="browser.asp?Cat=" & objRS("CatID") & 

        " onMouseOver="window.status=' & objRS("CatName") & "; return true" & _
        objRS("CatName") & ">
    objRS("CatName") & ">

        " window.status='; return true"">" & _
    objRS("CatName") & "</A></TD></TR>

' When customer select categories, show subcategories
If cstr(objRS("CatID")) = strCatID Then
    ' Delete HTML end table tag and reduce subcategories font size
    strCat = Left(strCat, (Len(strCat) - 10)) & "</B></SMALL>"

    ' Connect to subcategories table
Dim objRS2
Set objRS2 = Server.CreateObject("ADODB.Recordset")
objRS2.Open "SubCategories", objConn, _
    adOpenForwardOnly, adLockReadOnly, adCmdTable
objRS2.Filter = "CatID = " & strCatID

    ' Loop until end subcategories table
    While Not objRS2.EOF
        strCat = strCat & "<BR>&nbsp;&nbsp;&nbsp;<A HREF="browser.asp?Cat=" & _

            objRS("CatID") & 

            objRS("SubCatName") & 

            objRS2("SubCatName") & ">
        objRS2("SubCatName") & ">

            " onMouseOver="window.status='; return true" & _

            " window.status='; return true"">" & _
        objRS2("SubCatName") & ">

    objRS2.MoveNext
    Wend

    ' End subcategories table create new row
    strCat = strCat & "<BR></TD></TR></FONT></SMALL><B>"

    ' Close subcategories table
objRS2.Close
Set objRS2 = Nothing

End if
objRS.MoveNext

Wend

' End categories table add end tag
strCat = strCat & "</BxTRxTD HEIGHT=250>&nbsp;</TD</TR>"

' Write the string tag to client
Response.Write strCat

' Close subcategories table
objRS.Close
Set objRS = Nothing

</TABLE>

<!-- Create fream table for body-->
<TABLE BORDER="0" WIDTH=80% CELLPADDING="0" CELLSpACING="0">
<TD ALIGN="LEFT" VALIGN="TOP" WIDTH=85%>

CheckLogIn.asp

<!-- Language=VBScript -->
<![CDATA[
<!--#INCLUDE FILE="DataConn.asp" -->

<!--#-->

Dim strLogInName, strPassword
Dim strRedirect
strRedirect = ""
strLogInName = Request("LogInName")
strPassword = Request("Password")
'Turn the buffer on before redirect
Response.Buffer = True

' Connect to the database
Dim rsUsers
Set rsUsers = Server.CreateObject("ADODB.Recordset")
strSQL = "SELECT * FROM Customers WHERE LogInName = "" & strLogInName & " " &
rsUsers.Open strSQL, objConn

' If LogIn Name was not found
If rsUsers.EOF Then

168
' Allow only three times
Select Case Request("LogFail")
  Case "Second"
    ' Second Time Try again
    strRedirect = "login.asp?LogFail=Second"
  Case "Last"
    ' Last time forward to forget password
    strRedirect = "ForgetPassword.asp"
  Case Else
    ' Just the first try
    strRedirect = "login.asp?LogFail=First"
End select
' Redirect to other page
Response.Clear
Response.Redirect strRedirect
Else
  ' Found User Login Name
  Session("LogInName") = Request("LogInName")
  ' Check Password
  If UCase(rsUsers("Password"))= UCase(strPassword) Then
    ' Login Name and Password OK.
    Dim strName, strValue
    For Each strField in rsUsers.Fields
      ' Populate session variable
      strName = strField.Name
      strValue = strField.Value
      Session(strName) = strValue
    Next
    Session.Contents.Remove("Password")
    Session.Contents.Remove("VerifyPassword")
    Session.Contents.Remove("WebSite")
    Session.Contents.Remove("Since")
    Session.Contents.Remove("Phone")
    Session.Contents.Remove("FAX")
    ' Successful Login
    Session("VerifyPassword") = rsUsers("Password")
    Session("blnValidUser") = True
    Response.Cookies("SaveLogIn")("CusID") = Session("CusID")
  End If
  Else
    ' Go to browser page
    Response.Clear
    Response.Redirect "Message.asp?Msg=You are now loging" & _
    " in to our system.<BR>Thank You." & _
    "&redirect= browser.asp"
  End If
End If
  ' Go this far mean Password not correct
Select Case Request("LogFail")
  ' Allow only three time
Case "Second"
    ' Second time try again
    strRedirect = 
    "login.asp?LogFail=Second&WrongPW=True"
Case "Last"
    ' The last time go to forgetpassword
    strRedirect = "ForgetPassword.asp"
Case Else
    ' Just the first try
    strRedirect = 
    "login.asp?LogFail=First&WrongPW=True"
End select
    'Go to browser page
    Response.Clear
    Response.Redirect strRedirect
End IF
rsUsers.Close
%

CheckOrder.asp

<%@ Language=VBScript %>
<!*****************************************************************************
   * CheckOrder.asp
   * Program by Sumuscha Teesri
   * May 10, 2000
   *****************************************************************************>
<!include file needed >
<!--- #include file="Top.asp" -->
<!--- #include file="BrowseBar.asp" -->

<!-*****************************************************************************-
<!- Begin main -->
<!-*****************************************************************************-
<%Response.Buffer = True

    ' If customer does not log in redirect
    If Session("CusID") ="" Then
        Response.Clear
        Response.Redirect "Message.asp?Msg=Please log in or register with our system" & _
        "<BR>in order to buy our products." & _
        "&redirect=login.asp"
    End If

    Dim strMessage
    strMessage = "<CENTER><H1><FONT COLOR=#22FF22>Order Status</FONT></H1></CENTER>"
Set objRS=Server.CreateObject("ADODB.Recordset")
strSQL = "SELECT OrderID, OrderDate, s.Description Status, " & _
    "c.Description Carrier, Tracking " & _
    "FROM Orders o, OrderStatus s, Carriers c " & _
"WHERE Status=s.ID and o.CarrierID=c.CarrierId and CusId=" & _
Session("CusID") & ";"
objRS.Open strSQL, objConn, adOpenForwardOnly, adLockReadOnly, adCmdText

If Not objRS.EOF Then
  strMessage = strMessage & "<TABLE BORDER=1 WIDTH=600><TR>" & _
    "<TD WIDTH=60 ALIGN=CENTER><B>OrderID</B></TD>" & _
    "<TD WIDTH=90 ALIGN=CENTER><B>Date</B></TD>" & _
    "<TD WIDTH=90 ALIGN=CENTER><B>Status</B></TD>"& _
    "<TD WIDTH=200 ALIGN=CENTER><B>Carrier</B></TD>" & _
    "<TD WIDTH=160 ALIGN=CENTER><B>Tracking#</B></TD></TR>"
  While Not objRS.EOF
    strMessage = strMessage & "<TD WIDTH=60 ALIGN=CENTEx" & _
      "<A HREF=OrderDetail.asp?OrderID=" & objRS("OrderID") & _
      ">" & objRS("OrderID") & "</TD>" & _
      "<TD WIDTH=90 ALIGN=CENTER>" & _
      FormatDateTime(objRS("OrderDate"),2) & _
      "</TD>" & _
      "<TD WIDTH=90>"& _
      objRS("Status") & "</TD>"& _
      "<TD WIDTH=200>"& _
      objRS("Carrier") & "</TD>" & _
      "<TD WIDTH=160>"& objRS("Tracking") & _
      "</TD></TR>"
    objRS.MoveNext
  Wend
strMessage = strMessage & "</TABLE>"
Else
  strMessage = "<BR><BR><CENTER><H2><FONT COLOR=#FF0000>" & _
"Sorry you don't have any order now</CENTER></H2></FONT>"
End If
objRS.Close
Set objRS = Nothing
Response.Write strMessage
%

<!-- #INCLUDE FILE="Bottom.asp" -->
<!***************************************************************************>

171
Checkout.asp

<%@ Language=VBScript %>
<-- #INCLUDE FILE="DataConn.asp" -->
<% Response.Buffer = True

' If customer does not log in redirect
If Session("CusID") ="" Then
    Response.Clear
    Response.Redirect "Message.asp?Msg=Please log in or register with our system" & _
    "<BR>in order to buy our products." & _
    "&redirect=login.asp"
End If

' Declaration statement
Dim intCarrierID, intAccNum, intOrderID
Dim strTracking, intCartSize

' Set account number
intAccNum = Request("AccNum1") & Request("AccNum2") & _
            Request("AccNum3") & Request("AccNum4")

' Simulation shipping by Generate Tracking number randomize
strTracking = "1Z7X291303" & Int((99999999 - 10000000 + 1) * Rnd + 10000000)

' Find Carrier ID
Set objRS = Server.CreateObject("ADODB.Recordset")
strSQL = "SELECT CarrierID FROM Carriers WHERE " & _
            "Description =" & Session("Carrier") & ";";
objRS.Open strSQL, objConn, adOpenStatic, adLockReadOnly, adCmdText
intCarrierID = objRS("CarrierID")
objRS.Close

' ******************************
'* For debug Only
' ******************************
' Response.Write "Tracking Number : " & strTracking & "<BR>"
' Response.Write "CarrierID : " & intCarrierID & "<BR>"
' Response.Write "Payment Type : " & Request("PaymentType") & "<BR>"
'****************************

' Add orders table
Set objRS = Server.CreateObject("ADODB.Recordset")
strSQL = "SELECT CarrierID FROM Carriers WHERE " & _
            "Description =" & Session("Carrier") & ";"
objRS.Open strSQL, objConn, adOpenStatic, adLockReadOnly, adCmdText
intCarrierID = objRS("CarrierID")
objRS.Close

' Add new record
objRS.AddNew
objRS("CusID") = Session("CusID")
objRS("OrderDate") = Now()
objRS("Status") = "IP"
objRS("Payment") = Request("PaymentType")
objRS("AccNum") = intAccNum
objRS("CarrierID") = intCarrierID
objRS("Tracking") = strTracking
objRS.Update
objRS.Close

strSQL = "SELECT OrderID FROM Orders Where Tracking='" & strTracking & ";"
objRS.Open strSQL, objConn, adOpenStatic, adLockReadOnly, adCmdText
' Keep OrderID for create Transacts record
intOrderID = objRS("OrderID")
objRS.Close

' For debug only
'Response.Write "OrderID : " & intOrderID & "<BR>"

' Add Transacts table
objRS.Open "TRANSACTS", objConn, adOpenForwardOnly, adLockOptimistic, adCmdTable
' Put everything in shopping cart to Transaction
CartProd = Session("CartProd")
CartQuan = Session("CartQuan")
intCartSize = UBound(CartProd)
For i=0 to intCartSize
' Add Transacts records
objRS.AddNew
objRS("OrderID") = intOrderID
objRS("ProdID") = CartProd(i)
objRS("Quantity") = CartQuan(i)
objRS.Update

NEXT
objRS.Close
Set objRS = Nothing

' Clear shopping cart informaiton
Session.Contents.Remove("Carrier")
Session.Contents.Remove("TotalPrice")
Session.Contents.Remove("ShippingCost")
Session.Contents.Remove("Tax")
Session.Contents.Remove("GrandTotal")
Session.Contents.Remove("Shopping")
Session.Contents.Remove("CartProd")
Session.Contents.Remove("CartQuan")
Session.Contents.Remove("Checkout")

' Redirection back to browser page
Response.Clear
Response.Redirect "Message.asp?Msg=Your order was placed<BR>" & _
"<BR>Thank You." & _
"&redirect=browser.asp"
CheckRegisterInforInfo.asp

<%@ Language=VBScript %>  
***** CheckRegisterInforInfo.asp  
* Program by Sumuscha Teesri  
* May 4, 2000  
******************

<!** Include database connection file**>
<!---#INCLUDE FILE=DataConn.asp -->

<%  
Open buffer for compactible with ASP2.0  
Response.Buffer = True  

Dim strError, strValue  
    strError = ""  
' Populate session variable  
For Each Item In Request.Form  
    strValue = Request.Form(Item)  
    Session(Item) = strValue  
Next  
' Clear unimportant information from session variable  
    Session.Contents.Remove("Submit")  
    Session.Contents.Remove("VerifyPassword")  

'**********************************************************************  
'* Used for debug and testing only  
'**********************************************************************  
'For Each Item In Session.Contents  
'    Response.Write Item & ": " & Session(Item) & 
'Next  
'**********************************************************************

' Old user update information or  
' New user register  
If Not(Session("Update")) Then  
    Dim objRS  
    Set objRS = Server.CreateObject("ADODB.Recordset")  
    strSQL = "SELECT CompanyName FROM Customers WHERE CompanyName=" & Session("CompanyName") & ";"  
    objRS.Open strSQL, objConn  
    If Not objRS.EOF Then  
        strError = ":ComUnique=False"  
    End If  
    objRS.Close  

    strSQL = "SELECT LogInName FROM Customers WHERE LogInName=" & Session("LogInName") & ";"  
    objRS.Open strSQL, objConn  
    If Not objRS.EOF Then  
        If strError = "" Then  
            strError = ":LogInUnique=False"  
        End If  
End If

174
Else
    strError = strError & "&LogInUnique=False"
End If
End If
objRS.Close
strSQL = "SELECT Email FROM Customers WHERE Email='" & _
    Session("Email") & ";"
objRS.Open strSQL, objConn
If Not objRS.EOF Then
    If strError = "" Then
        strError = "?EmailUnique=False"
    Else
        strError = strError & "&EmailUnique=False"
    End If
End If
End If
Set objRS = Nothing
End If

' Check Not Null Fields
******************************************************************************
If Request("CompanyName") = "" Then
    If strError = "" Then
        strError = "?ComNameFail=True"
    Else
        strError = strError & "&ComNameFail=True"
    End If
End If
If Request("Address1") = "" Then
    If strError = "" Then
        strError = "?AddFail=True"
    Else
        strError = strError & "&AddFail=True"
    End If
End If
If Request("City") = "" Then
    If strError = "" Then
        strError = "?CityFail=True"
    Else
        strError = strError & "&CityFail=True"
    End If
End If
If Request("Country") = "" Then
    If strError = "" Then
        strError = "?CountryFail=True"
    Else
        strError = strError & "&CountryFail=True"
    End If
End If
If Request("Phone") = "" Then
    If strError = "" Then
        strError = "?PhoneFail=True"
    Else
        strError = strError & "&PhoneFail=True"
    End If
If Request("FirstName") = "" Then
    If strError = "" Then
        strError="?FirstFail=True"
    Else
        strError=strError & "&FirstFail=True"
    End If
End If
If Request("LastName") = "" Then
    If strError = "" Then
        strError="?LastFail=True"
    Else
        strError=strError & "&LastFail=True"
    End If
End If
If Request("Email") = "" Then
    If strError = "" Then
        strError="?EmailFail=True"
    Else
        strError=strError & "&EmailFail=True"
    End If
End If
If Request("LogInName") = "" Then
    If strError = "" Then
        strError="?LogInFail=True"
    Else
        strError=strError & "&LogInFail=True"
    End If
End If

'*******************************************************************
' End Check Not Null Fields
'
' Clear Buffer before redirect
'Response.Clear
If strError <> "" Then
    'There are some errors.
    Response.Redirect "Register.asp" & strError
    'Response.Write strError
Else
    'No error Adduser.
    Response.Redirect "AddUser.asp"
End If
%>
Comment.asp

<%@ Language=VBScript %>

Comment.asp
* Program by Sumuscha Teesri
* Send comment or question to Macrotoys
* May 4, 2000

<!**Make sure that user provide information request**>
<SCRIPT Language="JavaScript">

function VerifyData()
{
  var bSend;
  if (document.frmComment.SenderName.value == "")
  {
    alert("Please enter your name");
    return false;
  }
  else
  {
    if (document.frmComment.SenderEmail.value == "")
    {
      alert("Please enter your Email address");
      return false;
    }
    else
    {
      //bSend = confirm("Please click OK to send your mail")
      //if (bSend)
      //  return true;
      //else
      //  return false;
    }
  }

  return false;
}

</SCRIPT>

<!-- BEGIN main -->

<FORM ACTION="Sendmail.asp" NAME="frmComment" METHOD="POST"
onSubmit="return VerifyData()"

</FORM></CENTER><P>

<!-- Create input form-->

</CENTER>

<!-- Since this file normally redirected for other pages put the mail subject, and importance here-->

</BODY>
</HTML>
<INPUT TYPE=HIDDEN NAME="Importance" VALUE=<%= Select Case Request("Importance") Case "High" Response.Write CdoHigh Case "Low" Response.Write CdoLow Case Else Response.Write CdoNormal End Select %>>

<%** Populate data if user already login**>
Please enter your name:<BR>
<INPUT NAME=SenderName <%If Session("CusID") <> "" Then%>
  VALUE="<%= Session("LastName") & ", " & Session("FirstName")%>"
<%End If%>><BR>
Please enter your Email address:<BR>
<INPUT NAME=SenderEmail <%If Session("CusID") <> "" Then%>
  VALUE="<%= Session("Email")%>"
<%End If%>><BR>
Enter your comment or question here:<BR>
<TEXTAREA NAME="Comment" COLS=70 ROWS=5></TEXTAREA><P>
<INPUT TYPE="SUBMIT" VALUE="Submit: <%= Request("Subject")%>" >
</FORM>

<!-- #INCLUDE FILE="Bottom.asp" -->

DataConn.asp

<%** Include necessary libary**>
<%-- METADATA TYPE="typelib" FILE="C:\Program Files\Common Files\System\ado\msado15.dll" -->

Dim objConn
Set objConn = Server.CreateObject("ADODB.Connection")
Dim strSQL
' Define database connection modify this line as needed
objConn.Open "Provider=SQLOLEDB.1;Persist Security Info=False;User ID=IUSR;" & _
  "Initial Catalog=MacrotoysSQL;Data Source=ZEBRASERVER"
' If Session("blnValidUser") = True and Session("CusID") = "" Then
' Dim rsUsersCheck, strSQL
Set rsUsersCheck = Server.CreateObject("ADODB.Recordset")
strSQL = "SELECT * FROM Customers " & _
"WHERE CusID = " & Request.Cookies("SaveLogin")("CusID") & ";"
rsUsersCheck.Open strSQL, objConn
Dim strName, strValue
For Each strField in rsUsersCheck.Fields
' populate session variable
strName = strField.Name
strValue = strField.Value
Session(strName) = strValue
Next
rsUsersCheck.Close
Set rsUsersCheck = Nothing
End If

Default.asp

<%@ Language=VBScript %>
<%!******************************************************************************************
* Default.asp
* Program by Sumuscha Teesri
* Default file before forward to other page
* May 4, 2000
******************************************************************************************>
<!-- #INCLUDE FILE="DataConn.asp" -->

<% ' Clear Session variable
Session.Contents.RemoveAll
' Record start time
Session("Start") = Now

' Keep static to database file
' Connect to the database
Dim objRS, objBrowser
    Set objRS = Server.CreateObject("ADODB.Recordset")
    objRS.Open "WebStats", objConn, adOpenForwardOnly, adLockOptimistic, adCmdTable
' Add new record
    objRS.AddNew

' Populate data
    Set objBrowser = Server.CreateObject("MSWC.BrowserType")
    objRS("Browser") = Left(objBrowser.Browser,20)
    objRS("Version") = Left(objBrowser.Version,20)
    objRS("RemoteAddress") = Left(Request.ServerVariables("REMOTE_ADDR"),30)
    objRS("RemoteHost") = Left(Request.ServerVariables("REMOTE_HOST"),50)
    objRS("HttpRef") = Left(Request.ServerVariables("HTTP_REFERER"),50)
    objRS("SessionID") = Session.SessionID
    objRS("ActiveDate") = Now
    objRS.Update
' Close database
objRS.Close
Set objRS = Nothing
Set objBrowser = Nothing
%

<HTML>
<BASEFONT FACE="Century Gothic">
<HEAD>
<TITLE>B2Beca -- Project</TITLE>

<!-- Count before redirect to browser.asp -->
<SCRIPT LANGUAGE="JavaScript">
<!—
var start=new Date();
start=Date.parse(start)/1000;
var counts=5;
function CountDown(){
 var now=new Date();
 now=Date.parse(now)/1000;
 var x=parseInt(counts-(now-start),10);
 if(document.frm){document.frm.Count.value = x;}
 if(x>0){
 timerID=setTimeout("CountDown()", 100)
 }else{
 location.href="browser.asp"
 }
//-->}
</SCRIPT>
</HEAD>
<BODY BGCOLOR="#000080" TEXT="#33bbbb" ALINK="#cc3333" LINK="#ddff00" VLINK="#cc33ff" TOPMARGIN="0" LEFTMARGIN="0"></B>/** Show project information **>
</BR>
<BR>
<BR>
<HR>
</CENTER>This program is a part of ADMN998 Comprehensive Project.<BR>

In Partial Fulfillment of the Requirements of the Degree<BR>
<A HREF="http://www.spba.csusb.edu/mba/"
onMouseOver="window.status='SPBA';return true"
onMouseOut="window.status='';return true">Master of Business Administration: </A> Information Management<BR>
Fall 2000, <A HREF="http://www.csusb.edu"
onMouseOver="window.status='CSUSB';return true"
onMouseOut="window.status='';return true">California State University, San Bernardino</A><BR>

<TABLE BORDER=0 WIDTH=600>
<TR>
 <TD WIDTH=100 ROWSPAN=4>&nbsp;</TD>
</TR>
</TABLE>
<table>
<thead>
<tr>
<th>Committee Chair:</th>
<th>Professor</th>
<th>A href=&quot;<a href="mailto:flin@csusb.edu">mailto:flin@csusb.edu</a>&quot;</th>
<th>Frank M. Lin, Ph.D.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Second Faculty Reader:</td>
<td>Professor</td>
<td>A href=&quot;<a href="mailto:hdyck@csusb.edu">mailto:hdyck@csusb.edu</a>&quot;</td>
<td>Harold Dyck, Ph.D.</td>
</tr>
<tr>
<td>Department Chair:</td>
<td>Professor</td>
<td>A href=&quot;<a href="mailto:wstewart@csusb.edu">mailto:wstewart@csusb.edu</a>&quot;</td>
<td>Walter T. Stewart, Jr., Ph.D.</td>
</tr>
</tbody>
</table>

Create and Design By: Sumuscha Teesri

Copyright © 2000 All Rights Reserved

**Run countdown**

<SCRIPT language="JavaScript">!
```
window.setTimeout('CountDown()',10);
```
</SCRIPT>

181
FAQ.asp

<%@ Language=VBScript %>
<*****************************************************
* FAQ.asp
* Program by Sumuscha Teesri
* Frequently Ask Question about Macrotoys
* May 4, 2000
*****************************************************>
<!-- INCLUDE FILE="Top.asp" -->
<!-- INCLUDE FILE="BrowseBar.asp" -->

<!------ General HTML tag -->
<!------ Edit this file as needed -->
<!------ Begin main -->

<UL>
  <LI><A HREF="#Q1">What is the best way to shop at Macrotoys?</A>
  <LI><A HREF="#Q2">Is it safe to use my credit card?</A>
  <LI><A HREF="#Q3">Does Macrotoys charge sales tax?</A>
  <LI><A HREF="#Q4">Can people outside of the U.S. shop at Macrotoys?</A>
  <LI><A HREF="#Q5">How does the order process work?</A>
  <LI><A HREF="#Q6">How can I change or cancel my order?</A>
  <LI><A HREF="#Q7">How and when are confirmations sent out?</A>
  <LI><A HREF="#Q8">I didn't receive an order and/or shipping confirmation. What do I do?</A>
  <LI><A HREF="#Q9">Why was my order back ordered and when will it get here?</A>
  <LI><A HREF="#Q10">How can I check on the status of my order or track my package?</A>
</UL>

What is the best way to shop at Macrotoys? For instance, you can start with an age, a toy brand, or a feature like our Doll Boutique. You can also enter one of our seven stores directly by clicking on the tabs across the top of every page. You can use search to find exactly what you're looking for. You can search for products using our search box. When you're ready to make a purchase, we make it easy for you to order and pay for your items. For more details, read about how to place an order.

182
Is it safe to use my credit card? Absolutely! In fact, it's safer to use your credit card over the Internet than in a restaurant or department store.

Does Macrotoys charge sales tax? State laws require a California-based business (such as Macrotoys) to collect sales tax on orders shipped to California addresses. In our case, the amount of sales tax collected is based on the tax rate for the City of Commerce, California. We are currently not required to collect sales tax on orders shipped to residents of any other states.

Can people outside of the U.S. shop at Macrotoys? Macrotoys currently ships to the United States, Canada, American Samoa, Federated States of Micronesia, Guam, Marshall Islands, Northern Mariana Islands, Palau, Puerto Rico, the Virgin Islands, and APO/FPO U.S. Military Addresses. We are also planning to begin service to additional U.S. Territories and other international locations soon.

How does the order process work? When you see an item you want to buy, click on "add to cart." This places the item in your shopping cart. To continue shopping, select "return to shopping." When you're finished shopping, select "checkout" from the shopping cart page. In Checkout, you will be asked for your name, and shipping address. Next, you'll be shown the total amount of your order, including sales tax (for shipments to California addresses only) and shipping charges. If the information is correct, click "submit order." If any of the information is incorrect, you can click back and adjust.
information on previous pages. You can also cancel the order at this point. Once you have verified everything, click on "submit order." After a few seconds, you'll see a page confirming your order and giving you an Order Number to use if you need to contact us. We will also confirm your order via e-mail within 24 hours. &nbsp; Once you have verified everything, click on "submit order." After a few seconds, you'll see a page confirming your order and giving you an Order Number to use if you need to contact us. We will also confirm your order via e-mail within 24 hours. <IMG BORDER="0" SRC="image\UP.gif"></A><P><A NAME="Q6"></A><STRONG>How can I change or cancel my order? Our goal is to provide the fastest possible service to our customers, and we try to ship orders out as quickly as possible. This makes it extremely difficult to cancel or make changes to orders. There is a good possibility that by the time you call with a change or cancellation, your package could already be on its way to you! If you would like to make a change or cancel your order, please contact us as soon as possible. <IMG BORDER="0" SRC="image\UP.gif"></A><P><A NAME="Q7"></A><STRONG>How and when are confirmations sent out? Macrotoys order confirmations are sent out via e-mail automatically. You will receive your order confirmation within 24 hours of placing your order. You will receive your shipping confirmation once your package has shipped from our warehouse. The shipping confirmation may include a tracking number if the carrier provides this service. <IMG BORDER="0" SRC="image\UP.gif"></A><P><A NAME="Q8"></A><STRONG>I didn't receive an order and/or shipping confirmation. What do I do? If you have not received your order confirmation within 2 days of your order, please let us know. We will make every effort to respond within 1 business day regarding the status of your order. <IMG BORDER="0" SRC="image\UP.gif"></A><P><A NAME="Q9"></A><STRONG>Why was my order backordered and when will I get it?</P>
Our product availability policy is designed so when we say an item is in stock, we mean it. In a few cases, however, we may take your order and find out that our inventory is not what we thought it was (i.e. an item was damaged or defective). In this case, we will notify you via e-mail that it is on backorder and do our utmost to fulfill the order as soon as possible at no additional charge. If you would like us to cancel the item, just contact us by phone or e-mail.<

How can I check the status of my order or track my package? If you would like to check on the status of your order or track your package(s), go to "Check Orders" and click "Order Status." This will allow you to see the complete history of your order, including exactly when it was shipped and its current status. This is the easiest and fastest way to get the most up-to-date information on your order. Or, feel free to e-mail us.

ForgetPassword.asp

%@ Language=VBScript %>
<%@Include File="Top.asp" %>
<%@Include File="BrowseBar.asp" %>
<!-------------------------- -->
<%*
Begin main
%-->
Response.Write "We can not find your email Please try again."<BR>
Else
Response.Write "We will send your Username and Password to your email address."
End If
%
</FORM ACTION="SendPassword.asp">
Please enter you email address:<BR>
</INPUT TYPE="TEXT" NAME="Email" SIZE="40"/><BR>
</INPUT TYPE="SUBMIT" VALUE="Send" NAME="SUBMIT"&nbsp;&nbsp;
</INPUT TYPE="RESET" NAME="RESET">

<!-- #INCLUDE FILE="Bottom.asp" -->

Global.asp

<SCRIPT LANGUAGE=VBScript RUNAT=Server>
'Include file needed for send mail
<!--METADATA TYPE="TypeLib" NAME="Microsoft CDO for NTS 1.2 Library"
VERSION="1.2"-->
'******************************************************************************
'* Global.asp
'* Program by Sumuscha Teesri
'* Global file provide setup variable
'* May 4, 2000
'******************************************************************************

Sub Application_OnStart
  Crate Visits and Active counter
  Application("Visits") = 0
  Application("Active") = 0
  Application("IISAd") = 0
  Application("BackOfficeAd") = 0
  Application("WebAndHostAd") = 0
End Sub

Sub Application_OnEnd
End Sub

Sub Session_OnStart
Dim strStartPage, strCurrentPage

  ' Increase counters when user connect to the website
  Application.Lock
  Application("Active") = Application("Active") + 1
  Application("Visits") = Application("Visits") + 1
  Application.UnLock

  ' Make sure that user will alway start with the default page
  strStartPage = "default.asp"
  strCurrentPage = Request.ServerVariables("SCRIPT_NAME")

186
'Do a case-insensitive compare, and if they 'don't match, send the user to the start page.
If strcmp(strCurrentPage,strStartPage,1) then
  ' Turn on buffer for compactible
  Response.Buffer = True
  Response.Clear
  Response.Redirect(strStartPage)
End If

End Sub

Sub Session_OnEnd
  ' Reduce active counter when user logout
  Application.Lock
  Application("Active") = Application("Active") - 1
  Application.Unlock
End Sub
</SCRIPT>

LogIn.asp

<%@ Language=VBScript %>
<%!***********************************************************************
  * LogIn.asp
  * Program by Sumuscha Teesri
  * Current customer log in to the system
  * May 4, 2000
  ***********************************************************************%>
<!-- #INCLUDE FILE="Top.asp" -->
<!-- #INCLUDE FILE="BrowseBar.asp" -->

<*** Verify that user provide needed information***>
<SCRIPT Language="JavaScript">
<!--
  function VerifyData()
  {
    if (document.frmLogIn.LogInName.value == "")
    {
      alert("Please enter your Log in name");
      return false;
    }
    else
    {
      if (document.frmLogIn.Password.value == "")
      {
        alert("Please enter your Password");
        return false;
      }
      else
      {
        return true;
      }
    }
  }-->
</SCRIPT>
<CENTER><FONT COLOR="#22FF22"><H1>Member LogIn</H1></FONT></CENTER>

<!--*
Begin main *
-->

<% ' Check whether user fail to log in
  ' Recieve information form CheckLogin.asp
  Response.Write "Please enter your user name and password to
  login."<BR>
  Response.Write "New User please <A HREF="register.asp"">Click Here</A>.<BR>
  If Request("LogFail") <> "" Then
    ' User provide wrong password
    If Request("WrongPW") = "True" Then
      Response.Write "Invalid Password. Please try again, or
    Use link below<BR>"
    Else
      ' User provide wrong login name
      Response.Write "User LogIn Name not found. Please try
    again:"<BR>"
    End If
  End If
  ' Warning message
  If Request("LogFail") = "Second" Then
    ' If this is the last try
    Recieve SECOND form checklogin means already try the
  second
    Response.Write "<BP><B>This is your last try</BP></B>" 
    ' ELSE
    ' If this is the first or second try
    Response.Write "Please enter your user name and password to
  login."<BR>"
  End If
%>

<!** Create input form **>

<FORM ACTION="CheckLogIn.asp" NAME="frmLogln"
  onSubmit="return VerifyData()" METHOD="POST">
  <INPUT TYPE="HIDDEN" NAME="LogFail" VALUE=
    <% Send the time user try to login to
    checklogin
      Select Case Request("LogFail")
        Case "First"
          Response.Write "Second"
        Case "Second"
          Response.Write "Last"
        Case Else
          Response.Write ""
      End Select%>>

  <B>User LogIn Name:</B><BR>
  <INPUT TYPE="TEXT" NAME="LogInName"
    <% If Request("LogFail") <> "" Then %>
    VALUE="<%= Session("LogInName")%>"
    <% End IF %>
    </BODY>
</HTML>
**Password:**

```html
<input type="password" name="Password" size="40">
```

**Submit Button:**

```html
<input type="submit" value="Login">
```

**Reset Button:**

```html
<input type="reset" name="RESET">
```

**Forget Password or Username please:**

```
[Forget Password](ForgetPassword.asp)
```

---

**Message.asp**

```vbscript
<!-- #INCLUDE FILE="Bottom.asp" -->
```

**ModifyItem.asp**

```vbscript
<!-- #INCLUDE FILE="Bottom.asp" -->
```

---

```vbscript
Response.Buffer = True
' Declaration and set necessary variable
Dim intCartSize, intCartProd(), intCartQuan()
```
Dim objItem, intLoop, intLoop2, intLoop3
Dim intCountItems, intCountDeleteItems

' Set counter value
intCountItems = Cint(Request("Quantity").Count)
intCountDeleteItems = Cint(Request("DeleteItem").Count)

' Customer Empty the cart.
If intCountDeleteItems = intCountItems Then
    Session.Contents.Remove("Shopping")
    Session.Contents.Remove("CartProd")
    Session.Contents.Remove("CartQuan")
    Response.Clear
    Response.Redirect "Browser.asp"
End If

' Resize the cart in case customer want to delete some items, not empty
intCartSize = intCountItems - intCountDeleteItems - 1
ReDim intCartProd(intCartSize), intCartQuan(intCartSize)

'********************
'*
' The following code used for debug and testing only.
'*
'********************

'For Each objItem In Request.Form
    If Request.Form(objItem).Count > 1 Then
        Response.Write objItem & ":<BR>"
        For intLoop = 1 To Request.Form(objItem).Count
            Response.Write "SubKey " & intLoop & " Value = " & _
            Request.Form(objItem)(intLoop) & _
            "<BR>"
        Next
'    Else
'        Response.Write objItem & " = " & Request.Form(objItem) & _
'        "<BR>"
'    End If
'Next

'For i = 1 to Request("DeleteItem").Count
'    Response.Write "DeleteItem & i & ": & Request("DeleteItem")(i) & _
'    "<BR>"
'Next

Response.Write "CartSize : " & intCartSize & "<BR>" & _
          "CountItems: " & intCountItems & "<BR>" & _
          "CountDeleteItems: " & intCountDeleteItems & "<BR>"
'******************************************************************************
'*
' End debugging code
'******************************************************************************

' Customer change shopping something to the cart
CartProd = Session("CartProd")

' Customer want to delete and update some items.
If intCountDeleteItems > 0 Then
    ' intLoop is the Counter for old shopping cart size.
    intLoop=0

190
' intLoop2 is the Counter for delete items.
intLoop2=1
' intLoop3 is the Counter for new shopping cart size.
intLoop3=0

' Check whether the new cart size is full
While intLoop3 <= intCartSize
  ' Check whether delete all item need
  ' Or all items in the old cart is select
  While Cint(Request("DeleteItem")\(intLoop2\)) <> intLoop
    and intLoop < intCountItems
    intCartProd\(intLoop3\) = CartProd\(intLoop\)
    intCartQuan\(intLoop3\) = Request("Quantity")\(intLoop+1\)
    ' Move to next cart slot
    intLoop3=intLoop3+1
  ' Select next item to put in the cart
  intLoop = intLoop + 1
  Wend
  ' Move to next delete item
  intLoop = intLoop + 1
  If intLoop2 < intCountDeleteItems Then intLoop2 = intLoop2 + 1
  Wend
Else
  ' Customer don't delete any product, just modify quantity
  For intLoop=0 To intCartSize
    intCartProd\(intLoop\) = CartProd\(intLoop\)
    intCartQuan\(intLoop\) = Request("Quantity")\(intLoop+1\)
  Next
End If

****************************************************************
'* For Debug and testing only
' For intLoopp=0 To intCartSize
  Response.Write "ProdID : " & intCartProd\(intLoop\) & _
  " Quantity : " & intCartQuan\(intLoop\) & _
  "<BR>"
' Next

' Then update the shopping cart information into session variable
Session("CartProd") = intCartProd
Session("CartQuan") = intCartQuan

' Redirect to the purpose page
Response.Clear
If Len(Request.Form("UpdateItem")) Then Response.Redirect "ShoppingCart.asp"
If Len(Request.Form("Continue")) Then Response.Redirect "Browser.asp"
If Len(Request.Form("Checkout")) Then Response.Redirect "Shipping.asp"
%>

191
NonBrowserBar.asp

<!*******************************
* NonBrowserBar.asp
* Program by Sumuscha Teesri
* NonBrowserBar Template
* May 4, 2000
*******************************>

<!--*
In case need top and bottom template
but not the navigation bar
-->**>

<!--
End of nonbrowsbar.asp
-->

OrderDetail.asp

%@ Language=VBScript %>
<!*******************************
* OrderDetail.asp
* Program by Sumuscha Teesri
* May 10, 2000
*******************************>

<!--This is template file please delete and enter code here-->

% Response.Buffer = True

' If customer does not log in redirect
If Session("CusID") ="" Then
    Response.Clear
    Response.Redirect "Message.asp?Msg=Please log in or register with our system" & _
    "<BR>in order to buy our products." & _
    "&redirect=login.asp"
End If

Set objRS = Server.CreateObject("ADODB.Recordset")
strSQL = "SELECT P.ProdID, ProdName, Quantity " & _
    "FROM Products P, Transacts T " & _
    "WHERE P.ProdID=T.ProdID and OrderID=" & _
CInt(Request("OrderID")) & ";"
objRS.Open strSQL, objConn, adOpenForwardOnly, adLockReadOnly, adCmdText

strMessage = "<CENTER><H1><FONT COLOR=#22FF22>Order Detail</FONT>" & _
"</Hlx/CENTER><BR><FONT COLOR=#22FF22><B>ORDER ID: " & _
Request("OrderID") & "</B><BR><BR>" & _
"<TABLE BORDER=1 WIDTH=600><TR>" & _
"<TD WIDTH=100 ALIGN=CENTE"R><B>Product ID</B></TD>" & _
"<TD WIDTH=400 ALIGN=CENTE"R><B>Name</B></TD>" & _
"<TD WIDTH=100 ALIGN=CENTE"R><B>Quantity</B></TD>" & _

While Not objRS.EOF
strMessage = strMessage & "<TD WIDTH=100 ALIGN=CENTE"R>" & _
objRS("ProdID") & _
"</TD><TD WIDTH=400 >" & objRS("ProdName")& _
"</TD><TD WIDTH=100 ALIGN=RIGHT>"&
objRS("Quantity") & _
"</TD></TR>"_

objRS.MoveNext
strMessage = strMessage & "</TABLE><BR>"
Response.Write strMessage
objRS.Close
Set objRS = Nothing
%
<a href="CheckOrder.asp"
onmouseover="document [{'fpAnimswapImgCKO'}.imgRolln=document [{'fpAnimswapImgCKO'}.src=document [{'fpAnimswapImgCKO'}.lowsrc;
window.status='Order Statue';return true;"
onmouseout="document [{'fpAnimswapImgCKO'}.src=document [{'fpAnimswapImgCKO'}.limgRolln;
window.status='';return true>
<img border="0" src="image/Back.gif" width="100" height="30"
 id="fpAnimswapImgCKO"
name="fpAnimswapImgCKO" dynamanimation="fpAnimswapImgCKO"
lowsrsrc="image/Back0V.gif"></A>
<!-- #INCLUDE FILE="Bottom.asp" -->

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml">
<body>

Payment.asp

<% @ Language=VBScript %>

******************************************************************************
* Payment.asp
* Program by Sumuscha Teesri
* May 9, 2000
******************************************************************************
<%
Response.Buffer = True
If Session("CusID") ="" Then
    Response.Clear
    Response.Redirect "Message.asp?Msg=Please log in or register with our system" & _
"in order to buy our products." & _
"&redirect=login.asp"

</body>
<*/
function VerifyData()
{
    var d, m, y, AccountNumber
    d = new Date();
    m = d.getMonth();
    y = d.getYear();

    // Check account number buy convert strings to number
    AccountNumber = new Number(document.Payment.AccNum1.value +
                                document.Payment.AccNum2.value +
                                document.Payment.AccNum3.value +
                                document.Payment.AccNum4.value);

    // Check is the valid account is it not null and length 16 or more
    if (isNaN(AccountNumber) || AccountNumber < 999999999999999)
    {
        // Invalid account show message
        alert("Invalid Account Number");
        return false;
    }

    // Check expiration date
    if (document.Payment.ExpYear.value <= y)
    {
        // Check expiration month
        if (document.Payment.ExpMonth.value < m)
        {
            // Invalid month
            // For debug only
            // alert(document.Payment.ExpYear.value + " " + y);
            alert("Please enter the correct Expiration date");
            return false;
        }
    }
    else
    {
        // For debug only
        // alert(document.Payment.ExpYear.value + " " + y);
        // alert("Everything is just fine");
        // return false;

        // Everything is fine now.
        return true;
    }
    -->
</SCRIPT>
<!-- Include file needed -->
<!-- #INCLUDE FILE="Top.asp" -->
<!-- #INCLUDE FILE="BrowseBar.asp" -->
194
<% Find the carrier
Set objRS=Server.CreateObject("ADODB.Recordset")
objRS.Open "PaymentTypes", objConn, adOpenStatic, adLockReadOnly, adCmdTable
%
<!--**--->
<="
<-- **************************** -->
<!--**--->

<FONT COLOR="#22FF22" >Payment
Information</FONT></H2></CENTER>
<!--This is template file, please delete and enter code here-->

<FORM NAME=Payment ACTION="checkout.asp" onSubmit="return VerifyData()"
METHOD=POST>
Please Select one of the following payment types.<BR>
<Select NAME=PaymentType>
<% strMessage = strMessage & 
"<OPTION VALUE=" & objRS("ID") & 
" SELECTED" & objRS("Description") & 
"</OPTION>"
objRS.MoveNext
While Not objRS.EOF
strMessage = strMessage & 
"<OPTION VALUE=" & objRS("ID") & 
"" & objRS("Description") & 
"</OPTION>"
objRS.MoveNext
Wend
objRS.Close
Set objRS = Nothing
Response.Write strMessage
For Each item In Request.Form
  Response.Write Request.Form(item)
Next
Response.Write Request("TotalPrice")
%>
</SELECT><BR>
<TABLE BORDER=0 WIDTH = 550>
<TR><TD WIDTH=300>&nbsp;</TD>
<TD COLSPAN=2 ALIGN=LEFT>Date of Expiration</TD>
</TR>
<TR><TD WIDTH=300>Account Number </TD>
<TD WIDTH=100>Month</TD>
<TD WIDTH=80>Year</TD>
</TR>
<TR><TD>
<INPUT TYPE=TEXT NAME=AccNum1 MAXLENGTH=4 SIZE=4>&nbsp;
<INPUT TYPE=TEXT NAME=AccNum2 MAXLENGTH=4 SIZE=4>&nbsp;
<INPUT TYPE=TEXT NAME=AccNum3 MAXLENGTH=4 SIZE=4>&nbsp;
<INPUT TYPE=TEXT NAME=AccNum4 MAXLENGTH=4 SIZE=4>
</TD>
<TD>
<Select NAME=ExpMonth ID=ExpMonth>
  <OPTION VALUE=1 SELECTED>JANUARY</OPTION>
</SELECT>
</TD>
<OPTION VALUE=2>FEBRUARY</OPTION>
 <OPTION VALUE=3>MARCH</OPTION>
 <OPTION VALUE=4>APRIL</OPTION>
 <OPTION VALUE=5>MAY</OPTION>
 <OPTION VALUE=6>JUNE</OPTION>
 <OPTION VALUE=7>JULY</OPTION>
 <OPTION VALUE=8>AUGUST</OPTION>
 <OPTION VALUE=9>SEPTEMBER</OPTION>
 <OPTION VALUE=10>OCTOBER</OPTION>
 <OPTION VALUE=11>NOVEMBER</OPTION>
 <OPTION VALUE=12>DECEMBER</OPTION>

</SELECT>
</TD>
</TR>
</TABLE>
</FORM>

<!-- #INCLUDE FILE="Bottom.asp" -->
<!*****************************************************************************>

ProductDetail.asp

<!-- Language=VBScript -->
<!-- ***************************************************************************** -->
* ProductDetail.asp
* Program by Sumuscha Teesri
* May 7, 2000
***************************************************************************** -->

%! Include file needed >
<!-- #INCLUDE FILE="Top.asp" -->
<!-- #INCLUDE FILE="BrowseBar.asp" -->

<!--***************************************************************************** -->
<!-- Begin main -->
<!--***************************************************************************** -->

<CENTER><B><FONT COLOR="#22FF22">Product Detail</FONT></B></CENTER>
<% Set objRS = Server.CreateObject("ADODB.Recordset")
strSQL = "SELECT ProdID, ProdName, Description, SalePrice, ", " & 
"StockQuantity, Picture, SaleUnit, Measure " & 

196
FROM Products " & _
"WHERE ProdID" & Request(ProdID) & ";"

objRS.Open strSQL, objConn, adOpenForwardOnly, adLockReadOnly, adCmdText

If Not objRS.EOF Then
    Dim strMessage, strStock, intMeasure
    intMeasure = objRS("Measure")
    if objRS("StockQuantity") > 0 Then
        strStock = "In Stock"
    Else
        strStock = "Out of Stock"
    End If
    strMessage = "<TABLE BORDER=1 WIDTH=600>" & _
        "<TR><TD WIDTH=450>" & objRS("ProdID") & ": " & _
        ObjRS("ProdName") & "</TD>" & _
        "<TD ALIGN=LEFT WIDTH=150>" & strStock & _
        "</TD></TR>" & _
        "<TR><TD WIDTH=450>" & objRS("Description") & _
        "</TD><TD VALIGN=TOP WIDTH=150>" & _
        "<IMG SRC=image\product" & objRS("Picture") & _
        "WIDTH=150</br></CENTER>Price Per Unit <br>$" & _
        objRS("SalePrice") & _
    "</CENTER></TD></TR></TABLE>"
    Response.Write strMessage
End If
objRS.Close
Set objRS = Nothing

<form action="AddItem.asp" method="post">
    cInput type=hidden name="ProdID" value="<%=Request(ProdID)%>">
    This product is sold in a box of <b><%=intMeasure%> items per box.
    </b>
    select name="Quantity">
    <% For i = 1 To 10
        Response.Write "<option>" & i*intMeasure & "</option>"
    Next
    %>
</select>
    <input type=submit value="Buy">
</form>

</script>

{% include file="Bottom.asp" --}
<!----------------------------------

197
Register.asp

<% @ Language=VBScript %>
<%******************************************************************************
* Register.asp
* Program by Sumuscha Teesri
* May 24, 2000
******************************************************************************>
<!-- #INCLUDE FILE="Top.asp" -->
<!-- #INCLUDE FILE="BrowseBar.asp" -->

<%! Varify Password is not null and
Password = VarifyPassword %>

<SCRIPT Language="JavaScript">
<!--
function VerifyData()
{
   //Password = VarifyPassword
   if (document.frmUser.Password.value !=
   document.frmUser.VerifyPassword.value)
   {
      alert("Your passwords do not match - please re-
      enter");
      return false;
   }
   else
   {
      //Password is not Null
      if (document.frmUser.Password.value == "")
      {
         alert("Your need to provide a password");
         return false;
      }
      else
         return true;
   }

-->
</SCRIPT>

<!--******************************************************************************-->
<!--* Begin main *-->
<!--******************************************************************************-->

<% 'Print Page head for login user or new register user
Dim strMsg
strMsg = "<CENTER><FONT COLOR=#22FF22>"
If Request("Edit") = "True" Then
   strMsg = strMsg & "<H1>Update User Registration</H1>"
   Session("Update") = True
Else
   strMsg = strMsg & "<H1>New User Registration</H1>"
End If

198
strMsg = strMsg & "</CENTER></FONT><p>"

' Show Important message if user already login
If Request("Edit")= "True" Then
    strMsg = strMsg & "Please change your registration information as listed below."<P>
Else If Request("NotFound") = "True" Then
    strMsg = strMsg & "<I>We were unable to locate your information." & _
               "Please take the time to register again.</I><P>"
Else
    strMsg = strMsg & "<CENTER>If you're already registered with us," & _
             "then click the 'Member Login' link below.</CENTER><P>"
End If
strMsg = strMsg & "In order to use Macrotoys' services, please take " & _
        "a few minutes to complete the form below. Once you" & _
        "have done that, you will have full access to our system."
End If
Response.Write strMsg

' Show error message if information is not unique in the needed fields
Dim strErrMsg1, strErrMsg2, strNull, strErr
' Create template message
strErrMsg1 = "Your "
strErrMsg2 = " already in the system\n\n" & _
               "Please Provide the new "
' Check unique field error message
If Request("ComUnique") = "False" Then
    strErrMsg1 = strErrMsg1 + "Company Name, 
    strErrMsg2 = strErrMsg2 + "Company Name, 
End If
If Request("LogInUnique") = "False" Then
    strErrMsg1 = strErrMsg1 + "Login Name, 
    strErrMsg2 = strErrMsg2 + "Login Name, 
End If
If Request("EmailUnique") = "False" Then
    strErrMsg1 = strErrMsg1 + "Email, 
    strErrMsg2 = strErrMsg2 + "Email, 
End If
If Len(strErrMsg1) > 5 Then
    strErrMsg1 = Left(strErrMsg1,Len(strErrMsg1)-2)
    strErrMsg2 = Left(strErrMsg2,Len(strErrMsg2)-2)
End If
For Each Item in Request.QueryString
    If Request.QueryString(Item) = "True" and Item <> "Edit" Then
        strNull = "True"
End If

199
If Request.QueryString(Item) = "False" Then strErr = "True"
Next

' There are some error

If Len(strErrMsg1) > 5 or strNull = "True" Then

<!-- Run script to show error message -->
<SCRIPT LANGUAGE="JavaScript">
<!--
var strErrMsg, strNull, strErr;
// Receive message from Asp and print in Java
strNull = "<%= strNull %>";
strErr = "<%= strErr %>";
strErrMsg = "<%= strErrMsg1 & strErrMsg2 %>";
if (strNull == "True") { alert("Please provide missiong fields");}
if (strErr == "True") {alert(strErrMsg)};
-->
</SCRIPT>
%

<!-- End If -->

<HR>

<FONT COLOR="#22FF22">* Request information*</FONT>

<!** Create Input Form and call VerifyData() before send information**>
<FORM ACTION="CheckRegisterInfo.asp" NAME="frmUser" METHOD="POST"
onSubmit="return VerifyData()">
<TABLE BORDER=0>
    <TR><TD WIDTH=150 ROWSPAN=2><B>Company Information</B></TD></TR>
    <TD WIDTH=150><% 'Check wheather User forget this field
        If Request("ComNameFail") = "True" Then
            Response.Write "<FONT COLOR=#FF0000>" & _
            "* Company Name:"/&"</TD>
        Else
            Response.Write "* Company Name:"/&"</TD>
    End If
%
</TR>
    <TR><TD><% 'Check wheather User forget this field
        If Request("AddFail") = "True" Then
            Response.Write "<FONT COLOR=#FF0000>" & _
            "* Address:"/&"</TD>

</TR>
</FORM>
Else
    Response.Write "* Address:<TD>
End If%
</TD>
<INPUT TYPE="Text" NAME="Address1" VALUE="<%
Session("Address1")%>
SIZE="30">
</TD>
</TR>
<TR>
<TD ROWSPAN=8>&nbsp;</TD>
<TD></TD>
<INPUT TYPE="Text" NAME="Address2" VALUE="<%
Session("Address2")%>
SIZE="30">
</TD>
</TR>
'Through User forget this field
If Request("CityFail") = "True" Then
Response.Write "<FONT
COLOR=#FF0000>" & _
** City:<TD>/FONT>"
Else
Response.Write "* City:<TD>"
End If%
</TD>
<INPUT TYPE="Text" NAME="City" VALUE="<%
Session("City")%>
SIZE="30">
</TD>
</TR>
<TD>Region:</TD>
<INPUT TYPE="Text" NAME="Region" VALUE="<%
Session("Region")%>
SIZE="30">
</TD>
</TR>
<TD>Postal Code:</TD>
<INPUT TYPE="Text" NAME="PostalCode" VALUE="<%
Session("PostalCode")%>
SIZE="30">
</TD>
</TR>
'Through User forget this field
If Request("CountryFail") = "True" Then
Response.Write "<FONT
COLOR=#FF0000>" & _
** Country:<TD>/FONT>"
Else
Response.Write "* Country:<TD>"
End If%
</TD>
<INPUT TYPE="Text" NAME="Country" VALUE="<%
Session("Country")%>
SIZE="30">
</TD>
</TR>
<TD>% 'Check wheather User forget this field
If Request("PhoneFail") = "True" Then
Response.Write "<FONT
COLOR=#FF0000>" & _
& "201
Else
Response.Write "* Phone:"<TD>
End If%>
<TD><INPUT TYPE="Text" NAME="Phone" VALUE="<%= Session("Phone") %>"
SIZE="30"></TD>

"Fax:"<TD>
<TD><INPUT TYPE="Text" NAME="Fax" VALUE="<%= Session("Fax") %>"
SIZE="30"></TD>

Website:"<TD>
<TD><INPUT TYPE="Text" NAME="WebSite" VALUE="<%= Session("Website") %>"
SIZE="30"></TD>

Contact Information:<B><TD><B>
<TD WIDTH=150 ROWSPAN=2><FONT COLOR="#22FF22"><B>Contact Information:</B><TD><B></TD><B>
<TD>Title:"<TD>
<TD><INPUT TYPE="Text" NAME="Title" VALUE="<%= Session("Title") %>"
SIZE="30"></TD>

First Name:"<TD>
<TD>% 'Check wheather User forget this field
If Request("FirstFail") = "True" Then Response.Write "<FONT COLOR=#FF0000>" & _
** First Name:"<TD><FONT>"
Else Response.Write "* First Name:"<TD>"
End If%>
<TD><INPUT TYPE="Text" NAME="FirstName" VALUE="<%= Session("FirstName") %>"
SIZE="30"></TD>

Last Name:"<TD>
<TD>% 'Check wheather User forget this field
If Request("LastFail") = "True" Then Response.Write "<FONT COLOR=#FF0000>" & _
** Last Name:"<TD><FONT>"
Else Response.Write "* Last Name:"<TD>"
End If%>
<TD><INPUT TYPE="Text" NAME="LastName" VALUE="<%=
Session("LastName")%>" SIZE="30"></TD>
</TR>
<TR>
  <TD><% 'Check wheather User forget this field
   If Request("EmailFail") = "True" Then
   Response.Write "<FONT COLOR=#FF0000>" & _
   "* Email Address:</TD></FONT>"
   Else
   Response.Write "* Email Address:"</TD>
  <TD><INPUT TYPE="Text" NAME="Email" VALUE='<%=Session("Email")%>" SIZE="30"></TD>
</TR>
<TR>
  <TD><% 'Direct Line:
  If Request("LogInName") = "True" Then
  Response.Write "<FONT COLOR=#FF0000>" & _
  "* LogIn Name:"</TD></FONT>"
  Else
  Response.Write "* LogIn Name:"</TD>
  <TD><INPUT TYPE="Text" NAME="Direct" VALUE='<%=Session("Direct")%>" SIZE="30"></TD>
</TR>
<TR><TD COLSPAN=3>&nbsp;</TD></TR>
<TR>
  <TD WIDTH=150 ROWSPAN=2><FONT COLOR="#22FF22">B>
  Security Information:</TD></FONT></B>
  <TD><% 'Check wheather User forget this field
   If Request("LogInName") = "True" Then
   Response.Write "<FONT COLOR=#FF0000>" & _
   "* LogIn Name:"</TD></FONT>"
   Else
   Response.Write "* LogIn Name:"</TD>
  <TD><INPUT TYPE="Text" NAME="LogInName" VALUE='<%=Session("LogInName")%>" SIZE="30"></TD>
</TR>
<TR>
  <TD>* Password:</TD>
  <TD VALIGN=bottom><INPUT TYPE="Password" NAME="Password" VALUE='<%=Session("Password")%>" SIZE="30"></TD>
</TR>
<TR>
  <TD>* Verify Password:</TD>
  <TD><INPUT TYPE="Password" NAME="VerifyPassword" VALUE='<%=Session("VerifyPassword")%>" SIZE="30"></TD>
</TR>
button depend on user

If Request("Update") = "True" Then
    Response.Write "Submit Information"
Else
    Response.Write "Submit Registration"
End If%>

NAME=RESET></TD>
</TR>
</TABLE>
</FORM>
<!-#INCLUDE FILE="Bottom.asp" -->

---

Search.asp

<%@ Language=VBScript %> 
<!--
* Search.asp
* Program by Sumuscha Teesri
* May 7, 2000
*------------------------------->

<! Include file needed >
<!-- #INCLUDE FILE="Top.asp" -->
<!-- #INCLUDE FILE="BrowseBar.asp" -->

<!-----------------------------------> 
<!-- Begin main --> 
<!-----------------------------------> 

><CENTER><B><FONT COLOR="#22FF22">Search Product</FONT></B></CENTER>

'Connect to the database using SQL statement 
'-------------------------------
'Dim objRS : already declare in browserbar

If Request("SearchKey") <> "" Then
Dim strSearchKey
    strSearchKey = ";" & Request("SearchKey") & ";" 
    Set objRS = Server.CreateObject("ADODB.Recordset")
    strSQL = "SELECT ProdID, ProdName, Picture, SalePrice " & _
        "FROM Products " & _
        "WHERE Description LIKE " & strSearchKey & _
        " ORDER BY ProdID"
' Open Database with the query
    objRS.Open strSQL, objConn, adOpenStatic , adLockReadOnly,
    adCmdTxt
    objRS.PageSize = 3

'-------------------------------

204
'Write information
If Not objRS.EOF Then

   ' Set Page Number
   Dim PageMove
   PageMove = Request("PageMove")
   If Request("PageNo") = "" Then
      PageNo=1
   Else
      PageNo = Request("PageNO")
   End If

   ' Set Navigation button
   If PageMove = "Previous" or PageMove = "Next" Then
      Select Case PageMove
         Case "Previous"
            If PageNo > 1 Then
               PageNo = PageNo -1
            Else
               PageNo = 1
            End If
         Case "Next"
            If objRS.AbsolutePage < objRS.PageCount Then
               PageNo = PageNo + 1
            Else
               PageNo = objRS.PageCount
            End If
         Case Else
            PageNo = 1
      End Select
   End If

   objRS.AbsolutePage = PageNo

   Dim strMessage, intItemCount
   strMessage = "<TABLE BORDER=0 WIDTH=550><TR><TD WIDTH=450>
   &<SMALL>& Found " & objRS.RecordCount & " records."</B></SMALL>"

   ' Write page number
   strMessage = strMessage & "</TD></SMALL><TD ALIGN=RIGHT WIDTH=100>" & _
   &"<SMALL>& PageNo & " of " & objRS.PageCount & _
   "</SMALL></TD></TR></TABLE>"
   Response.Write strMessage

   ' Table head
   Response.Write "<TABLE BORDER=1 WIDTH=550><TR>" & _
   &"<TD ALIGN=LEFT WIDTH="30">"<B>ID</B></TD>" & _
   &"<TD ALIGN=LEFT WIDTH="452">"<B>Product</B></TD>"
   & _
   &"<TD ALIGN=LEFT WIDTH="60">"<B>Price</B></TD>" & _

205
Loop 3 record per page
For intItemCount = 1 to objRS.PageSize
  ' Write table body
  Response.Write "<TR><TD WIDTH="30" VALIGN=TOP ALIGN=CENTER>"
  HREF=ProductDetail.asp?ProdID=" & objRS("ProdID") & "">
  objRS("ProdID") & ";"</A></TD>" & _
  "<TD VALIGN=TOP WIDTH="452" width=125>" & _
  HREF=ProductDetail.asp?ProdID=" & objRS("ProdID") & ">" & objRS("ProdName")
  & ";<TD ALIGN=RIGHT WIDTH=60;">" & _
  objRS("SalePrice") & "<A HREF=http://www.batavia.com/ProdID=" & _
  BORDER=0 SRC=product\" & _
  WIDTH=58 HEIGHT=72="/A"></TD></TD>"<TR>
  objRS.MoveNext
  ' No more record
  If objRS.EOF Then Exit for
Next

Response.Write "</TABLE>"
strMessage = "PageNo=" & PageNo & ";&SearchKey=" & 
Request("SearchKey")
If PageNo > 1 or PageNo < objRS.PageCount Then
  Response.Write "<TABLE BORDER=2 WIDTH=450><TR><TD WIDTH=125 HEIGHT=30>"
End If
If PageNo > 1 Then
  'Write Animation button for the Previour record navigator
  <A HREF=search.asp?%=%= strMessage %>&PageMove=Previous" 
onmouseover="document[fpAnimswapImgPre].imgRolln=document[fpAni
mswapImgPre].src; 
document[fpAnimswapImgPre].src=document[fpAnimswapImgPre].lows 
rc; 
window.status='Previous Record';return true;"
onmouseout="document[fpAnimswapImgPre].src=document[fpAnimswapI
mgPre].imgRolln;"
window.status=''
<% End IF %>
Response.Write '</TD><TD WIDTH=100 HEIGHT=30>"'
If PageNo < objRS.PageCount Then
' Write Animation button of Next record navigator
<%>
<A HREF="search.asp?c=%=strMessage %>>&PageMove=Next"
onclick="document['fpAnimswapImgNext'].imgRolln=document['fpAnimswapImgNext'].src;
document['fpAnimswapImgNext'].src=document['fpAnimswapImgNext'].lowsrc;
window.status='Next Record';return true;"
onmouseover="document['fpAnimswapImgNext'].imgRolln=document['fpAnimswapImgNext'].src;
document['fpAnimswapImgNext'].src=document['fpAnimswapImgNext'].lowsrc;
window.status='';return true">'
<% End If %>
Response.Write '</TD><TD WIDTH=125 HEIGHT=30></TD></TR></TABLE>"
'Close record set
objRS.Close
Set objRS = Nothing
Else
' This is no information in subcategory
Response.Write "<BR><BR>" & _
"<P><B>Sorry,</B> there is no record found." & _
"<BR>Please try again.<BR>" & _
"You may find the product you want<BR><BR>"
End If
<%>
</A>
</A>
</%>
</FORM ACTION="Search.asp" METHOD="POST">
Please type your search word here<BR>
(INPUT TYPE="TEXT" NAME="SearchKey" VALUE="<%= Request("SearchKey") %>"
SIZE="30">
(INPUT TYPE="SUBMIT" VALUE="Search")
**SendMail.asp**

<!-- #INCLUDE FILE="DataConn.asp" -->

```vbscript
<%@ Language=VBScript %>
<!— #INCLUDE FILE="DataConn.asp" —>

```

' "**************************************************************************
'  SendComment.asp
'  Program by Sumuscha Teesri
'  SendComment to Macrotoys
'  April 25, 2000
'  **************************************************************************>

' Create mail object
Dim objSendMail
Set objSendMail = CreateObject("CDONTS.NewMail")
Dim strTo
' Turn buffer on for compactibility
Response.Buffer = True

' If user login send mail to representative employee
If Session("CusID") <> "" Then
    ' Connect to employee database
    Dim objRS
    Set objRS = Server.CreateObject("ADODB.Recordset")
    strSQL = "SELECT email FROM Employees WHERE EmpID=" & Session("EmpID") & ";"
    objRS.Open strSQL, objConn, adOpenForwardOnly, adLockReadOnly, adCmdText
    ' Provide representative employee email address
    strTo = objRS("email")
    ' Response.Write strTo & "<BR>"
    objRS.Close
    Set objRS = Nothing

Else
    strTo = "default@macrotoys.com"
End If

' Send Mail
With objSendMail
    .From = Request.Form("SenderEmail")
    .To = strTo
    .Subject = Request.Form("Subject") & ",; SenderName:" & Request.Form("SenderName")
    .Importance = Request.Form("Importance")
    .Body = Request.Form("Comment")
    .Send
End With
Set objSendMail = Nothing

' Redirection to browser.asp
Response.Clear
Response.Redirect "Message.asp?Msg=Your mail was send.<BR>Thank you." & 

"&redirect=browser.asp"
```
SendPassword.asp

<%@ Language=VBScript %>
<!—#INCLUDE FILE="DataConn.asp" —>
<%*
***************
* SendPassword.asp
* Program by Sumuscha Teesri
* Send password and user name to customer
* April 25, 2000
***************>

Dim strEmail
strEmail=Request("Email")
Response.Buffer = True

' Connect to customers table to check whether the email provided is in the database
Dim rsUsers
Set rsUsers = Server.CreateObject("ADODB.Recordset")
strSQL = 'SELECT * FROM Customers WHERE Email = ' & strEmail & 
"';
rsUsers.Open strSQL, objConn

If rsUsers.EOF Then
  ' Wrong email address try again
  Response.Clear
  Response.Redirect "ForgetPassword.asp?WrongEmail=True"
Else

  ' Correct email address
  ' Resend information
  Dim objSendMail
  Dim strMessage, strCRLF

  ' Create necessary message
  strCRLF = chr(10) & chr(13)
  strMessage = strCRLF & "This mail was sent to you by Macrotoys.com" & 
  strCRLF & "Your Login name is: " & rsUsers("LogInName") & 
  strCRLF & "Your password is: " & rsUsers("Password")

  ' Create New mail
  Set objSendMail = CreateObject("CDONTS.NewMail")
  With objSendMail
    .From = "webmaster@macrotoys.com"
    .To = rsUsers("Email")
    .Subject = "Your Macrotoys Password"
    .Body = strMessage
    .Send
  End With
  Set objSendMail = Nothing

  ' Back to browser page

209
Shipping.asp

<%@ Language=VBScript %>
<%#******************************************************************************
* CheckOut.asp
* Program by Sumuscha Teesri
* May 9, 2000
******************************************************************************%>
<% Response.Buffer = True'
' Set CheckOut status
If Session("CheckOut") = "" Then
   Session("CheckOut") = "True"
End If
'
' Check wheather customer is loging in
If Session("CusID") = "" Then
   'Customer dont log in redirect to login page
   Response.Clear
   Response.Redirect "Message.asp?Msg=Please log in or register with our system" & _
   "<BR>in order to buy our products." & _
   "&redirect=login.asp"
End If
'
' Customer already login
%>

<!--Include file needed -->
<-- #INCLUDE FILE="Top.asp" -->
<-- #INCLUDE FILE="BrowseBar.asp" -->

<--******************************->
  Begin main  *-->
<--******************************->

<CENTER><H1><FONT COLOR="#22FF22">Check Out</FONT></H1></CENTER>
<%  'Declaration and set necessary variable
  If Len(Request("PlaceOrder")) Then
     If Session("Carrier") <> "" Then
        Response.Clear
        Response.Redirect "Payment.asp"
     Else
        Response.Write "<FONT COLOR=#FF0000><B>Please Select Shipping method " & _

210
before move to the next page.

End If

End If

Dim intCartSize, curPrice, curTotalPrice, curGrandTotal
Dim curTax, curShippingCost, MaxWeight, intBoxes
Dim strFilter, strFilter2, strMessage, strCarrier

MaxWeight = 0
Dim StateChargeRate
StateChargeRate = 0
curTax = 0'

' Find the state shipping zone charge rate
Set objRS=Server.CreateObject("ADODB.Recordset")
strSQL = "SELECT Region, ChargeRate " & _
"FROM Customers, Products, states, statezones " & _
"WHERE region=states.id " & _
"and States.statezoneid=statezones.statezoneid " & _
"and Region='" & Session("Region") & ";"
objRS.Open strSQL, objConn, adOpenStatic, adLockReadOnly, adCmdText

If found the rate apply it
If Not objRS.EOF Then
    StateChargeRate = objRS("ChargeRate")
End If

objRS.Close
Set objRS = Nothing

' Find the carrier
Dim objRSCarr
Set objRSCarr=Server.CreateObject("ADODB.Recordset")
strSQL = "SELECT C.CarrierID, Description, MinWeight, MaxWeight, ShippingCost " & _
"FROM Carriers c, Weightzones w, shippingrates s " & _
"WHERE c.carrierID=s.carrierID and
w.weightzoneid=s.weightzoneid;"
objRSCarr.Open strSQL, objConn, adOpenStatic, adLockReadOnly, adCmdText

If Request("Carrier") = "" Then
    If Session("Carrier") = "" Then
        Session("Carrier") = objRSCarr("Description")
    End If
Else
    Session("Carrier") = Request("Carrier")
End If

' Connect to the product table
Set objRS=Server.CreateObject("ADODB.Recordset")
strSQL = "SELECT ProdID, ProdName, SalePrice, Measure, P.SizeZoneID, " & _
"Weight, ChargeRate " & _
"FROM Products P, SizeZones S " & _
"WHERE P.SizeZoneID=S.SizeZoneID;"
objRS.Open strSQL, objConn, adOpenStatic, adLockReadOnly, adCmdText

' Write table head to client's browser
strMessage = "<TABLE BORDER=1 WIDTH=600><TR>
" & _
"<TD ALIGN=LEFT WIDTH=50><B>ID</B></TD>" & _
"<TD ALIGN=LEFT WIDTH=250><B>Product Name</B></TD>"
& _
"<TD ALIGN=LEFT WIDTH=80><B>Quantity</B></TD>" & _
"<TD ALIGN=LEFT WIDTH=80><B>Packages</B></TD>" & _
"<TD ALIGN=LEFT WIDTH=80><B>Price</B></TD>" & _
"</TR>"

' Show product detail
CartProd = Session("CartProd")
CartQuan = Session("CartQuan")
intCartSize = UBound(CartProd)
For i=0 to intCartSize
  ' Select only the products that are in the shopping cart
  strFilter = "ProdID Like " & CartProd(i)
  objRS.Filter = strFilter
  If Not objRS.EOF Then
    ' Calculate number of packages per item
    intBoxes = (CartQuan(i)/objRS("Measure"))
    ' Calculate Total Price
    curPrice = CCur(CartQuan(i) * objRS("SalePrice"))
    curTotalPrice = CCur(curTotalPrice + curPrice)
    ' Find the maximum weight used for shipping
    If objRS("Weight") > MaxWeight Then
      MaxWeight = objRS("Weight")
    End If
    ' If customer change carrier
    If Session("Carrier")<>"" Then
      strFilter2 = "Description = " & Session("Carrier") & 
      ' and MinWeight <= " & Round(objRS("Weight"))
      & _
      ' and MaxWeight >= " & Round(objRS("Weight"))
      ' Select only carrier need
      objRSCarr.Filter = strFilter2
      ' For debug only
      Response.Write strFilter2 & "<BR>"
      Response.Write "ChargeRate : " & objRSCarr("ChargeRate") & "
      ShippingCost : " & _
      objRSCarr("ShippingCost") & "
      StateChargeRate : " & _
      StateChargeRate & " Boxes : " & 
      intBoxes & "<BR>"
      '******************************************************************************
      ' Calculate shipping cost
      '******************************************************************************

212
curShippingCost = CcUr(curShippingCost + ((1+StateChargeRate) * (1+objRS("ChargeRate"))) * intBoxes)

End If

' ** for debug ** Response.Write ShippingCost

' Write table product information to table
strMessage = strMessage & "<TR><TD ALIGN=LEFT WIDTH=50>
objRS("ProdID") & "<TD ALIGN=RIGHT WIDTH=250>
WIDTH=80>" & _
objRS("ProdName") & "<TD ALIGN=RIGHT WIDTH=80>
WIDTH=80>" & _
CartQuan(i) & "<TD ALIGN=RIGHT WIDTH=80>
WIDTH=80>" & _
intBoxes & "<TD ALIGN=RIGHT WIDTH=80>
FormatCurrency(curPrice) & "</TD></TR>"

End If

' Check whether sell tax apply
If Session("Region")="CA" Then
curTax = FormatCurrency(curTotalPrice * 0.075)
End If

' Calculate Grand total
curGrandTotal = FormatCurrency(curTotalPrice + curTax + curShippingCost)

' Write summary detail to browser
strMessage = strMessage & "<TR><TD ALIGN=RIGHT WIDTH=300
ROWSPAN=4>&nbsp;</TD>
<TD ALIGN=RIGHT><B>Total Price</B></TD>
<TD ALIGN=RIGHT><B>Tax</B></TD>
<TD ALIGN=RIGHT><B>Shipping Cost</B></TD>
<TD ALIGN=RIGHT><B>Grand Total</B></TD>
</TR><TABLE BORDER=0 WIDTH=600><TR>" & _
"<TD WIDTH=300 ALIGN=RIGHT><B>Total Price</B></TD>
"<TD ALIGN=RIGHT><B>Tax</B></TD>
"<TD ALIGN=RIGHT><B>Shipping Cost</B></TD>
"<TD ALIGN=RIGHT><B>Grand Total</B></TD>"</TR></TABLE>"
'*' for debug Response.Write curShippingCost

If Session("Carrier") = "" Then
    Response.Write "<FONT COLOR=#22FF22><B>Please Select Shipping method</B></FONT>"
End If
strMessage = strMessage & "<BR><FORM ACTION=Shipping.asp METHOD=POST>" & 
    "<SELECT NAME=Carrier>"
' Connect to carriers table
strFilter2 = "MaxWeight >= " & MaxWeight & _
    " AND MinWeight <= " & MaxWeight
' Select only the carrier need
objRSCarr.Filter = strFilter2

While Not objRSCarr.EOF
    ' write carrier option to form
    If Session("Carrier")= objRSCarr("Description") Then
        strMessage = strMessage & "<OPTION SELECTED>" & objRSCarr("Description") & _
            "</OPTION>"
    Else
        strMessage = strMessage & "<OPTION>" & objRSCarr("Description") & _
            "</OPTION>"
    End If
    objRSCarr.MoveNext
Wend

objRS.Close
Set objRS = Nothing

objRSCarr.Close
Set objRSCarr = Nothing

Session("TotalPrice") = curTotalPrice
Session("ShippingCost") = curShippingCost
Session("Tax") = curTax
Session("GrandTotal") = curGrandTotal

strMessage = strMessage & "</SELECT><BR>" & _
    "<INPUT TYPE=Submit NAME=ReCalculate Value="Re
    Calculate"">&nbsp;" & _
        "<INPUT TYPE=Submit NAME=PlaceOrder Value="Place
    Order">"</FORM>"

Response.Write strMessage

<!---- INCLUDE FILE="Bottom.asp" -->
<!*********************************************************************>
**ShippingCart.asp**

```vbscript
<%@ Language=VBScript %>
<%!**********************************
* ShoppingCart.asp
* Program by Sumuscha Teesri
* May 7, 2000
**********************************%>
<% Response.Buffer = True
    If Session("Shopping")="" Then
        Response.Redirect "Browser.asp"
    End If
%>
<%!--
<%- #INCLUDE FILE="Top.asp" -->
<%- #INCLUDE FILE="BrowseBar.asp" -->
<%- --------------------------------------------------%>
<%- * Begin main *-->
<%- --------------------------------------------------%>

<CENTER><H1><FONT COLOR="#22FF22">Shopping Cart</FONT></H1></CENTER>
<% ' Declaration and set necessary variable
Dim intCartSize

' Connect to the product table
Set objRS=Server.CreateObject("ADODB.Recordset")
strSQL = "SELECT ProdID, ProdName, SalePrice, Measure " & _
    "FROM Products;"
objRS.Open strSQL, objConn, adOpenStatic, adLockReadOnly, adCmdText
Dim strFilter, strMessage

' Write form and table head to client's browser
strMessage = "<FORM ACTION="ModifyItem.asp" METHOD= POST>
    "<TABLE BORDER=1 WIDTH=600><TR>"
    "<TD ALIGN=CENTER WIDTH=50><B>ID</B></TD>"
    "<TD ALIGN=CENTER WIDTH=350><B>Product Name</B></TD>"
    "<TD ALIGN=CENTER WIDTH=100><B>Quantity</B></TD>"
    "<TD ALIGN=CENTER WIDTH=100><B>Remove</B></TD></TR>"

' Show product detail
CartProd = Session("CartProd")
CartQuan = Session("CartQuan")
intCartSize = UBound(CartProd)
For i=0 to intCartSize
    ' Select only the products that are in the shopping cart
    strFilter = "ProdID Like " & CartProd(i)
    objRS.Filter = strFilter
    If Not objRS.EOF Then
        ' Write product detail
        strMessage = strMessage & "<TR>"
        strMessage = strMessage & "<TD ALIGN=CENTER WIDTH=50>"
```
For \( j = 1 \) to 10

' Limit quantity of the product to the amount that fit in box size

\[
\text{If } (\text{objRS("Measure")}*j) = \text{Cint(CartQuan(i))} \text{ Then}
\]

\[
\text{strMessage = strMessage } \& \text{"<OPTION SELECTED>"}
\]

\[
\text{Else}
\]

\[
\text{strMessage = strMessage } \& \text{"<OPTION>"}
\]

End If

\[
\text{strMessage = strMessage } \& \text{(objRS("Measure") } * \text{ j) } \&
\]

"</OPTION>"

Next

' Provide delete option

strMessage = strMessage & "</TD><TD ALIGN=Center WIDTH=100>"

& _

"<INPUT TYPE=CHECKBOX NAME=DeleteItem VALUE=" & _

i & "</TD></TR>"

End If

Next

objRS.Close

Set objRS = Nothing

strMessage = strMessage & "</TABLE>"

Response.Write strMessage

%>

<TABLE BORDER=0 WIDTH=600><TR><TD>

<INPUT TYPE="SUBMIT" NAME="UpdateItem" VALUE="Update Cart">

<INPUT TYPE="SUBMIT" NAME="Continue" VALUE="Continue Shopping">

<INPUT TYPE="SUBMIT" NAME="Checkout" VALUE="Check Out">

</FORM>

<!--- #INCLUDE FILE="Bottom.asp" -->

<!**************************************************************************>

Template.asp

<%@ Language=VBScript %>>

<**************************************************************************>

* ProgramName.asp
* Program by Sumuscha Teesri
* May 4, 2000
**************************************************************************>

<! Include file needed >

<!--- #INCLUDE FILE="Top.asp" -->

<!--- #INCLUDE FILE="BrowseBar.asp" -->

<!---**************************************************************************-->

<!---* Begin main *-->

<!---**************************************************************************-->

216
<CENTER><H1><FONT COLOR="#22FF22">Title Here</FONT></H1></CENTER>
<!--This is template file please delete and enter code here-->

<!-- #INCLUDE FILE="Bottom.asp" -->
<!*****************************************************************************>
* Top.asp
* Program by Sumuscha Teesri
* Top Template
* May 4, 2000
*****************************************************************************>
<!--*****************************************************************************-->
<!--*****************************************************************************-->
<BASEFONT FACE="Century Gothic">
<HTML>
<HEAD>
<!*****************************************************************************>
* Define meta data **>
<META NAME="Programmer" CONTENT="Sumuscha Teesri">
<META NAME="Designer" CONTENT="Sumuscha Teesri">
<META NAME="University" CONTENT="California State University, San Bernardino">
<META NAME="Department" CONTENT="Business Administration">
<META NAME="Degree" CONTENT="Master of Business Administration">
<META NAME="Concentration" CONTENT="Information Management">
<META NAME="Subject" CONTENT="ADMN 998 Comprehensive Project">
<META NAME="Committee Chair" CONTENT="Prof. Frank M. Lin, Ph.D."/>
<META NAME="Second Faculty Reader" CONTENT="Prof. Harold Dyck, Ph.D."/>
<META NAME="Department Chair" CONTENT="Prof. Walter T. Stewart, Ph.D.">
<TITLE>Macrotoys - Comprehensive Project</TITLE>

<!*****************************************************************************>
* Animation button **>
<script language="JavaScript" fptype="dynamicanimation">
|--
function dynAnimation() {}
function clickSwapImg() {}
//-->
</script>
<!--*****************************************************************************-->
<SCRIPT LANGUAGE="JavaScript">
function fnClock()
{
    var oDate=new Date();
    var sSwitch="AM";
    var iHours=oDate.getHours();
    if(iHours>=12)
if(iHours>12)
{
  iHours-=12;
}

var sMinutes=oDate.getMinutes() + ";
if(sMinutes.length==1)
{
  sMinutes=*0* + sMinutes;
}
var sSeconds=oDate.getSeconds() + ";
if(sSeconds.length==1)
{
  sSeconds=*0* + sSeconds;
}
oTimer.innerHTML=iHours + ":" + sMinutes + ":" + sSeconds + " " + sSwitch;
}]
</SCRIPT>

<!** Show Date, Time and Statistic**>
<TD ROWSPAN=2 VALIGN=Top WIDTH=150><FONT COLOR="#22FF22">Show current Month, Day, and Year</FONT>

<STRONG>
%= Response.Write (MonthName(Month(DATE)))&" & (Day(Date)) & ", " & (Year(Date))%=</STRONG></TD>

<!** Show Time using JavaScript **>
<B><FONT COLOR="#DDFF22" ID=oTimer >%= Time()%</FONT></B>

<SCRIPT LANGUAGE="JavaScript">
var oInterval="";
while (oInterval=""){oInterval=window.setInterval("fnClock()",1000);}
</SCRIPT>

<!** Show Statistic **>
<TD><FONT COLOR="#22FF22">Total Visitors</FONT>

<FONT COLOR="#DDFF22">%= Application("Visits") + 9870%</FONT>

<BR><FONT COLOR="#22FF22">Active Users</FONT>
<FONT COLOR="#DDFF22"><% Application("Active") + 1340%></FONT>

<!-- Show Banner header -->
<TD VALIGN="TOP" WIDTH=500 COLSPAN=5><IMG BORDER="0" SRC="image/banner.gif" WIDTH="500" HEIGHT="60"></TD>

<!-- Tool link bar -->
<TD VALIGN="TOP" ROWSPAN=2 ALIGN="RIGHT" WIDTH=150><%
Dim strToolBox, blnBrowserPage
blnBrowserPage = False
If Session("Shopping") = "True" Then
  If shopping cart has some item show shopping cart
  strToolBox = "<BR><A HREF="ShoppingCart.asp" onmouseover="window.status='Shopping Cart';return true;" onmouseout="window.status='';return true;">Shopping cart</A>
End if

' Log in user show account information
If Session("CusID") <> "" Then
  strToolBox = strToolBox & "<BR>" & _
  "<A HREF="Register.asp?Edit=True" onmouseover="window.status='Your Account';return true;" onmouseout="window.status='';return true;">Your Account</A>"
& _
  "<BR><A HREF="CheckOrder.asp" onmouseover="" onmouseout="" >Check Order</A>"
Else
  ' Use not login shoe Login and new user register
  strToolBox = strToolBox & "<BR><A HREF="Login.asp" onmouseover="window.status='Member LogIn';return true;" onmouseout="window.status='';return true;">Member LogIn</A>"
& _
  "<BR><A HREF="Register.asp" onmouseover="" onmouseout="" >New User</A>"
End If
Response.Write strToolBox
%

<!-- Show animation bottom -->
</TD>

<!-- Home -->
<TD WIDTH="100" HEIGHT="30"><a href="default.asp"
<TD WIDTH="100" HEIGHT="30" ID="fpAnimswapImgFPl" NAME="fpAnimswapImgFPl" DYNAMICANIMATION="fpAnimswapImgFPl" LOWSRC="image/HomeOV.gif"><A><IMG BORDER="0" SRC="image/Home.gif" WIDTH="100" HEIGHT="30" ID="fpAnimswapImgFPl" NAME="fpAnimswapImgFPl" DYNAMICANIMATION="fpAnimswapImgFPl" LOWSRC="image/HomeOV.gif"> </A></TD>

<!** Search **>
<TD WIDTH="100" HEIGHT="30"><A HREF="search.asp"

onmouseover="document["fpAnimswapImgFP2"].imgRolln=document["fpAnimswapImgFP2"].src;

document["fpAnimswapImgFP2"].src=document["fpAnimswapImgFP2"].lowsrc;

window.status='Search';return true;"

onmouseout="document["fpAnimswapImgFP2"].imgRolln=document["fpAnimswapImgFP2"].src;

document["fpAnimswapImgFP2"].src=document["fpAnimswapImgFP2"].lowsrc;

window.status='';return true"> <IMG BORDER="0" SRC="image/Search.gif" WIDTH="100" HEIGHT="30" ID="fpAnimswapImgFP2"

NAME="fpAnimswapImgFP2" DYNAMICANIMATION="fpAnimswapImgFP2" LOWSRC="image/SearchOV.gif"> </A></TD>

<!** FAQ**>
<TD WIDTH="100" HEIGHT="30"><A HREF="faq.asp"

onmouseover="document["fpAnimswapImgFP6"].imgRolln=document["fpAnimswapImgFP6"].src;

document["fpAnimswapImgFP6"].src=document["fpAnimswapImgFP6"].lowsrc;

window.status='FAQ';return true;"

onmouseout="document["fpAnimswapImgFP6"].imgRolln=document["fpAnimswapImgFP6"].src;

window.status='';return true"> <IMG BORDER="0" SRC="image/FAQ.gif" WIDTH="100" HEIGHT="30" ID="fpAnimswapImgFP6"

NAME="fpAnimswapImgFP6" DYNAMICANIMATION="fpAnimswapImgFP6" LOWSRC="image/FAQOV.gif"> </A></TD>

<!** Comment **>
<TD WIDTH="100" HEIGHT="30"><a href="comment.asp?Subject=Comment"
    onmouseover="document['fpAnimswapImgFP5'].imgRolln=document['fpAnimswapImgFP5'].src;
    document['fpAnimswapImgFP5'].src=document['fpAnimswapImgFP5'].lowsrc;
    window.status='Comment';return true;"
    onmouseout="document['fpAnimswapImgFP5'].src=document['fpAnimswapImgFP5'].imgRolln;
    window.status='';return true">" onmouseout="document['fpAnimswapImgFP5'].src=document['fpAnimswapImgFP5'].imgRolln;
    window.status='';return true"</a></td>

<!** About Us **>

<TD WIDTH="100" HEIGHT="30"><a href="about.asp"
    onmouseover="document['fpAnimswapImgFP7'].imgRolln=document['fpAnimswapImgFP7'].src;
    document['fpAnimswapImgFP7'].src=document['fpAnimswapImgFP7'].lowsrc;
    window.status='About Us';return true;"
    onmouseout="document['fpAnimswapImgFP7'].src=document['fpAnimswapImgFP7'].imgRolln;
    window.status='';return true">" onmouseout="document['fpAnimswapImgFP7'].src=document['fpAnimswapImgFP7'].imgRolln;
    window.status='';return true"</a></td>

</TR></TABLE><HR>221
Admin Website Programming Code

AddCat.asp

```vbscript
<%@ Language=VBScript %>
<%Response.Buffer = True%>
<-- #INCLUDE FILE="DataConn.asp" -->
<%
'******************************************************************************
' * AddCat.asp
' * Program by Sumuscha Teesri
' * May 20, 2000
'******************************************************************************

' Connect to the database
Dim objRS
Set objRS = Server.CreateObject("ADODB.Recordset")
If Request("Update") = "True" Then
    strSQL = "SELECT * FROM Categories WHERE CatID=" & CInt(Request("CatID"))
    objRS.Open strSQL, objConn, adOpenForwardOnly, adLockOptimistic, adCmdText
Else
    objRS.Open "Categories", objConn, adOpenForwardOnly,
    adLockOptimistic, adCmdTable
    ' Add new record
    objRS.AddNew
End if
    objRS("CatName") = Request("CatName")
    objRS("Description") = Request("Description")
objRS.Update
' Close the database
objRS.Close
Set objRS = Nothing

' Back to the category list
Response.Clear
Response.Redirect "Category.asp"
%>
```

AddProd.asp

```vbscript
<%@ Language=VBScript %>
<%Response.Buffer = True
'******************************************************************************
' * AddProd.asp
' * Program by Sumuscha Teesri
' * May 20, 2000
'******************************************************************************>

%>
<-- #INCLUDE FILE="Head.asp" -->
<-- #INCLUDE FILE="NavigatorBar.asp" -->
<SCRIPT Language="JavaScript">
```
function VerifyData()
{
    // Verify all necessary fields
    var intPurchasePrice, intSalePrice, intStockQuantity;
    var intReorderPoint, intMeasure, intWeight, strMessage;
    strMessage = ""

    // Check numeric field value
    intPurchasePrice = new Number(document.frmProduct.PurchasePrice.value);
    intSalePrice = new Number(document.frmProduct.SalePrice.value);
    intStockQuantity = new Number(document.frmProduct.StockQuantity.value);
    intReorderPoint = new Number(document.frmProduct.ReorderPoint.value);
    intMeasure = new Number(document.frmProduct.Measure.value);
    intWeight = new Number(document.frmProduct.Weight.value);
    if (isNaN(intPurchasePrice)) strMessage = "Purchase Price; "
    if (isNaN(intSalePrice)) strMessage = strMessage + "Sale Price; ";
    if (isNaN(intStockQuantity)) strMessage = strMessage + "Quantity; "
    if (isNaN(intReorderPoint)) strMessage = strMessage + "Reorder Point; ";
    if (isNaN(intMeasure)) strMessage = strMessage + "Measure; ";
    if (isNaN(intWeight)) strMessage = strMessage + "Weight; ";

    // For debug only
    /*
    alert("PurchasePrice: " + intPurchasePrice + " SalePrice: " +
    intSalePrice + " StockQuantity: " + intStockQuantity + " ReorderPoint: " +
    intReorderPoint + " Measure: " + intMeasure + " Weight: " +
    intWeight);
    */

    if (strMessage !=="")
    {
        // Some numeric fields are not the number.
        strMessage = "The following filed(s) is not a correct numeric type\n\n" +
        strMessage + "\n\nPlease provide new information." ;
        alert(strMessage);
        return false;
    }
    else
    {
        // Check for Not Null field
        strMessage = "";
        if (document.frmProduct.ProdName.value =="") strMessage = "Product Name; ";
if (document.frmProduct.Description.value=="") strMessage = strMessage + "Description; ";
if (document.frmProduct.PurchasePrice.value=="") strMessage = strMessage + "Purchase Price; ";
if (document.frmProduct.SalePrice.value=="") strMessage = strMessage + "Sale Price; ";
if (document.frmProduct.StockQuantity.value=="") strMessage = strMessage + "Quantity; ";
if (document.frmProduct.ReorderPoint.value=="") strMessage = strMessage + "Reorder Point; ";
if (document.frmProduct.Measure.value=="") strMessage = strMessage + "Measure; ";
if (document.frmProduct.Weight.value=="") strMessage = strMessage + "Weight; ";

if (strMessage !="") {
    // Some require fields are null
    strMessage = "The following field(s) is null

    Please provide new information" ;
    alert(strMessage);
    return false;
} else {
    // No error at all
    return true;
}

</SCRIPT>
<TR><TD WIDTH=150><B>Supplier ID:</B></TD>
<INPUT TYPE=HIDDEN NAME=SupID VALUE=%Request("SupID")%>
</TR>
<TR VALIGN=TOP WIDTH=150><B>Product Name:</B></TR>
<INPUT TYPE=TEXT NAME=ProdName SIZE=50 ROWS=4></TR>
<TR VALIGN=TOP WIDTH=150><B>Description:</B></TR>
<TextArea NAME="Description" COLS=50 ROWS=4></TextArea></TR>
<TR VALIGN=TOP WIDTH=150><B>Price:</B></TR>
<INPUT TYPE=TEXT NAME=PurchasePricex/TD>
<INPUT TYPE=TEXT NAME=SalePricex/TD></TR>
<TR VALIGN=TOP WIDTH=150><B>Quantity:</B></TR>
<INPUT TYPE=TEXT NAME=StockQuantityx/TD>
<INPUT TYPE=TEXT NAME=ReorderPointx/TD></TR>
<TR VALIGN=TOP WIDTH=150><B>Unit:</B></TR>
<INPUT TYPE=TEXT NAME=Purchasex/TD>
<INPUT TYPE=TEXT NAME=SaleUnitx/TD></TR>
<TR VALIGN=TOP WIDTH=150><B>Type:</B></TR>
<INPUT TYPE=RADIO NAME=Gender VALUE="M" CHECKED>Male<br>
<INPUT TYPE=RADIO NAME=Gender VALUE="F">Female<br>
<INPUT TYPE=RADIO NAME=Gender VALUE="U">Unisex</TR>
<TR VALIGN=TOP WIDTH=150><B>Dimension:</B></TR>
<INPUT TYPE=TEXT NAME=Measurex/TD>
<INPUT TYPE=TEXT NAME=Weight (lb.)x/TD>
<INPUT TYPE=RADIO NAME=SizeZoneID VALUE=1 CHECKED>12 x 12 x 16<br>
<INPUT TYPE=RADIO NAME=SizeZoneID VALUE=2>16 x 18 x 18<br>
<INPUT TYPE=RADIO NAME=SizeZoneID VALUE=3>24 x 18 x 18<br>
<INPUT TYPE=RADIO NAME=SizeZoneID VALUE=4>48 x 24 x 24<br>
</TR>
<TR WIDTH=150><B>Picture:</B></TR>
<INPUT TYPE=TEXT NAME=Measurex/TD>
</TR>
</TABLE>
AddProdCat.asp

 <%@ Language=VBScript %>
 <%Response.Buffer=True%>
 <!--INCLUDE FILE=DataConn.asp-->
 <!--
  Program by Sumuscha Teesri
  May 20, 2000
  -->

 If Len(Request("btnBack")) Then
   Response.Clear
   Response.Redirect "SubCat.asp?CatID=" & CInt(Request("CatID"))
 End If
 Dim intSubCatID
 intSubCatID = CInt(Request.Form("SubCatID"))

 ' For debug only
 'Response.Write intSubCatID & "<BR>"

 ' Delete old records from ProductCats
 Set objRS=Server.CreateObject("ADODB.Recordset")
 objRS.Open "ProductCats", objConn, adOpenStatic, adLockOptimistic, adCmdTable
 objRS.Filter = "SubCatID=" & intSubCatID
 While Not objRS.EOF
   objRS.Delete.
   objRS.MoveNext
 Wend

 ' Add New Records to ProductCats
 For intCount=1 To Request.Form("ProdList").Count
   Response.Write CInt(Request.Form("ProdList") (intCount)) & "<BR>"
   objRS.AddNew
   objRS("ProdID") = CInt(Request.Form("ProdList") (intCount))
objRS("SubCatID") = intSubCatID
objRS.Update
Next
objRS.Close
Set objRS = Nothing
Response.Clear
Response.Redirect "SubCatDetail.asp?SubCatID=" & intSubCatID
%

AddSubCat.asp

<%@ Language=VBScript %>
<%Response.Buffer=True%>
<!— #INCLUDE FILE="DataConn.asp" -->
<%
'******************************************************************************
' AddSubCat.asp
' Program by Sumuscha Teesri
' May 20, 2000
******************************************************************************
If Len(Request("btnBack")) Then
    Response.Clear
    Response.Redirect "Category.asp"
End If

' Connect to the database
Dim objRS
Set objRS = Server.CreateObject("ADODB.Recordset")
If Request("Update") = "True" Then
    strSQL = "SELECT * FROM SubCategories WHERE SubCatID=" & CInt(Request("SubCatID"))
    objRS.Open strSQL, objConn, adOpenForwardOnly, adLockOptimistic, adCmdText
Else
    objRS.Open "Subcategories", objConn, adOpenForwardOnly, adLockOptimistic, adCmdTable
    ' Add new record
    objRS.AddNew
End If

    objRS("CatID") = CInt(Request("CatID"))
    objRS("SubcatName") = Request("SubcatName")
    objRS("Description") = Request("Description")
objRS.Update
' Close the database
objRS.Close
Set objRS = Nothing

' Back to the category list
Response.Clear
Response.Redirect "SubCat.asp?CatID=" & CInt(Request("CatID"))
%>
AddSup.asp

<%@ Language=VBScript %>
<%Response.Buffer = True
**************
* AddSup.asp
** Program by Sumuscha Teesri
** May 20, 2000
**************>
%
<! Include file needed >
<!-- #INCLUDE FILE="Head.asp" -->
<!-- #INCLUDE FILE="NavigatorBar.asp" -->
<SCRIPT Language="JavaScript">
function VerifyData()
{
    //Verify each necessary field
    var blnCheck, strMsg;
    blnCheck = true;
    strMsg= "You need to provide the following information\n\n"
    //Verify not Null fields
    if (document.frmAddSup.CompanyName.value=="")
    {
        strMsg= strMsg + " Companyname";
        blnCheck = false;
    }
    if (document.frmAddSup.Address1.value=="")
    {
        strMsg= strMsg + " Address";
        blnCheck = false;
    }
    if (document.frmAddSup.City.value=="")
    {
        strMsg= strMsg + " City";
        blnCheck = false;
    }
    if (document.frmAddSup.Country.value=="")
    {
        strMsg= strMsg + " Country";
        blnCheck = false;
    }
    if (document.frmAddSup.Phone.value=="")
    {
        strMsg= strMsg + " Phone";
        blnCheck = false;
    }
    if (document.frmAddSup.FirstName.value=="")
    {
        strMsg= strMsg + " FirstName";
        blnCheck = false;
    }
    if (document.frmAddSup.LastName.value=="")
    {
        strMsg= strMsg + " LastName";
    }
blnCheck = false;
}
if (document frmAddSup.Email.value=="")
{
    strMsg= strMsg + " Email";
    blnCheck = false;
}
if (blnCheck)
{
    //There is no error at all.
    return true;
}
else
{
    // There are some errors.
    alert(strMsg)
    return false
}

<!** Run script to show error message **>
<SCRIPT LANGUAGE = "JavaScript">
<!--
{
    var strErrMsg, strErr

    if (document frmAddSup.Email.value=="")
    {
        strMsg= strMsg + " Email";
        blnCheck = false;
    }
    if (blnCheck)
    {
        //There is no error at all.
        return true;
    }
    else
    {
        // There are some errors.
        alert(strMsg)
        return false
    }

    if (document frmAddSup.CompanyName.value=="")
    {
        strMsg= strMsg + " Company Name";
        blnCheck = false;
    }
    if (blnCheck)
    {
        //There is no error at all.
        return true;
    }
    else
    {
        // There are some errors.
        alert(strMsg)
        return false
    }

    if (document frmAddSup.Address.value=="")
    {
        strMsg= strMsg + " Address";
        blnCheck = false;
    }
    if (blnCheck)
    {
        //There is no error at all.
        return true;
    }
    else
    {
        // There are some errors.
        alert(strMsg)
        return false
    }

    if (document frmAddSup.City.value=="")
    {
        strMsg= strMsg + " City";
        blnCheck = false;
    }
    if (blnCheck)
    {
        //There is no error at all.
        return true;
    }
    else
    {
        // There are some errors.
        alert(strMsg)
        return false
    }

    if (document frmAddSup.State.value=="")
    {
        strMsg= strMsg + " State";
        blnCheck = false;
    }
    if (blnCheck)
    {
        //There is no error at all.
        return true;
    }
    else
    {
        // There are some errors.
        alert(strMsg)
        return false
    }

    if (document frmAddSup.ZipCode.value=="")
    {
        strMsg= strMsg + " Zip Code";
        blnCheck = false;
    }
    if (blnCheck)
    {
        //There is no error at all.
        return true;
    }
    else
    {
        // There are some errors.
        alert(strMsg)
        return false
    }

    if (document frmAddSup.Country.value=="")
    {
        strMsg= strMsg + " Country";
        blnCheck = false;
    }
    if (blnCheck)
    {
        //There is no error at all.
        return true;
    }
    else
    {
        // There are some errors.
        alert(strMsg)
        return false
    }

    if (document frmAddSup.Phone.value=="")
    {
        strMsg= strMsg + " Phone";
        blnCheck = false;
    }
    if (blnCheck)
    {
        //There is no error at all.
        return true;
    }
    else
    {
        // There are some errors.
        alert(strMsg)
        return false
    }

    if (document frmAddSup.Fax.value=="")
    {
        strMsg= strMsg + " Fax";
        blnCheck = false;
    }
    if (blnCheck)
    {
        //There is no error at all.
        return true;
    }
    else
    {
        // There are some errors.
        alert(strMsg)
        return false
    }

    strErrMsg1 = "The following record(s) is already in the system

    Company name is not unique
    If Request("ComUnique") = "False" Then
        strErrMsg2 = strErrMsg2 + "Company Name, "
    End If

    Email address is not unique
    If Request("EmailUnique") = "False" Then
        strErrMsg2 = strErrMsg2 + "Email, "
    End If

    There are some error
    If Len(strErrMsg2) Then
        Dim strErr
        strErr = "True"
    %>
    </SCRIPT>
Receive message from Asp and print in Java
strErr = "<%= strErr %>";
strErrMsg = "<%= strErrMsg1 & strErrMsg2 %>
" "Please re-enter new record"
if (strErr == "True") {alert(strErrMsg)};

</SCRIPT>
<%End If %>

<FORM METHOD=POST ACTION="RecSup.asp" NAME=frmAddSup onSubmit="return VerifyData()"
</FORM>

<CENTER><H1><FONT COLOR=#22FF22>Add Supplier</FONT></H1></CENTER>

<BR><TABLE BORDER=0 WIDTH=500>
  <TR><TD WIDTH=150><B>* Company Name:</B></TD>
  <TD COLSPAN=2><INPUT TYPE=TEXT NAME=CompanyName VALUE=""
SIZE=50"></TD></TR>
  <TR><TD WIDTH=150 ROWSPAN=6 VALIGN=TOP><B>Address :</B></TD>
    <TD><INPUT TYPE=TEXT NAME=Address1 VALUE=""></TD>
    <TD><INPUT TYPE=TEXT NAME=Address2 VALUE=""></TD></TR>
  <TR><TD><INPUT TYPE=TEXT NAME=City VALUE=""></TD>
    <TD><INPUT TYPE=TEXT NAME=Region VALUE=""></TD></TR>
  <TR><TD><INPUT TYPE=TEXT NAME=Postal Code VALUE=""></TD>
    <TD><INPUT TYPE=TEXT NAME=Country VALUE=""></TD></TR>
  <TR><TD WIDTH=150><B>* Phone:</B></TD>
    <TD><INPUT TYPE=TEXT NAME=Phone></TD>
    <TD><INPUT TYPE=TEXT NAME=Fax></TD></TR>
  <TR><TD WIDTH=150><B>Website:</B></TD>
    <TD COLSPAN=3><INPUT TYPE=TEXT NAME=WebSite SIZE=50></TD></TR>
  <TR><TD WIDTH=150 ROWSPAN=6 VALIGN=TOP><B>Contactor:</B></TD>
    <TD><INPUT TYPE=TEXT NAME=Title></TD>
    <TD><INPUT TYPE=TEXT NAME=Last Name></TD>
    <TD><INPUT TYPE=TEXT NAME=FirstName></TD></TR>
  <TR><TD><INPUT TYPE=TEXT NAME=EmailAddress></TD>
    <TD><INPUT TYPE=TEXT NAME=EmailAddress></TD>
    <TD><INPUT TYPE=TEXT NAME=EmailAddress></TD></TR>
</TABLE>
Adrot.txt

Redirect AdRotatorRedirector.asp
width 88
height 31
border 0
* image\nts_iis.gif
Microsoft Internet Information Services
33
image\iisside.gif
http://www.microsoft.com/backoffice
Microsoft Backoffice
33
image\webandhost.gif
http://www.webandhost.com
Web hosting with Webandhost.com
33

AdRotatorRedirector.asp

<%@ Language=VBScript %>
<%Response.Buffer = True
'******************************************************************************
' AdRotatorRedirector.asp
' Program by Sumuscha Teesri
' Record Ad click stratics
' before redirection
' May 5, 2000
'******************************************************************************

strURL = Request.QueryString("url")

Select Case lcase(strURL)
Case
  Application.Lock
  Application("IISAd") = Application("IISAd") + 1
  Application.UnLock
Case "http://www.microsoft.com/backoffice"
  Application.Lock
  Application("BackOfficeAd") = Application("BackOfficeAd") + 1

231
Application.UnLock
Case "http://www.webandhost.com"
    Application.Lock
    Application("WebAndHostAd") = Application("WebAndHostAd") + 1
End Select
Response.Clear
Response.Redirect strURL
%

Category.asp

@ Language=VBScript %>
<%Response.Buffer = True
'********************************************************************************
' * Category.asp
' * Program by Sumuscha Teesri
' * May 20, 2000
'********************************************************************************
%>
<! Include file needed >
<!-- #INCLUDE FILE="Head.asp" -->
<!-- #INCLUDE FILE="navigatorBar.asp" -->
<SCRIPT Language="JavaScript">
<!--
  function VerifyData()
  {
    if (document.frmAddCat.CatName.value=="")
    {
      alert("Please provide the category's name");
      return false;
    }
    else
    {
      alert("New record was added");
      return true;
    }
  }
-->
</SCRIPT>
<!--********************************************************************************--->
<!--* Begin main  *-->
<!--********************************************************************************--->
<% If Session("EmpID") = "" Then
    Response.Redirect "EmpLogIn.asp"
End If

Dim strMessage
strMessage ="<CENTER><H1><FONT COLOR=#22FF22>Category List</FONT></H1></CENTER>" & _
"<TABLE BORDER=1 WIDTH=600>" & _
"<TR><TD ALIGN=CENTER WIDTH=50><B>ID</B></TD>" & _
"<TD ALIGN=CENTER WIDTH=150><B>Name</B></TD>" & _
<table border="0" width="100%" cellspacing="0" cellpadding="0">
<tr><td align="center" width="300">Description</td></tr>
<tr><td align="center" width="100">Tools</td></tr>
</table>

Set objRS = Server.CreateObject("ADODB.Recordset")
objRS.Open "Categories", objConn, adOpenForwardOnly, adLockReadOnly, adCmdTable

WHILE Not objRS.EOF
    strMessage = strMessage & _
    "<tr width="50"><a href="SubCat.asp?CatID=" & _
        objRS("CatID") & ">" & objRS("CatID") & 
        objRS("CatID") & ">" & objRS("CatName") & 
        objRS("Description") & "</a></td>" & _
    "<tr width="150"><a href="SubCat.asp?CatID=" & _
        objRS("CatID") & ">" & 
        objRS("CatID") & "Modify</a></td>" & _
    "<tr width="300"><a href="CatUpdate.asp?CatID=" & _
        objRS("CatID") & "><b>Description</b></a></td>
    objRS.MoveNext
WEND

objRS.Close
Set objRS = Nothing
strMessage = strMessage & "</TABLE>
Response.Write strMessage
%

<!-- #INCLUDE FILE="Footer.asp" -->

<!**********************************>
233
CatUpdate.asp

<%@ Language=VBScript %>
<%Response.Buffer = True
'****************************************************************************** 
' * CatUpdate.asp
' * Program by Sumucha Teesri
' * May 20, 2000
'******************************************************************************
%
<! Include file needed >
<!include FILE="Head.asp" -->
<!include FILE="NavigatorBar.asp" -->
<SCRIPT Language="JavaScript">
<!--
function VerifyData()
{
    if (document.frmCatUpdate.CatName.value=="")
    {
        alert("Please provide the category's name");
        return false;
    }
    else
    {
        alert("The record was updated");
        return true;
    }
-->
</SCRIPT>
<!--***************************************************-->
<!-- Begin main -->
<!--***************************************************-->
<% If Session("EmpID") = "" Then
    Response.Redirect "EmpLogIn.asp"
End If

Set objRS=Server.CreateObject("ADODB.Recordset")
strSQL = "SELECT * FROM Categories WHERE CatID=" &
CInt(Request("CatID"))
objRS.Open strSQL, objConn, adOpenForwardOnly, adLockReadOnly, objCmdText
%
<!H1><FONT COLOR="#22FF22">Modify Category</H1></FONT>
<!FONThis is template file please delete and enter code here--> <FORM ACTION="AddCat.asp?Update=True" METHOD=POST NAME=frmCatUpdate onSubMit="return VerifyData()">
<INPUT TYPE=HIDDEN NAME=CatID VALUE=<%=Request("CatID")%>>
<INPUT TYPE=TEXT NAME=CatName VALUE="<%=objRS("CatName")%>">
<INPUT TYPE=TEXT NAME=Description SIZE=50 MAXLENGTH=50 VALUE="<%=objRS("Description")%>">
</FORM>
</B>Modify Category</B></FONT>
</TABLE BORDER=0 WIDTH=400></TR></TD><TD>Name:</TD><TD>Description:</TD></TR></TABLE BORDER=0 WIDTH=400></TR></TD><TD>Description:
<%=
</TD>
</TABLE BORDER=0 WIDTH=400></TR></TD><TD>Description:
<%=
</TD>
</TABLE BORDER=0 WIDTH=400></TR></TD>
<TD><INPUT TYPE=SUBMIT NAME=btnAdd VALUE="Modify record"></TR></TABLE></FORM>
<% objRS.Close Set objRS = Nothing %>
<!—#INCLUDE FILE="Footer.asp"—>
<!……………………………>
ChangePassword.asp

<%@ Language=VBScript %>
<!***************************************************************************>
* ProgramName.asp
* Program by Sumuscha Teesri
* May 11, 2000
***************************************************************************>
<! Include file needed >
<!—#INCLUDE FILE="Head.asp"—>
<!—#INCLUDE FILE="NavigatorBar.asp"—>
<SCRIPT Language="JavaScript">
<!—
    function VerifyDataO {
        // Verify that password and confirm password is the same
        if (document.frmPassword.Verify.value !=
            document.frmPassword.NewPassword.value)
        {
            alert("Your passwords do not match\nPlease try again");
            return false;
        }
        else
        {
            return true;
        }
    }
—>
</SCRIPT>
<!—***************************************************************************—>
<!—* Begin main *—>
<!—***************************************************************************—>
<%'
    Make sure that only employee is allowed to access this site
Response.Buffer = True
If Session("EmpID") = "" Then
    Response.Redirect "EmpLogIn.asp"
End If
' Declaration necessary value
Dim blnConfirm, strMessage
blnConfirm = False
strMessage = ""

235
If Len(Request.Form("Submit")) Then
    ' Employee change password
    ' For debug only
    
    ' Select only the employee who currently logging in
    strSQL = "SELECT Password FROM Employees WHERE EmpID=" & Session("EmpID") & ";"
    Set objRS=Server.CreateObject("ADODB.Recordset")
    objRS.Open strSQL, objConn, adOpenForwardOnly, adLockOptimistic, adCmdText
    ' Check old password for security
    If UCase(objRS("Password")) = UCase(Request("OldPassword")) Then
        ' Old password match, update new password
        objRS("Password") = Request("NewPassword")
        objRS.Update
        ' Send message to user show the password has been changed
        strMessage = "<CENTER><H1><FONT COLOR=#22FF22>" & _
        "Password Confirm</FONT></H1></CENTER><BR>" & _
        "<CENTER><B>Your Password has been changed</B></CENTER></B>"
        blnConfirm = True
    Else
        ' Old password does not match
        strMessage = "<FONT COLOR=#FF0000><B>Wrong Password Please try again</B></FONT>"
    
    End If
    ' Close the employees table
    objRS.Close
    Set objRS = Nothing
Else
    ' Employee first visit this page, create the form for input password
    strMessage = "<CENTER><H1><FONT COLOR=#22FF22>" & _
    "Change Password</FONT></H1></CENTER>"
End If
Response.Write strMessage
If Not blnConfirm Then

</FORM ACTION="ChangePassword.asp" NAME="frmPassword"
OnSubmit="return VerifyData()" METHOD="POST">
Please Enter the following information:
<TABLE WIDTH=300>
    <TR><TD>Old Password:</TD><TD></TD></TR>
    <TR><TD><INPUT TYPE=Password NAME="OldPassword"></TD></TR>
    <TR><TD>New Password:</TD><TD></TD></TR>
    <TD>Confirm Password:</TD><TD></TD></TR>
    <TR><TD><INPUT TYPE=Password NAME="NewPassword"></TD></TR>
</TABLE>

236
<TD><INPUT TYPE=Password NAME="Verify"></TD></TR></TABLE>
<INPUT TYPE=SUBMIT NAME="Submit" VALUE="Submit">
<INPUT TYPE=RESET></FORM>

<% End If %>

<%<-- #INCLUDE FILE="Footer.asp" -->
<!********************************************************************>

CheckEmp.asp

<%@ Language=VBScript %>
<!-- #INCLUDE FILE="DataConn.asp" -->
<% Response.Buffer = True

' Declaration statement
Dim strPassword, blnFail

' blnFail if False means correct email and password
' allow employee login
blnFail = True
strPassword = Request.Form("Password")

' Connect to Employees table and select only the employee
' who is logging in
Set objRS = Server.CreateObject("ADODB.Recordset")
strSQL = "SELECT * FROM Employees WHERE Email='" & Request.Form("Email") & "}"
objRS.Open strSQL, objConn, adOpenForwardOnly, adLockReadOnly, adCmdText
Response.Write "Email :" & Request.Form("Email") & " Password : " & _
strPassword & "<BR>"

' Check whether the correct email address
If Not objRS.EOF Then
' Correct email address
Response.Write "Email :" & objRS("Email") & " Password : " & objRS("Password") & "<BR>"

' Check password
If (UCase(objRS("Password"))= UCase(strPassword)) Then
    Session("EmpID") = objRS("EmpID")
    Session("FirstName") = objRS("FirstName")
    Session("LastName") = objRS("LastName")
    Session("Email") = objRS("Email")
' Correct employee login
    blnFail = False
End If
End If

' Close the employees table
objRS.Close
Set objRS = Nothing

' Clear buffer before redirection

237
Response.Clear
If blnFail Then
    ' Login fail try again
    Response.Redirect "EmpLogIn.asp?LoginFail=True"
Else
    ' Correct user go to mainmenu
    Response.Redirect "MainMenu.asp"
End If
%

**CusDetail.asp**

<%@ Language=VBScript %>
<--#INCLUDE FILE="Head.asp" -->
<--#INCLUDE FILE="NavigatorBar.asp" -->

<--**********************************-->
<!--#INCLUDE FILE="NavigatorBar.asp" -->
<--**********************************-->

' Make sure that only employee can access this website
Response.Buffer = True
If Session("EmpID") = "" Then
    Response.Redirect "EmpLogIn.asp"
End If

' Connect to Customer database
Set objRS = Server.CreateObject("ADODB.Recordset")
strSQL = "SELECT * FROM Customers WHERE CusID=" & CInt(Request("CusID")) & "];

' Check wheather chang employee in customer database
If Len(Request("btnChgEmp")) Then
    ' If chang employee update data
    objRS.Open strSQL, objConn, adOpenForwardOnly, adLockOptimistic, adCmdText
    ' For debug only
    'Response.Write Request("EmpID") & "<BR>"
    objRS("EmpID") = CInt(Request("EmpID"))
    objRS.Update
    objRS.Close
End IF

' Connect to Customer database
objRS.Open strSQL, objConn, adOpenForwardOnly, adLockReadOnly, adCmdText

Dim strMessage, strType, objRSType

238
' Find Customer type detail
Set objRSType = Server.CreateObject("ADODB.Recordset")
strSQL = "SELECT Description FROM CustomerTypes WHERE ID='" &
objRSType.Type & ";"
objRSType.Open strSQL, objConn, adOpenForwardOnly, adLockReadOnly,
adCmdText
strType = objRSType("Description")
objRSType.Close
Set objRSType = Nothing

' Write web detail to client's browser
strMessage = "<CENTER><H1><FONT COLOR=#22FF22>Customer
Detail</FONT></H1></CENTER>" & _
"<BR><TABLE BORDER=0 WIDTH=600><TR>" & _
"<TD WIDTH=150><B>Customer ID:</B></TD><TD>" & _
objRS("CusID") & "</TD><TD COLSPAN=2>" & strType & _
"</TD><TR>" & _
"<TD WIDTH=150><B>Company Name:</B></TD><TD>" & _
objRS("CompanyName") & "</TD><TD></TD><TR>" & _
"<TD WIDTH=150 ROWSPAN=2 VALIGN=TOP><B>Address</B>:" & _
objRS("Address1") & "</TD><TD VALIGN=TOP>" & _
objRS("Address2") & "</TD><TD></TD><TR>" & _
"<TR><TD>" & objRS("City") & ""," & objRS("Region") & _
Else
    strMessage = strMessage & "<OPTION"
End If
strMessage = strMessage & " VALUE=" & objRSEmp("EmpID") & ">
 objRSEmp("FirstName") & _
   " OPTION>"
objRSEmp.MoveNext
Wend

' Write closing information to client's browser
strMessage = strMessage & "<TD><TD COLSPAN=2><INPUT TYPE=SUBMIT.
NAME=btnChgEmp " & _
   "VALUE=""Change
Representative""</TD></TR></SELECT></TABLE>"
Response.Write strMessage

' Close database
objRSEmp.Close
Set objRSEmp = Nothing

objRS.Close
Set objRS = Nothing
%

<!-- #INCLUDE FILE="Footer.asp" -->
<!**************************>

Customer.asp

%@ Language=VBScript %>
<!**************************
* Customer.asp
* Program by Sumuscha Teesri
* May 11, 2000
**************************>

<! Include file needed >
<!-- #INCLUDE FILE="Head.asp" -->
<!-- #INCLUDE FILE="NavigatorBar.asp" -->
<!--**************************-->
<!-* Begin main *-->
<!--**************************-->

<%'
' Make sure that only employee login
Response.Buffer = True
If Session("EmpID") = "" Then
    Response.Redirect "EmpLogIn.asp"
End If
' Write head to client's browser
Dim strMessage
strMessage="<CENTER><H1><FONT COLOR=#22FF22>CUSTOMER</FONT></H1></CENTER>"
Set objRS = server.CreateObject("ADODB.Recordset")
' Select only current employee
strSQL = "SELECT CusID, CompanyName, Phone, Fax, Email " & _
"FROM Customers "
If Len(Request("SearchMethod")) Then
' Response.Write Request("SearchMethod")
Select Case Request.Form("SearchMethod")
Case "ID"
strSQL = strSQL & "WHERE CusID Like " & Request("Keyword") & "'" & _
Case "LogInName"
strSQL = strSQL & "WHERE LogInName Like " & Request("Keyword") & "'" & _
Case "Email"
strSQL = strSQL & "WHERE Email Like " & Request("Keyword") & "'" & _
Case "CompanyName"
strSQL = strSQL & "WHERE CompanyName Like " & Request("Keyword") & "'" & _
End Select
Else
strSQL = strSQL & "WHERE EmpID =" & Session("EmpID")
End If
' Connect to database
objRS.Open strSQL, objConn, adOpenStatic, adLockReadOnly
If Not objRS.EOF Then
' Write detail to client's browser
strMessage = strMessage & "<TABLE BORDER=1 WIDTH=625><TR>" & _
"<TD WIDTH=30 ALIGN=LEFT><B>ID</B></TD>" & _
"<TD WIDTH=200 ALIGN=LEFT><B>Name</B></TD>" & _
"<TD WIDTH=150 ALIGN=LEFT><B>Phone</B></TD>" & _
"<TD WIDTH=150 ALIGN=LEFT><B>Fax</B></TD>" & _
"<TD WIDTH=80 ALIGN=LEFT><B>Email</B></TD>"<TR>
While Not objRS.EOF
strMessage = strMessage & 
"<A HREF=CusDetail.asp?CusID=" & 
objRS("CusID") & ">" & _
"<A HREF=CusDetail.asp?CusID=" & 
objRS("CusID") & ">
objRS("CusID") & ">" & _
objRS("CompanyID") & ">" & _
"<TD WIDTH=150><SMALL>" & objRS("Phone")
"<TD WIDTH=150><SMALL>" & objRS("Fax") &
"<TD WIDTH=80><SMALL>" & _
"<A HREF=mailto: & objRS("Email") & "> "$ &
objRS("Email") & "</A></TD></TR>

objRS.MoveNext
Wend
strMessage = strMessage & "</TABLE>"
Else
strMessage = strMessage & "</BR><BR><CENTER>No customer was found." & _
"<BR>Please try again."
End If
objRS.Close
Set objRS = Nothing
Response.Write strMessage
</FORM>

SEARCH CUSTOMER DETAIL
Please select one of the following searching method:<BR>

<table>
<thead>
<tr>
<th>Search Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Customer ID</td>
</tr>
<tr>
<td>Login Name</td>
<td>Login Name</td>
</tr>
<tr>
<td>Email</td>
<td>Email</td>
</tr>
<tr>
<td>Company Name</td>
<td>Company Name</td>
</tr>
</tbody>
</table>

Search Keyword: <BR>

Search using WildCard:<BR>

*WildCard:* Use (underscore) for one character or % for many characters

Example: %Computer% for every company with the word Computer

<!— INCLUDE FILE="Footer.asp" —>

***************
DataConn.asp

<!** Include necessary library**>
<!-- METADATA TYPE="typelib"
FILE="C:\Program Files\Common Files\System\ado\msado15.dll" -->
<%

' *****************************************
' DataConn.asp
' Program by Sumuscha Teesri
' Database connection file
' Modify this file to connect to other
' Database such as Access, Oracle, or MS SQL
' May 4, 2000
' *******************************************

Dim objConn
Set objConn = Server.CreateObject("ADODB.Connection")
Dim strSQL
' Define database connection modify this line as needed
objConn.Open "Provider=SQLOLEDB.1;Persist Security Info=False;User ID=IUSR;" & 
"Initial Catalog=MacrotoysSQL;Data Source=ZEBRASERVER"

If Session("blnValidUser") = True and Session("CusID") = "" Then
  Dim rsUsersCheck, strSQL
  Set rsUsersCheck = Server.CreateObject("ADODB.Recordset")
  strSQL = "SELECT * FROM Customers " & 
  "WHERE CusID = " & Request.Cookies("SaveLogin")("CusID") & ";" 
  rsUsersCheck.Open strSQL, objConn
  Dim strName, strValue
  For Each strField in rsUsersCheck.Fields
    'populate session variable
    strName = strField.Name
    strValue = strField.Value
    Session(strName) = strValue
  Next
  rsUsersCheck.Close
  Set rsUsersCheck = Nothing
End If
%>
Default.asp

<%@ Language=VBScript %>
<%!*****************************************************
* Default.asp
* Program by Sumuscha Teesri
* Default file before forward to other page
* May 4, 2000
*****************************************************%>
<!-- #INCLUDE FILE="DataConn.asp" -->
<%  ' Clear Session variable
Session.Contents.RemoveAll
  ' Record start time
Session("Start") = Now

  ' Keep static to database file
  ' Connect to the database:
Dim objRS, objBrowser
Set objRS = Server.CreateObject("ADODB.Recordset")
objRS.Open "WebStats", objConn, adOpenForwardOnly, adLockOptimistic, adCmdTable
  ' Add new record
objRS.AddNew

  ' Populate data
Set objBrowser = Server.CreateObject("MSWC.BrowserType")
objRS("Browser") = Left(objBrowser.Browser,20)
objRS("Version") = Left(objBrowser.Version,20)
objRS("RemoteAddress") = Left(Request.ServerVariables("REMOTE_ADDR"),30)
objRS("RemoteHost") = Left(Request.ServerVariables("REMOTE_HOST"),50)
objRS("HttpRef") = Left(Request.ServerVariables("HTTP_REFERER"),50)
objRS("SessionID") = Session.SessionID
objRS("ActiveDate") = Now
objRS.Update

  ' Close database
objRS.Close
Set objRS = Nothing
Set objBrowser = Nothing
%>

<HTML>
<BASEFONT FACE="Century Gothic">
<HEAD>
<TITLE>B2Beca -- Project</TITLE>

<!-- #INCLUDE FILE="DataConn.asp" -->
<%! ** Count before redirect to browser.asp**>
<SCRIPT LANGUAGE="JavaScript"><!--
  var start=new Date();
  start=Date.parse(start)/1000;
  var counts=5;

  var recordID = parseInt(Request.ServerVariables("ZONE"));
  if (recordID == 1)
    var count = (start + 5) - start;
  else
    var count = (start + 10) - start;

  objRS("Count") = count;
  objRS.Update;

  var count = (start + 15) - start;
  objRS("Count") = count;
  objRS.Update;

  objRS("ActiveDate") = Now;
  objRS.Update;

  ' Close database
  objRS.Close;
  Set objRS = Nothing;
</script>
function CountDown()
{
    var now=new Date();
    now=Date.parse(now)/1000;
    var x=parseInt(counts-(now-start),10);
    if(document frm){document frm.Count.value = x;}
    if(x>0){
        timerID=setTimeout("CountDown()", 100)
    }
    else{
        location.href="EmpLogIn.asp"
    }
}

/*-->
</SCRIPT>
</HEAD>
<BODY BGCOLOR="#000080" TEXT="#33bbbb" ALINK="#cc3333" LINK="#ddff00" VLINK="#cc33ff"
TOPMARGIN="0" LEFTMARGIN="0">
<!-- Show project information -->
<br>
<CENTER><FONT COLOR="#22ff22"><H3>IMPLEMENTATION BUSINESS-TO-BUSINESS ELECTRONIC COMMERCE</H3></CENTER>
<br>
<CENTER>This program is a part of ADMN998 Comprehensive Project.<BR>
In Partial Fulfillment of the Requirement of the Degree<BR>
Master of Business Administration:<A HREF="http://www.spba.csusb.edu/mba/"
 onMouseOver="window.status='SPBA';return true"
 onMouseOut="window.status='';return true">Information Management</A>
California State University,
San Bernardino<br>
<br>
<TABLE BORDER=0 WIDTH=600>
<tr>
    <td WIDTH=100 ROWSPAN=4>&nbsp;</td>
    <td WIDTH=200><B>Committee Chair:</B></td>
    <td>Professor <A href="mailto:flin@csusb.edu">Frank M. Lin</A>, Ph.D.</td>
</tr>
<tr>
    <td>Professor <A href="mailto:hdyck@csusb.edu">Harold Dyck</A>, Ph.D.</td>
</tr>
<tr>
    <td>Department Chair:</td>
    <td>Professor <A href="mailto:wstewart@csusb.edu">Walter T. Stewart, Jr.</A>, Ph.D.</td>
</tr>
<tr>
    <td>Create and Design By:</td>
    <td>Sumuscha Teesri</td>
</tr>
</TABLE>
<TR>
</TABLE>
<P>
<HR>
<CENTER>
<FORM Name=frm >
<H3>Redirected to Macrotoys in <INPUT NAME="Count" SIZE="2" VALUE="5"> seconds.
<BR>To redirect now click <A href="EmpLogIn.asp"
onmouseover="window.status='Click here to continue'; return true">here</A>. </H3>
</FORM>
</CENTER>

<BLOCKQUOTE><FONT COLOR="#22ff22">
Copyright © 2000 All Rights Reserved</FONT></BLOCKQUOTE>

<SCRIPT language="JavaScript">
<!--
window.setTimeout('CountDown()',10);
-->
</SCRIPT>
</BODY>
</HTML>

**Run countdown**

**EmpLogIn.asp**

<%@ Language=VBScript %>

<--*******************************
* EmpLogIn.asp
* Program by Sumuscha Teesri
* May 11, 2000
*******************************

<--**** Verify that user provide needed information**>
<SCRIPT Language="JavaScript">
<!--
function VerifyData()
{
  if (document.frmLogIn.Email.value == "")
  {
    alert("Please enter your Log in name");
    return false;
  }
  else
  {
    if (document.frmLogIn.Password.value == "")
    {
      alert("Please enter your Password");
      return false;
    }
    else
      return true;
  }
}
</SCRIPT>

246
Warning! only Macrotoys' staffs are allowed to access this site

Unauthorized login is illegal and will be fined for $1,000,000

If Request("LoginFail") <> "" Then
    Response.Write"<b>Login Fail! Please try again</b>"
End If

EmpMessage.asp

EmpMessage.asp

<! Include file needed >
<!-- #INCLUDE FILE="Head.asp" -->
<!-- #INCLUDE FILE="NavigatorBar.asp" -->

<!--
Begin main -->

<HR TYPE="middle" WIDTH="100%">

<HR TYPE="middle" WIDTH="100%">

<HR TYPE="middle" WIDTH="100%">

</HR TYPE="middle" WIDTH="100%">

<!-- #INCLUDE FILE="Footer.asp" -->
<!**************************************************************************>

<form action="EmpMessage.asp" method="post">
    Please enter your Email Address:<br>
    <input type="text" name="Email" size=50><br>
    Please enter your Password:<br>
    <input type="password" name="Password" size=50><br>
    <input type="submit" value="Click here to continue"> 
</form>

<! INCLUDE FILE="Footer.asp" -->
<!**************************************************************************>
**EmpOrderDetail.asp**

<!--
#INCLUDE FILE="Head.asp"
-->

<!--
#INCLUDE FILE="NavigatorBar.asp"
-->  

arehouse Buffer = True
' Make sure that only employee is allowed to login  
If Session("EmpID") = "" Then  
    Response.Redirect "EmpLogIn.asp"
End If  

'**************************************************************  
'* Start Order detail part  
'**************************************************************

' Prepare recordset  
Set objRS = Server.CreateObject("ADODB.Recordset")

' Employee change order status  
If Len(Request.Form("OrderStatus")) Then  
    Dim strFilter  
    ' For debug only  
    'Response.Write Request.Form("OrderStatus") & "<BR>"  
    'Response.Write Request("OrderID") & "<BR>"

    ' Make sure that update the correct record  
    strFilter="OrderID =" & Request("OrderID")  
' Connect to database  
    objRS.Open "Orders", objConn, adOpenForwardOnly, adLockOptimistic,  
    adCmdTable  
    objRS.Filter = strFilter  
    ' Update order status  
    objRS("Status") = Request.Form("OrderStatus")  
    objRS.Update  
    objRS.Close  
End If  

' Select only fields need from the database  
strSQL = "SELECT OrderId, CompanyName, S.Description OrderStatus,  
OrderDate, " & _  

248
"Ca.Description Carrier, Tracking, E.FirstName EmpName, " & _
"E.LastName EmpLastName, P.Description Payment " & _
"FROM Orders O, Customers C, Employees E, " & _
"OrderStatus S, Carriers Ca, PaymentTypes P " & _
"WHERE C.EmpID = E.EmpID and C.CusID = O.CusID " & _
"and S.ID = O.Status and Ca.CarrierID = O.CarrierID and " & _
"OrderByID = " & Cint(Request("OrderID"))
objRS.Open strSQL, objConn, adOpenForwardOnly, adLockReadOnly, adCmdText

' Write head and and table to client’s browser
strMessage = "<CENTER><H1><FONT COLOR=#22FF22>Order Detail</FONT>" & _
"</H1></CENTER><TABLE WIDTH=600><TR>" & _
"<TD VALIGN=TOP WIDTH=70><B>Order ID:</B></TD><TD VALIGN=TOP WIDTH=70><B>Date :</B></TD>
"<TD VALIGN=TOP WIDTH=70><B>Company:</B></TD><TD VALIGN=TOP WIDTH=70><B>Payment :</B></TD>
"<TD VALIGN=TOP WIDTH=70><B>Carrier :</B></TD><TD VALIGN=TOP WIDTH=70><B>Tacking :</B></TD>

' Connect to OrderStatus table
Dim objRSStatus
Set objRSStatus = Server.CreateObject("ADODB.Recordset")
objRSStatus.Open "OrderStatus", objConn, adOpenForwardOnly, adLockReadOnly, adCmdTable

' Write List box to the form in client’s browser
strMessage = strMessage & "<TR><TD><TD COLSPAN=3><FORM
METHOD=POST>" & _
"<INPUT TYPE=HIDDEN NAME=OrderID VALUE="&
Request("OrderID") & _
"SELECT NAME=OrderStatus>"
objRSStatus.MoveNext
Wend
' Close table after used
objRSStatus.Close
Set objRSStatus = Nothing
objRS.Close
Set objRS = Nothing

'***************************************************************
'* Finite Order detail part
'* Begin Product detail part
'***************************************************************
strMessage = strMessage & "</SELECT><INPUT TYPE=SUBMIT NAME=ChangeStatus " & _
"VALUE=""Change Order Status""/></FORM></TD></TR></TABLE>"

Set objRS = Server.CreateObject("ADODB.Recordset")

' Select only the product in this order
strSQL = "SELECT P.ProdID, ProdName, Quantity, SalePrice " & _
"FROM Products P, Transacts T " & _
"WHERE P.ProdID=T.ProdID and OrderID=" & Request("OrderID") & ":"
objRS.Open strSQL, objConn, adOpenForwardOnly, adLockOptimistic, adCmdText

strMessage = strMessage & "<TABLE BORDER=1 WIDTH=600><TR>
" & _
"<TD WIDTH=100 ALIGN=LEFT>Product ID</TD>
" & _
"<TD WIDTH=300 ALIGN=LEFT>Name</TD>
" & _
"<TD WIDTH=100 ALIGN=RIGHT>Quantity</TD>
" & _
"<TD WIDTH=100 ALIGN=RIGHT>Total Price</TD></TR>

' Write down the detail
While Not objRS.EOF
objRS("ProdID") & _
"</TD><TD WIDTH=300 >" & objRS("ProdName")& _
"</TD><TD WIDTH=100 ALIGN=RIGHT>"&

objRS("Quantity") & _
"</TD><TD WIDTH=100 ALIGN=RIGHT>"& _
FormatCurrency(objRS("Quantity")* 

objRS("SalePrice")& _
"</TD></TR>"

objRS.MoveNext
Wend
strMessage = strMessage & "</TABLE><BR>"
Response.Write strMessage

' Close the table after finish
objRS.Close
Set objRS = Nothing

"
EmpProdDetail.asp

<script Language="JavaScript">

function VerifyData()
{
    var intPurchasePrice, intSalePrice, intStockQuantity;
    var intReorderPoint, intMeasure, intWeight, strMessage;
    strMessage = "
    intPurchasePrice = new Number(document.frmProduct.PurchasePrice.value);
    intSalePrice = new Number(document.frmProduct.SalePrice.value);
    intStockQuantity = new Number(document.frmProduct.StockQuantity.value);
    intReorderPoint = new Number(document.frmProduct.ReorderPoint.value);
    intMeasure = new Number(document.frmProduct.Measure.value);
    intWeight = new Number(document.frmProduct.Weight.value);

    //For debug only
    /*
        alert("PurchasePrice: " + intPurchasePrice + " SalePrice: " + intSalePrice + " StockQuantity: " + intStockQuantity + " ReorderPoint: " + intReorderPoint + " Measure: " + intMeasure + " Weight: " + intWeight);
    */
    if (isNaN(intPurchasePrice)) strMessage = "Purchase Price; ";
    if (isNaN(intSalePrice)) strMessage = strMessage + "Sale Price; ";
    if (isNaN(intStockQuantity)) strMessage = strMessage + "Quanatity; ";
    if (isNaN(intReorderPoint)) strMessage = strMessage + "Reorder Point; ";
    if (isNaN(intMeasure)) strMessage = strMessage + "Measure; ";

    return strMessage;
}
</script>

251
if (isNaN(intWeight)) strMessage = strMessage + "Weight; ";
if (strMessage !="")
{
    strMessage = "The following filed(s) is not a correct numeric type\n\n" + strMessage + "\n\nPlease provide new information" ;
    alert(strMessage);
    return false;
}
else
{
    alert("The record was updated")
    return true;
}

-->
</SCRIPT>

<!--**********************************************************************
<!-- Begin main -->
<!--**********************************************************************
<% If Session("EmpID") = "" Then
    Response.Redirect "EmpLogIn.asp"
End If

If Len(Request("btnBack")) Then
    ' Back to Product page
    Response.Clear
    Response.Redirect "Product.asp?SupID=", Cint(Request("SupID"))
End If
strSQL = "SELECT * FROM Products Where ProdID=", Cint(Request("ProdID")) & ";"
Set objRS = Server.CreateObject("ADODB.Recordset")

    ' Check whether update suppliers table is made
If Len(Request("btnUpdate")) Then
    
    ' For debug only
    '**********************************************************************
    'Response.Write Request("ProdName") & "<BR>"
    'Response.Write Request("Description") & "<BR>"
    'Response.Write Request("PurchasePrice") & "<BR>"
    'Response.Write Request("SalePrice") & "<BR>"
    'Response.Write Request("StockQuantity") & "<BR>"
    'Response.Write Request("ReorderPoint") & "<BR>"
    'Response.Write Request("SaleUnit") & "<BR>"
    'Response.Write Request("PurchaseUnit") & "<BR>"
    'Response.Write Request("Age") & "<BR>"
    'Response.Write Request("Gender") & "<BR>"
    'Response.Write Request("Measure") & "<BR>"
    'Response.Write Request("SizeZoneID") & "<BR>"
    'Response.Write Request("Weight") & "<BR>"
    '**********************************************************************

252
Update data

objRS.Open strSQL, objConn, adOpenForwardOnly, adLockOptimistic, adCmdText

    objRS("ProdName") = Request("ProdName")
    objRS("Description") = Trim(Request("Description"))
    objRS("PurchasePrice") = CCur(Request("PurchasePrice"))
    objRS("SalePrice") = CCur(Request("SalePrice"))
    objRS("StockQuantity") = CInt(Request("StockQuantity"))
    objRS("ReorderPoint") = CInt(Request("ReorderPoint"))
    objRS("SaleUnit") = Request("PurchaseUnit")
    objRS("PurchaseUnit") = Request("SaleUnit")
    objRS("Age") = Request("Age")
    objRS("Gender") = Request("Gender")
    objRS("Measure") = CSng(Request("Measure"))
    objRS("SizeZoneID") = Request("SizeZoneID")
    objRS("Weight") = CSng(Request("Weight"))
    objRS.Update

objRS.Close

End IF

objRS.Open strSQL, objConn, adOpenForwardOnly, adLockOptimistic, adCmdText

%>

< CENTER>< H1> < FONT COLOR="#22FF22"> Product Detail </ FONT></ H1> </ CENTER>
< FORM METHOD=POST NAME= frmProduct >
< BR > < TABLE BORDER=0 WIDTH=500 >
< TR > < TD WIDTH=150 > B > Product ID: </ B > < TD >&equals; objRS( "ProdID" ) &equals; </ TD >
< TD > &lt; B &gt; Supplier ID: </ B > &lt; TD >&equals; objRS( "SupID" ) &equals; &lt; / TD &gt; &lt; / TR &gt;
< TR > < TD COLSPAN=2 > &lt; INPUT TYPE=TEXT NAME= ProdName SIZE=50 VALUE= " &equals; objRS( "ProdName" ) &equals; " &gt; &lt;/ TD &gt; &lt;/ TR &gt;
< TR > < TD VALIGN=TOP WIDTH=150 ROWSPAN=1 VALIGN=TOP &gt; Description : &lt;/ TD &gt; &lt; / TR &gt;
< TD COLSPAN=3 > &lt; TEXTAREA NAME= "Description" COLS=50 ROWS=4&equals;Cstr(objRS("Description")) &equals; &gt; &lt;/ TEXTAREA &gt; &lt;/ TD &gt; &lt; / TR &gt;
< TR > < TD VALIGN=TOP WIDTH=150 ROWSPAN=2 VALIGN=TOP &gt; B &gt; Price &lt; / TD &gt;
< TD > &lt; SMALL &gt; Purchase &lt; / SMALL &gt; &lt;/ TD &gt;
< TD > &lt; SMALL &gt; Sale &lt; / SMALL &gt; &lt;/ TD &gt; &lt; / TR &gt;
< TR > < TD > &lt; INPUT TYPE=TEXT NAME= PurchasePrice VALUE= " &equals; objRS( "PurchasePrice" ) &equals; " &gt; &lt;/ TD &gt;
< TD VALIGN=TOP &gt; &lt; INPUT TYPE=TEXT NAME= SalePrice VALUE= " &equals; objRS( "SalePrice" ) &equals; " &gt; &lt;/ TD &gt; &lt;/ TR &gt;
< TR > < TD WIDTH=150 ROWSPAN=2 &gt; &nbsp; &lt;/ TD &gt;
< TD > &lt; SMALL &gt; Quantity &lt; / SMALL &gt; &lt;/ TD &gt;
< TD > &lt; SMALL &gt; Reorder Point &lt; / SMALL &gt; &lt;/ TD &gt; &lt; / TR &gt;
< TR > < TD > &lt; INPUT TYPE=TEXT NAME= StockQuantity VALUE= " &equals; objRS( "StockQuantity" ) &equals; " &gt; &lt;/ TD &gt;
< TD > &lt; INPUT TYPE=TEXT NAME= ReorderPoint VALUE= " &equals; objRS( "ReorderPoint" ) &equals; " &gt; &lt;/ TD &gt; &lt;/ TR &gt;
< TR > < TD VALIGN=TOP WIDTH=150 ROWSPAN=2 &gt; &lt; B &gt; Unit &lt; / B &gt; &lt; / TD &gt; &lt;/ TR &gt;
<table>
<thead>
<tr>
<th>Type</th>
<th>Age</th>
<th>Gender</th>
<th>Measure</th>
<th>Weight (lb)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Purchase</strong></td>
<td><strong>Age</strong></td>
<td><strong>Gender</strong></td>
<td><strong>Measure</strong></td>
<td><strong>Weight (lb)</strong></td>
</tr>
<tr>
<td><strong>Sale</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Dimension**

1. **Gender**
   - **M**
     - If `objRS("Gender") = "M"` then **CHECKED**
     - Response.Write
   - **F**
     - If `objRS("Gender") = "F"` then **CHECKED**
     - Response.Write
   - **U**
     - If `objRS("Gender") = "U"` then **CHECKED**
     - Response.Write
   - **Unisex**
     - If `objRS("Gender")` is not **M**, **F**, or **U** then **CHECKED**

2. **SizeZoneID**
   - **SizeZoneID = 1**
     - If `objRS("SizeZoneID") = 1` then **CHECKED**
   - **SizeZoneID = 2**
     - If `objRS("SizeZoneID") = 2` then **CHECKED**
<INPUT TYPE=INPUT NAME=ProdID VALUE=""><TD><INPUT TYPE=INPUT NAME=SupID VALUE=""><TD><INPUT TYPE=SUBMIT NAME="btnBack" VALUE="" onClick="return btnBack()">
</TD><TR>
</TABLE></FORM>
<% objRS.Close Set objRS = Nothing %>  
<!--This is template file please delete and enter code here-->  
<!-- #INCLUDE FILE="Footer.asp" -->  
<**********************************>.255
EmpSendMail.asp

<%@ Language=VBScript %>
<!-- #INCLUDE FILE="DataConn.asp" -->

<%'
  '**********************************************************
  ' EmpSendMail.asp
  ' Program by Sumuscha Teesri
  ' May 9, 2000
  '**********************************************************>
Response.Buffer = True
' Create mail object
Dim objSendMail
Set objSendMail = CreateObject("CDONTS.NewMail")

' Send Mail
With objSendMail
  .From = Session("Email")
  .To = Request.Form("Recipient")
  .Subject = Request.Form("Subject")
  .Body = Request.Form("Message")
  .Send
End With
Set objSendMail = Nothing

' Redirection to MainMenu.asp
Response.Clear
Response.Redirect "EmpMessage.asp?Msg=Your mail was send.<BR>Thank you."
&_ 
  "&redirect=MainMenu.asp"
%

Footer.asp

<!**************************************************************************
* Footer.asp
* Program by Sumuscha Teesri
* Footer Template for Macrotoys admin page
* May 9, 2000
**************************************************************************>
<!-*--Begin Close--*-->
<!-**************-->

<** Link bar **>
<HR>
<!-- <TABLE BORDER=0 WIDTH=600>
  <TR ALIGN=CENTER>
    <TR ALIGN= CENTER>
      <!-- Link To Browser page -->
      <TD WIDTH=200><!-- If blnBrowserPage = True Then
        Response.Write "<B>[Browse the

Products]"<B></TD>"
Else
Response.Write "<A HREF=""Browser.asp""><B>[Browse the Products]</B>" End If
%


<!* Link to Register for new user or shopping cart for logon user *><TD WIDTH=200><%If Session("CusID")= "" Then Response.Write "<A HREF=""Register.asp""><B>[Your Account]</B>" Else If Session("Shopping") = "True" Then Response.Write "<A HREF=""ShoppingCart.asp""><B>[Shopping Cart]</B>" Else If Session("Shopping") =
<B>[Shopping Cart]</B><BR>

End If

End If

"<BR>

</TR>
</TABLE>
</TD>
</TR></TABLE>

<!-- General Information -->
<TABLE BORDER=0 WIDTH=600 CELLPADDING="0" CELLSPACING="0">
<TR>
  <TD VALIGN="TOP" ALIGN="CENTER" WIDTH=100 ROWSPAN=6 onMouseOver="window.status='Advertising with Macrotoy';return true" onMouseOut="window.status='';return true;">
    <%
      Dim objAd
      Set objAd = Server.CreateObject("MSWC.AdRotator")
      Response.Write objAd.GetAdvertisement("adrot.txt")
    %>
  </TD>
</TR>
<TR>
  <TD VALIGN="TOP" WIDTH="200" ROWSPAN=5><FONT COLOR="#22FF22">Contact Information:</FONT></TD>
</TR>
<TR>
  <TD WIDTH 300 >Sumuscha Teesri</TD>
</TR>
<TR>
  <TD WIDTH="300">3249 Shepard Ln, San Bernardino, CA 92407</TD>
</TR>
<TR>
  <TD WIDTH="300">Phone: 1-530-685-5016<br>FAX: 1-530-235-7314</TD>
</TR>
<TR>
  <TD WIDTH="300">Website:<SMALL>
    <LI><A HREF="Http://members.xoom.com/steeersi">XOOM.COM</A><BR>
    <LI><A HREF="Http://geocity.com/steeersi">GEOCITY.COM</A><BR>
    <LI><A HREF="Http://webandhost.com">WEBANDHOST.COM</A></SMALL>
</TD>
</TR>
<TR>
  <TD>&nbsp</TD>
</TR>
<TR>
  <TD COLSPAN=2><H5><A HREF="mailto:teesri@hotmail.com">Questions or comments about this website.</A>
    <BR><FONT COLOR="#22FF22">Copyright © 1999-2000</FONT>
  &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nabla
</H5></TD>
</TR>
Global.asp

<SCRIPT LANGUAGE=VBScript RUNAT=Server>
' Include file needed for send mail
<!--METADATA TYPE="TypeLib" NAME="Microsoft CDO for NTS 1.2 Library"
VERSION="1.2"-->
******************************************************************************
' Global.asp
' Program by Sumuscha Teesri
' Global file provide setup variable
' May 4, 2000
******************************************************************************

Sub Application_OnStart
    ' Create Visits and Active counter
    Application("Visits") = 0
    Application("Active") = 0
    Application("IISAd") = 0
    Application("BackOfficeAd") = 0
    Application("WebAndHostAd") = 0
End Sub

Sub Application_OnEnd
End Sub

Sub Session_OnStart
    Dim strStartPage, strCurrentPage
    Application.Lock
    Application("Active") = Application("Active") + 1
    Application("Visits") = Application("Visits") + 1
    Application.UnLock

    ' Make sure that user will alway start with the default page
    strStartPage = "default.asp"
    strCurrentPage = Request.ServerVariables("SCRIPT_NAME")

    ' Do a case-insensitive compare, and if they
    ' don't match, send the user to the start page.
    If strcomp(strCurrentPage,strStartPage,1) then
        ' Turn on buffer for compactible
        Response.Buffer = True
End Sub

259
Response.Clear
Response.Redirect(strStartPage)
End If

End Sub

Sub Session_OnEnd
    ' Reduce active counter when user logout
    Application.Lock
        Application("Active") = Application("Active") - 1
    Application.UnLock
End Sub
</SCRIPT>

<HEAD>
<!** Define meta data **>
<META NAME="Programmer" CONTENT="Sumuscha Teesri">
<META NAME="Designner" CONTENT="Sumuscha Teesri">
<META NAME="University" CONTENT="California State University, San Bernarino">
<META NAME="Department" CONTENT="Business Administration">
<META NAME="Degree" CONTENT="Master of Business Administration">
<META NAME="Concentration" CONTENT="Information Management">
<META NAME="Subject" CONTENT="ADMIN 998 Comprehensive Project">
<META NAME="Committee Chair" CONTENT="Prof. Frank M. Lin, Ph.D.">
<META NAME="Second Faculty Reader" CONTENT="Prof. Harold Dyck, Ph.D.">
<META NAME="Department Chair" CONTENT="Prof. Walter T. Stewart, Ph.D.">

<TITLE>Macrotoys - Comprehensive Project</TITLE>

<!** Animation botton **>
<script language="JavaScript" fptype="dynamicanimation">
<!--
function dynAnimation() {}
function clickSwapImg() {}
//-->
</script>
<script language="JavaScript1.2" fptype="dynamicanimation" src="animate.js"></script>
<!** Clock Function **>
<SCRIPT LANGUAGE="JavaScript">
function fnClock()
{
    var oDate=new Date();
    var sSwitch="AM";
    var iHours=oDate.getHours();
    if(iHours>=12)
    {
        sSwitch="PM";
    }
    if(iHours>12)
    {
        iHours-=12;
    }
    var sMinutes=oDate.getMinutes() + "";
    if(sMinutes.length==1)
    {
        sMinutes="0" + sMinutes;
    }
    var sSeconds=oDate.getSeconds() + "";
    if(sSeconds.length==1)
    {
        sSeconds="0" + sSeconds;
    }
    oTimer.innerHTML=iHours + ":" + sMinutes + ":" + sSeconds + " " + sSwitch;
}
</SCRIPT>
</HEAD>

<BODY ONLOAD="dynAnimation()" LANGUAGE="Javascript1.2" BGCOLOR="#000080" TEXT="33BBBB" ALINK="#CC3333" LINK="DDFF00" VLINK="#CC33FF" TOPMARGIN="0" LEFTMARGIN="0">
<A NAME="PageTop"></A>
<TABLE BORDER="0" WIDTH=100% CELLPACING="0" CELLPADDING="0">
    <TR>
        <!-- Show Date, Time and Statistic-->
    </TR>
    <TD ROWSPAN=2 VALIGN="Top" WIDTH=150><FONT COLOR="#22FF22"><CENTER>
        <!-- Show current Month, Day, and Year -->
        <STRONG><%= Response.Write (MonthName(Month(DATE)))%>& 
        (Day(Date))& 
        & (Year(Date))%>
    </FONT><BR>
        <!-- Show Time using JavaScript -->
        <B><FONT COLOR="#DDFF22" ID=oTimer ><%= Time()%></FONT></B>
        <SCRIPT LANGUAGE="JavaScript">
            var oInterval="";
            while (oInterval=="")oInterval=window.setInterval("fnClock()",1000);}
        </SCRIPT>
    </TD>
</TABLE>
Total Visitors: 9870
Active Users: 1340

MailConn.asp

```<%
'******************************************************************************
'* MailConn.asp
'* Program by Sumuscha Teesri
'* Mail connection used
'* to connect to mail serverce
'* May 11, 2000
'******************************************************************************
Dim objInbox, objCurSession
Dim strUserName, strUserEmail, colMsgs

' Default user used to connect to mail service
' Change this value if necessary
strUserName = "IUSR"
' Employee email address
strUserEmail = Session("Email")
Set objCurSession = CreateObject("CDONTS.Session")
' Log on to the mail server
objCurSession.LogonSMTP strUserName, strUserEmail
Set Session("CurSession") = objCurSession
Set objInbox = objCurSession.Inbox
Set colMsgs = objInbox.Messages
%>```
```vbscript
<!-- #INCLUDE FILE="MailConn.asp" -->

'******************************************
' MailMan.asp
' Program by Sumuscha Teesri
' Mail Manager used to manage mail
' May 11, 2000
'******************************************
Response.Buffer = True
Dim intLoop

' Check whether employee delete some mails
If Len(Request("btnDelete")) Then
    ' Delete the selected mails
    For intLoop = 1 To Request("Delete").Count
        Response.Write colMsgs.Item(intLoop)
        colMsgs(Request("Delete")(intLoop)).Delete
    Next
End If
Set colMsgs = Nothing

' Employee want to send new mail
If Len(Request("btnNewMail")) Then
    Response.Clear
    Response.Redirect "NewMail.asp"
End If

' Employee want to reply the mail
If Len(Request("btnReply")) Then
    ' For debug only
    ' Response.Write Request("ReplyAdd")
    Response.Clear
    Response.Redirect "NewMail.asp?replyadd=" & Request("ReplyAdd") &
    "&Subject=" & Request("Subject")
End If

' For debug only
'Response.Write request("btnReply")
'Response.Write Request("btnBack")

' Back to the mainmenu
Response.Clear
Response.Redirect "MainMenu.asp"
%
```
MainMenu.asp

<%@ Language=VBScript %>
<!**********************************
* MainMenu.asp
* Program by Sumuscha Teesri
* May 11, 2000
**********************************>

<!--*
Include file needed >
<!--
<-- #INCLUDE FILE="Head.asp" -->
<-- #INCLUDE FILE="NavigatorBar.asp" -->
<-- #INCLUDE FILE="MailConn.asp" -->

<!--* Begin main *-->

<% ' Make sure that only employee is allowed to access this site
Response.Buffer = True
If Session("EmpID") = "" Then
   Response.Redirect "EmpLogIn.asp"
End If

Dim strMessage, intLoop

' Write number of mail
strMessage = "<CENTER><H1><FONT COLOR=#22FF22>Main Menu</FONT>" & _
   "</H1></CENTER>" & _
   "Welcome, " & Session("FirstName") & ". You have " & _
   colMsgs.Count & " messages in your inbox. <BR><BR>"

If (colMsgs.Count > 0) Then
   ' Employee has some mails
   ' Write table head
   strMessage = strMessage & "<FORM METHOD=POST ACTION=MailMan.asp>"
   & _
   "<TABLE BORDER=1 CELLPADDING=1 & _
   CELLSPACING=1 WIDTH=600><TR BGCOLOR=#FFFF22>" & _
   "<TD WIDTH=80 ALIGN=LEFT><STRONG>Delete</STRONG></TD>" & _
   "<TD WIDTH=180 ALIGN=LEFT><STRONG>From</STRONG></TD>" & _
   "<TD WIDTH=260 ALIGN=LEFT><STRONG>Subject</STRONG></TD>" & _
   "<TD WIDTH=80 ALIGN=LEFT><STRONG>Date</STRONG></TD>"/&TR>

   ' Show mail information
   For intLoop = 1 to colMsgs.Count
      strMessage = strMessage & "<TR><TD WIDTH=80 ALIGN=LEFT>"
      & _
   "<TD WIDTH=80 ALIGN=LEFT>"
   End For

264
<INPUT TYPE=CheckBox NAME=Delete Value="" TD WIDTH=180> &
colMsgs(intLoop).Sender & _
HREF=ViewMessage.ASP?MsgID=" & _
intLoop & "> & colMsgs(intLoop).Subject & _
"<TD WIDTH=80 ALIGN=RIGHT>" & _
FormatDateTime(colMsgs(intLoop).TimeSent,2) & "</A></TD>" & _
Next
   ' Write button
   strMessage = strMessage & ">
"<INPUT TYPE=SUBMIT NAME="btnNewMail"
VALUE="$Compose Mail$"/>
" _
"<INPUT TYPE=SUBMIT NAME="btnDelete"
VALUE="$Delete Mail$"></FORM>"
End If
Response.Write strMessage

******************************************************************
* The following code used for debug and testing only.
*
******************************************************************

' For Each objItem In Session.Contents
    If Not IsArray(Session(objItem)) Then
        Response.Write objItem & 
"" = " & Session(objItem) & 
"<BR>"
    Else
        Response.Write objItem & 
" Is an Array<BR>"
    End If
' Next
%
<!-- #INCLUDE FILE="Footer.asp" -->

<****************** NavigatorBar.asp

<!******************
* NavigatorBar.asp
* Program by Sumuscha Teesri
* Navigation bar Template
* May 9, 2000
******************>
<!------------->
<!-- Begin Navigation -->
<!------------->

<TABLE BORDER="0" WIDTH=150 CELLPACING="0" CELLPADDING="0"
ALIGN="LEFT">
    <TR>
        <TD ALIGN="CENTER" WIDTH="150" HEIGHT="30"><IMG
SRC="image\wellcome.gif"></TD>
    </TR>
</TABLE>
If login user show user name or show Guess

Dim strMsg
strMsg = Session("FirstName") & "Main Menu"; return true"
& _
"Customer"; return true"
& _
"Order"; return true"
& _
"Supplier & & Product"; return true"
& _
"Category"; return true"
& _
return true"
<SMALL>" & _
"Change Password"; return true"
& _
"Default"; return true"
& _
"Log Out";

Else
strMsg = "</STRONG><FONT></TD>"
End If
Response.Write strMsg
%>

</TR>
</TABLE>

<!-- Create fream table for body -->
<TABLE BORDER="0" WIDTH=80% CELLPADDING="0" CELLCSPACING="0">
<tr>
<td ALIGN="LEFT" VALIGN="TOP" WIDTH=85%>
NewMail.asp
</td>
</tr>
</TABLE>

<% @,Language=VBScript %>
<!**************************>
* NewMail.asp
* Program by Sumuscha Teesri
* May 11, 2000
<**************************>

<! Include file needed >
<!-- #INCLUDE FILE="Head.asp" -->
<!-- #INCLUDE FILE="NavigatorBar.asp" -->

<--************************--><
---*
Begin main  *--->
<--************************--><

' Make sure that only employee is allowed to access this site
Response.Buffer = True
If Session("EmpID") = "" Then
    Response.Redirect "EmpLogIn.asp"
End If

' Write mail header information
Dim strMessage
strMessage = "<CENTER><H1><FONT COLOR=#22FF22>"

' Check wheather employee reply mail
If Len(Request("ReplyAdd")) Then
    ' Reply Mail
    strMessage = strMessage & "Reply Mail"
Else
    ' Compose new mail
    strMessage = strMessage & "Compose New Mail"
End If
strMessage = strMessage & "</FONT></H1></CENTER>"
Response.Write strMessage

' Write input form
%>
<!-- Create input form -->
<form ACTION="EmpSendMail.asp" METHOD="POST">
Please enter your Recipient Email address:<br>
<INPUT NAME=Recipient SIZE=50 VALUE="">
<%If Len(Request("ReplyAdd")) Then
   Response.Write Request("ReplyAdd")
   End If%">

Please enter the Subject Here:<BR>
<INPUT NAME=Subject SIZE=50 VALUE="">
<%If Len(Request("ReplyAdd")) Then
   Response.Write Request("Subject")
   End If%">

Enter your message here:<BR>
<TEXTAREA NAME="Message" COLS=70 ROWS=7></TEXTAREA><P>
<INPUT TYPE="SUBMIT" VALUE="Send">&nbsp;&nbsp;
<INPUT TYPE="RESET"> </FORM>

<!--This is template file please delete and enter code here-->
<!-- #INCLUDE FILE="Footer.asp" -->
<!*************

NonNavigatorBar.asp

<**************************
* NonNavigatorBar.asp
* Program by Sumuscha Teesri
* NonNavigatorBar Template
* May 11, 2000
**************************>

<%** In case need head and footer template
but not the navigation bar
%>
<TABLE BORDER="0" WIDTH=100% CELLPADDING="0" CELLSPACING="0">
<TR>
<TD ALIGN="LEFT" VALIGN="TOP" WIDTH=100%>
<!-- End of nonbrowsbar.asp -->

Order.asp

<%@ Language=VBScript %>
<**************************
* Order.asp
* Program by Sumuscha Teesri
* May 11, 2000
**************************>

<! Include file needed >
<!-- #INCLUDE FILE="Head.asp" -->
<!-- #INCLUDE FILE="NavigatorBar.asp" -->

<!--**************************-->
<!--* Begin main
*-->
<!--**************************-->
<!%>
' Make sure that only employee is allow to access this site
Response.Buffer = True
If Session("EmpID") = "" Then
    Response.Redirect "EmpLogIn.asp"
End If

Dim strMessage
strMessage = "<CENTER><H1><FONT COLOR=#22FF22>Order Status</FONT></H1></CENTER>"

' Connect to the database select only the records needed from the selected tables
Set objRS = Server.CreateObject("ADODB.Recordset")
strSQL = "SELECT OrderID, CompanyName, S.Description OrderStatus, " & _
    "Ca.Description Carrier,Tracking " & _
    "FROM Orders O, Customers C, Employees E, " & _
    "OrderStatus S, Carriers Ca " & _
    "WHERE C.EmpID = E.EmpID and C.CusID = O.CusId " & _
    "and S.ID = O.Status and Ca.CarrierID = O.CarrierID"

' Check if employee want to search for specific order
If Len(Request("SearchMethod")) Then
    ' For debug only
    Response.Write Request("SearchMethod")

    ' Create SQL depends on searching Method
    Select Case Request.Form("SearchMethod")
    Case "OrderID"
        strSQL = strSQL & "OrderID Like " & Request("Keyword") & ";"
    Case "CusID"
        strSQL = strSQL & "C.CusID Like " & Request("Keyword") & ";"
    Case "Email"
        strSQL = strSQL & "C.Email Like " & Request("Keyword") & ";"
    Case "CompanyName"
        strSQL = strSQL & "CompanyName Like " & Request("Keyword") & ";"
    Else
        ' Employee do not specific criteria select only the order related to the employee
        strSQL = strSQL & "C.EmpID = " &Session("EmpID") & ";"
    End Select
Else
    ' Employee do not specific criteria select only the order related to the employee
    strSQL = strSQL & "C.EmpID = " & Session("EmpID") & ";"
End If

' Connect to database
objRS.Open strSQL, objConn, adOpenForwardOnly, adLockReadOnly, adCmdText
If Not objRS.EOF Then
    ' Write table head
    strMessage = strMessage & "<TABLE BORDER=1 WIDTH=600>" & _

"<TR><TD ALIGN= CENTER WIDTH= 80><B>OrderID</B></TD>" & _ "<TD ALIGN= CENTER WIDTH= 220><B>Company Name</B></TD>" & _ "<TD ALIGN= CENTER WIDTH= 100><B>Status</B></TD>" & _ "<TD ALIGN= CENTER WIDTH= 100><B>Carrier</B></TD>" & _ "<TD ALIGN= CENTER WIDTH= 100><B>Tracking</B></TD>"

While Not objRS.EOF
    ' Write detail
    strMessage = strMessage & "<TR><TD ALIGN= CENTER WIDTH= 80>" & _
    objRS("OrderID") & _ "</A></TD>" & _
    objRS("CompanyName") & _ "</A></TD>" & _
    objRS("OrderStatus") & _ "</A></TD>" & _
    objRS("Carrier") & _ "</A></TD>" & _
    objRS("Tracking") & _ "</A></TD>"

    objRS.MoveNext
    strMessage = strMessage & "<A HREF=EmpOrderDetail.asp?OrderID= " & _
    objRS("OrderID") & ""><SMALL>" & _
    objRS("CompanyName") & ""><SMALL>" & _
    objRS("OrderStatus") & ""><SMALL>" & _
    objRS("Carrier") & ""><SMALL>" & _
    objRS("Tracking") & ""></SMALL></TD></TR>"

Else
    strMessage = strMessage & "<BR>There is no orders found."
End If
'

' Close database
objRS.Close
Set objRS = Nothing
Response.Write strMessage
'

' Write search form
%>

<FORM METHOD= POST>
<B>Search Order Detail</B>/</B><BR>
Please select one of the following searching method:<BR>
.INPUT TYPE=RADIO NAME= "SearchMethod" VALUE= "OrderID" CHECKED>Order ID &nbsp; &nbsp;
&nbsp; &nbsp;
<input TYPE=RADIO NAME= "SearchMethod" VALUE= "CusID" >Customer ID &nbsp; &nbsp;
&nbsp; &nbsp;
(INPUT TYPE=RADIO NAME= "SearchMethod" VALUE= "Email" >Customer Email &nbsp; &nbsp;
<INPUT TYPE=RADIO NAME="SearchMethod" VALUE="CompanyName">Company Name
</FORM>

<table>
<thead>
<tr>
<th>Search Keyword:</th>
<th>&lt;INPUT TYPE=TEXT NAME=&quot;Keyword&quot;&gt; Search Keyword:</th>
</tr>
</thead>
</table>

* WildCard: Use an underscore (_) for one character or a percent sign (%) for many characters.

<table>
<thead>
<tr>
<th>Example:</th>
<th>Example:</th>
</tr>
</thead>
<tbody>
<tr>
<td>%Computer% for every company with the word Computer</td>
<td></td>
</tr>
</tbody>
</table>

<!-- #INCLUDE FILE="Footer.asp" -->

<Product.asp>

<%@ Language=VBScript %>
<% Response.Buffer = True

'* Product.asp
'* Program by Sumuscha Teesri
'* May 19, 2000

%>

<! Include file needed >
<!-- #INCLUDE FILE="Head.asp" -->
<!-- #INCLUDE FILE="NavigatorBar.asp" -->

<!--*************->
<!-* Begin main *-->
<!--*************->

<%
'Make sure that only employee login
If Session("EmpID") = "" Then
    Response.Redirect "EmpLogIn.asp"
End If

'Write head to client's browser
Dim strMessage, intSupID
intSupID = CInt(Request("SupID"))

If Len(Request("btnBack")) Then
    'Back to Supplier detail
    Response.Clear
    Response.Redirect "SupDetail.asp?SupID=" & intSupID
End If

strMessage="<CENTER><H1><FONT COLOR=#22FF22>Product List</FONT></H1></CENTER>"

271
Set objRS = server.CreateObject("ADODB.Recordset")

' Select supplier's company name
strSQL = "SELECT CompanyName FROM Suppliers WHERE SupID=" & _
intSupID & ";"
objRS.Open strSQL, objConn, adOpenForwardOnly, adLockReadOnly, adCmdText
strMessage=strMessage & "<BR><B><FONT COLOR=#22FF22>Supplier: " & _
objRS("CompanyName") & "<BR>"
objRS.Close

' Select only information needed
strSQL = "SELECT ProdID, ProdName, StockQuantity, ReorderPoint, SalePrice " & _
"FROM Products WHERE SupID=" & _
intSupID & ";"

' Connect to database
objRS.Open strSQL, objConn, adOpenStatic, adLockReadOnly
If Not objRS.EOF Then
strMessage = strMessage & "<TABLE BORDER=0 WIDTH= 550><TR><TD WIDTH=450>" & _
"<A HREF=AddProd.asp?SupID=" & Request("SupID")
& _
"& <B>Add New Product</B></A></TD></TR></TABLE>" & _
"<TD WIDTH=625><TR>" & _
"<TD WIDTH=30 ALIGN=CENTE" "<B>Product Name</B><B>/TD>" & _
"<TD WIDTH=90 ALIGN=CENTE" "<B>Quantity</B></TD>" & _
"<TD WIDTH=90 ALIGN=CENTE" "<B>Reorder</B></TD>" & _
"<TD WIDTH=90 ALIGN=CENTE" "<B>Sale Price</B></TD>" & _
"While Not objRS.EOF
  ' Write table body
  strMessage = strMessage & "<TR><TD WIDTH=30 ALIGN=CENTE" "<B>ProdID</B></TD>" & _
"<A HREF=EmpProDetail.asp?ProdID=" & objRS("ProdID") & "/A</TD><SMALL>" & _
"<TD WIDTH=300><SMALL>" & _
"<A HREF=EmpProDetail.asp?ProdID=" & objRS("ProdID") & "/A</TD><SMALL>" & _
"<TD WIDTH=90 ALIGN=RIGHT" "<B>ProdID</B></TD>" & _
"<TD WIDTH=90 ALIGN=RIGHT" "<B>ProdID</B></TD>" & _
"<TD WIDTH=90 ALIGN=RIGHT" "<B>ProdID</B></TD>" & _
"<TD WIDTH=90 ALIGN=RIGHT" "<B>ProdID</B></TD>" & _
"FormatCurrency(objRS("SalePrice")) & "</TD></TR>"
objRS.MoveNext
"</TD></TR>"
Wend
strMessage = strMessage & "</TABLE>"
Response.Write strMessage
Else
' Find not found
Response.Write "<BR><BR><CENTER>" & _
"<P>Sorry We are currently do not have any supplier you are looking for." & _
"<BR>Please try again.<BR/></CENTER><BR><BR>"
End If
objRS.Close
Set objRS = Nothing
%
<form method=POST>
<input type=submit name="btnBack" value="« Back">
<input type=hidden name=SubID value="<%=intSupID%»">
</form>
<!—#INCLUDE FILE="Footer.asp" —>

RecProd.asp

<%@ Language=VBScript %>
<%Response.Buffer = True%>
<!—#INCLUDE FILE="DataConn.asp" —>
<%
' For debug only
'******************************************************************************
'Response.WriteLine(Request("SupID").Value & "<BR>"
'Response.WriteLine(Request("ProdName").Value & "<BR>"
'Response.WriteLine(Request("Description").Value & "<BR>"
'Response.WriteLine(Request("PurchasePrice").Value & "<BR>"
'Response.WriteLine(Request("SalePrice").Value & "<BR>"
'Response.WriteLine(Request("StockQuantity").Value & "<BR>"
'Response.WriteLine(Request("ReorderPoint").Value & "<BR>"
'Response.WriteLine(Request("SaleUnit").Value & "<BR>"
'Response.WriteLine(Request("PurchaseUnit").Value & "<BR>"
'Response.WriteLine(Request("Age").Value & "<BR>"
'Response.WriteLine(Request("Gender").Value & "<BR>"
'Response.WriteLine(Request("Measure").Value & "<BR>"
'Response.WriteLine(Request("SizeZoneID").Value & "<BR>"
'Response.WriteLine(Request("Weight").Value & "<BR>"
'******************************************************************************
' Add a new record to the products table
Set objRS = Server.CreateObject("ADODB.Recordset")
objRS.Open "Products", objConn, adOpenForwardOnly, adLockOptimistic, adCmdTable
objRS.AddNew
    objRS("SupID") = CInt(Request("SupID"))
    objRS("ProdName") = Request("ProdName")
    objRS("Description") = Trim(Request("Description"))
    objRS("PurchasePrice") = CCur(Request("PurchasePrice"))
    objRS("SalePrice") = CCur(Request("SalePrice"))
    objRS("StockQuantity") = CInt(Request("StockQuantity"))
    objRS("ReorderPoint") = CInt(Request("ReorderPoint"))
    objRS("SaleUnit") = Request("SaleUnit")
    objRS("PurchaseUnit") = Request("PurchaseUnit")
    objRS("Age") = Request("Age")
    objRS("Gender") = Request("Gender")
    objRS("Measure") = CSng(Request("Measure"))
    objRS("SizeZoneID") = Request("SizeZoneID")
    objRS("Weight") = CSng(Request("Weight"))
objRS.Update
objRS.Close
Set objRS = Nothing

' Redirection after the new record was added
Response.Clear
Response.Redirect "EmpMessage.asp?MSG=New product was added" & _
    "&Redirect=SupDetail.asp?SupID=" & CInt(Request("SupID"))
%

---

RecSup.asp

<%# Language=VBScript %>
<%Response.Buffer = True%>
<!-- #INCLUDE FILE="DataConn.asp" -->
<%
'**********************************************************************
' * RecSup.asp
' * Program by Sumuscha Teesri
' * May 19, 2000
'**********************************************************************

Dim strError, strValue
    strError ="

Dim objRS
Set objRS = Server.CreateObject("ADODB.Recordset")
'**********************************************************************
' Check Error
'**********************************************************************
strSQL = "SELECT CompanyName FROM Suppliers WHERE CompanyName='" & _
    Request.Form("CompanyName")& ";" 
objRS.Open strSQL, objConn

If Not objRS.EOF Then
' CompanyName is not unique
strError = "ComUnique=False"
End If
objRS.Close

strSQL = "SELECT Email FROM Suppliers WHERE Email='' & _
Request.Form("Email")& '';"
objRS.Open strSQL, objConn
If Not objRS.EOF Then
' Email address is not Unique
If strError = "" Then
  strError = "EmailUnique=False"
Else
  strError = strError & "&EmailUnique=False"
End If
End If
objRS.Close

If Len(strError) Then
' There are some errors re-enter information
Response.Clear
Response.Redirect "AddSup.asp" & strError
Else
' No error at all add new record
' Connect to Suppliers database
objRS.Open "Suppliers", objConn, adOpenForwardOnly, _
adLockOptimistic, adTable
objRS.AddNew
  ' Add new record to Suppliers table
Response.Write Request.Form("CompanyName")
objRS("CompanyName") = Request.Form("CompanyName")
objRS("Address1") = Request.Form("Address1")
objRS("Address2") = Request.Form("Address2")
objRS("City") = Request.Form("City")
objRS("Region") = Request.Form("Region")
objRS("PostalCode") = Request.Form("PostalCode")
objRS("Country") = Request.Form("Country")
objRS("Phone") = Request.Form("Phone")
objRS("Fax") = Request.Form("Fax")
objRS("WebSite") = Request.Form("WebSite")
objRS("Title") = Request.Form("Title")
objRS("LastName") = Request.Form("LastName")
objRS("FirstName") = Request.Form("FirstName")
objRS("Direct") = Request.Form("Direct")
objRS("Email") = Request.Form("Email")
objRS.Update
objRS.Close
Set objRS = Nothing

' Redirection back to suppliers menu
Response.Clear
Response.Redirect "EmpMessage.asp?MSG=""A new supplier record has been added"
" & _
  "&Redirect="Supplier.asp"
End If

%>

275
Stats.asp

%@ Language=VBScript @
<!-- #INCLUDE FILE="DataConn.asp" -->
<html>
<head>
<title>Web Statistic</title>
</head>
<body>

 Dim objRS, strMessage
 Set objRS = Server.CreateObject("ADODB.Recordset")
 objRS.Open "WebStats", objConn, adOpenStatic, adLockReadOnly, adCmdTable

 ' For debug only
 'Response.Write objRS("Browser") & objRS("Version") &
 objRS("RemoteAddress") & _
 'objRS("RemoteHost")
 strMessage ="<TABLE BORDER=0 WIDTH="80%"">" & _
 "<TR><TD WIDTH="80%">" & _
 objRS.RecordCount & _
 " visitors.&/TD>" & _
 "</TR></TABLE>" & _
 "<TABLE BORDER=1 WIDTH="100%"">" & _
 "<TR><TD WIDTH="10%">&<B>Browser</B></TD>" & _
 "<TD WIDTH="10%">&<B>Version</B></TD>" & _
 "<TD WIDTH="15%">&<B>Address</B></TD>" & _
 "<TD WIDTH="15%">&<B>HostName</B></TD>" & _
 "<TD WIDTH="40%">&<B>Reference</B></TD>" & _
 "<TD WIDTH="10%">&<B>Date</B></TD>"& <TR>"

While Not objRS.EOF
 strMessage = strMessage & _
 "<TD WIDTH="10%">&" & objRS("RemoteAddress") & _
 "<TD WIDTH="15%">& objRS("RemoteHost")
 & "</TD>" & _
 "<TD WIDTH="40%">& objRS("HttpRef") &_
 "<TD WIDTH="10%">& _
 FormatDateTime(objRS("Activedate"),2) &

 "</TD></TR>"

 objRS.MoveNext
 WEND
 objRS.Close
 Set objRS = Nothing
 Response.Write strMessage
%
</table>
</body>
SubCat.asp

<!-- Include file needed -->
<SCRIPT Language="JavaScript">
  function VerifyData()
  {
    if (document.frmAddSubCat.SubCatName.value=="")
    {
      alert("Please provide the subcategory's name");
      return false;
    }
    else
    {
      alert("The new record was added");
      return true;
    }
  }
</SCRIPT>
</HTML>

Dim strMessage
strMessage = "<CENTER><H1><FONT COLOR=#22FF22>Subcategory List</FONT></H1></CENTER>" & _
  "<TABLE BORDER=1 WIDTH=600>" & _
    "<TR><TD ALIGN=LEFT WIDTH=50><B>ID</B></TD>" & _
      "<TD ALIGN=LEFT WIDTH=150><B>Name</B></TD>" & _
        "<TD ALIGN=LEFT WIDTH=300><B>Description</B></TD>" & _
"<TD ALIGN=CENTR WIDTH=100><B>Tools</B></TD></TR>

Set objRS = Server.CreateObject("ADODB.Recordset")
strSQL = "SELECT * FROM Subcategories WHERE CatID=" & CInt(Request("CatID"))
objRS.Open strSQL, objConn, adOpenForwardOnly, adLockReadOnly, adCmdText

WHILE Not objRS.EOF
    strMessage = strMessage & "<TR><TD WIDTH=50><A HREF=SubCatDetail.asp?SubCatID=" & _
    objRS("SubCatID") & ""><" & objRS("SubCatID") & "</A></TD>" & _
    "<TD WIDTH=150><A HREF=SubCatDetail.asp?SubCatID=" & _
    objRS("SubCatName") & ""><" & objRS("SubCatID") & "</A></TD>" & _
    "<TD WIDTH=300><A HREF=SubCatUpdate.asp?SubCatID=" & _
    objRS("Description") & ""><" & objRS("SubCatID") & "</A></TD>" & _
    "<TD WIDTH=100><A HREF=Modify.asp?SubCatID=" & _
    objRS("SubCatID") & "">Modify</A></TD>" & _
    objRS.MoveNext
WEND

objRS.Close
Set objRS = Nothing
strMessage = strMessage & "</TABLE>"
Response.Write strMessage %>

<FONT COLOR=#22FF22>B>Add new Subcategory</B></FONT><BR>
<TABLE BORDER=0 WIDTH=400><TR><TD>Name:</TD><TD>Description:</TD><TD></TD></TR>
<INPUT TYPE=HIDDEN NAME=CatID VALUE="%Request("CatID")%" >>
<TR><TD><INPUT TYPE=TEXT NAME=SubCatName></TD>
<TD><INPUT TYPE=TEXT NAME=Description></TD>
<INPUT TYPE=SUBMIT NAME=btnAdd VALUE="Add new record" onClick="return VerifyData()"></TD>
<INPUT TYPE=SUBMIT NAME=btnBack VALUE="<< Back"></TD></TR></TABLE></FORM>
<!-- #INCLUDE FILE="Footer.asp" -->
SubCatDetail.asp

<%@ Language=VBScript %>
<% Response.Buffer = True %>
'* SubCatDetail.asp
'* Program by Sumuscha Teesri
'* May 20, 2000
'*******************************************************************************
%

<! Include file needed >
<!-- #INCLUDE FILE="Head.asp" -->
<!-- #INCLUDE FILE="NavigatorBar.asp" -->
<!--******************************************************************************>
<!--* Begin main -->
<!--******************************************************************************>
%

' Make sure that only employee can login
If Session("EmpID") = "" Then
  Response.Redirect "EmpLogIn.asp"
End If

' Connect to database
Set objRS = Server.CreateObject("ADODB.Recordset")
Dim strMessage, intSubCatID, intCatID
intSubCatID = Cint(Request("SubCatID"))

' Find Categories and subcategory name
strSQL = "SELECT * FROM Categories, Subcategories WHERE SubCatID=" & intSubCatID
objRS.Open strSQL, objConn, adOpenForwardOnly, adLockReadOnly, adCmdText
intCatID = Cint(objRS("CatID"))

' Show that name on screen
strMessage ="<CENTER><H1><FONT COLOR=#22FF22>Subcategory Detail</FONT></H1></CENTER>" & _
  "<TABLE BORDER=0 WIDTH=600>" & _
  "<TR><TD><FONT COLOR=#22FF22>B.Category: " & _
  objRS("CatName") & _
  "</TD><FONT COLOR=#22FF22>B.Subcategory: " & _
  objRS("SubCatName") & _
  "</TR></TD>" & _
  "<TR><TD><CENTER WIDTH=550>" & _
  "<TD ALIGN=LEFT WIDTH=100>" & _
  "<TD ALIGN=LEFT WIDTH=95>" & _
  ID</B></TD>" & _
  "<TD ALIGN=LEFT WIDTH=400>" & _
  Name</B></TD>" & _
  "<TD ALIGN=LEFT WIDTH=100>" & _
  Checked</B></TD>" & _
  "<FORM ACTION=AddProdCat.asp METHOD=POST NAME=frmAddProdCat>" & _

' Close recordset after used
objRS.Close

' Select all product but only the same subcategory will has SubCatID, ' otherwise, subcat id is null.
'********************************************************************
strSQL = "SELECT P.ProdID PID, ProdName, SubCatID " & _
"FROM Products P LEFT JOIN ProductCats C ON P.ProdID = C.ProdID " & _
"WHERE SubCatID= " & intSubCatID & _
" UNION (SELECT ProdID, ProdName, SubCatID = NULL " & _
"FROM Products " & _
"WHERE ProdID NOT IN " & _
"(SELECT P.ProdID " & _
"FROM Products P left join ProductCats C on P.ProdID = C.ProdID " & _
"WHERE SubCatID=" & intSubCatID & _
") " & _
") " & _
" Order By PID "
'********************************************************************

objRS.Open strSQL, objConn, adOpenForwardOnly, adLockReadonly, adCmdText

' Write the table body
WHILE Not objRS.EOF
strMessage = strMessage & _
"<TR><TD ALIGN=CENTER WIDTH=100>" & objRS("PID") & _
"</TD>" & _
"<TD WIDTH=400>" & objRS("ProdName") & _
"</TD>" & _
"<TD ALIGN=CENTER WIDTH=100><INPUT TYPE=CheckBox NAME=ProdList " & _
"VALUE="& objRS("PID") & _
" If objRS("SubCatID")= intSubCatID Then
' If the product already on the list mark the CheckBox
strMessage = strMessage & " Checked"
End If
strMessage = strMessage & "</TD></TR>"
objRS.MoveNext
WEND

' Close the recordset
objRS.Close
Set objRS=Nothing

' Write information
strMessage = strMessage & "</TABLE>"
Response.Write strMessage

</TABLE BORDER=0 WIDTH=400>
</INPUT TYPE=HIDDEN NAME=SubCatID VALUE="% intSubCatID%»
</INPUT TYPE=HIDDEN NAME=CatID VALUE="% intCatID%»
</TR><TD><INPUT TYPE=SUBMIT NAME=btnAdd VALUE="Modify Subcategory"></TD>
</TD><INPUT TYPE=SUBMIT NAME=btnBack VALUE="<< Back"></TD>
</TR>
</TABLE></FORM>
SubCatUpdate.asp

<%@ Language=VBScript %>
<%Response.Buffer = True
1******************************************************************************
' SubCatUpdate.asp
' Program by Sumuscha Teesri
' May 20, 2000
1******************************************************************************
%

<! Include file needed >
<!-- INCLUDE FILE="Head.asp" -->
<!-- INCLUDE FILE="NavigatorBar.asp" -->
<SCRIPT Language="JavaScript">

function VerifyData()
{
    if (document.frmSubCatUpdate.SubCatName.value=="")
    {
        alert("Please provide the subcategory's name");
        return false;
    }
    else
    {
        alert("The record was updated");
        return true;
    }
}

</SCRIPT>
<!--******************************************************************************-->  
<!--*  Begin main  -->
<!--******************************************************************************-->
<%
If Session("EmpID") = "" Then
    Response.Redirect "EmpLogIn.asp"
End If

Set objRS=Server.CreateObject("ADODB.Recordset")
strSQL = "SELECT * FROM SubCategories WHERE SubCatID=" &
CInt(Request("SubCatID"))
objRS.Open strSQL, objConn, adOpenForwardOnly, adLockReadOnly, objCmdText
%
<CENTER><H1><FONT COLOR="#22FF22">Modify
Subcategory</FONT></H1></CENTER>
<!--This is template file please delete and enter code here-->  
<FORM ACTION="AddSubCat.asp?Update=True" METHOD=POST
Modify Category

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SubCatID</td>
<td></td>
</tr>
<tr>
<td>CatID</td>
<td></td>
</tr>
<tr>
<td>SubCatName</td>
<td>Description</td>
</tr>
</tbody>
</table>

```
function DeleteSup() {
    if (confirm("Are you sure you want to delete?")) {
        alert("Delete per your request");
        return true;
    } else {
        alert("Delete cancelled");
        return false;
    }
}
```

Make sure that only employee can access this web site

282
If Session("EmpID") = "" Then
  Response.Redirect "EmpLogIn.asp"
End If

If Len(Request("btnBack")) Then
  ' Back to supplier page
  Response.Clear
  Response.Redirect "Supplier.asp"
End If

' Connect to Suppliers database
Set objRS = Server.CreateObject("ADODB.Recordset")
strSQL = "SELECT * FROM Suppliers WHERE SupID=" & CInt(Request("SupID")) & ";"

' The following code is for deleting
' a supplier record only used when
' delete cascade trigger is implemented
'*******************************************************************************
If Len(Request("btnDelete")) Then
  ' Delete this Supplier
  objRS.Open strSQL, objConn, adOpenForwardOnly, adLockOptimistic, adCmdText
  'objRS.Delete
  'objRS.Close
  ' Response.Clear
  Response.Redirect "Supplier.asp"
' End If
'*******************************************************************************

' Check wheather update suppliers table is make
If Len(Request("btnUpdate")) Then
  ' Update data
  objRS.Open strSQL, objConn, adOpenForwardOnly, adLockOptimistic, adCmdText
  objRS("CompanyName") = Request("CompanyName")
  objRS("Address1") = Request("Address1")
  objRS("Address2") = Request("Address2")
  objRS("City") = Request("City")
  objRS("Region") = Request("Region")
  objRS("PostalCode") = Request("PostalCode")
  objRS("Country") = Request("Country")
  objRS("Phone") = Request("Phone")
  objRS("Fax") = Request("Fax")
  objRS("WebSite") = Request("WebSite")
  objRS("Title") = Request("Title")
  objRS("LastName") = Request("LastName")
  objRS("FirstName") = Request("FirstName")
  objRS("Direct") = Request("Direct")
  objRS("Email") = Request("Email")
  objRS.Update
  objRS.Close
End IF

' Connect to Supplier database
objRS.Open strSQL, objConn, adOpenForwardOnly, adLockReadOnly, adCmdText

' For debug only
'Dim strCompanyName
'strCompanyName = Replace(objRS("CompanyName"), chr(32), "&nbsp;")
strCompanyName = objRS("CompanyName")
Response.Write strCompanyName & "<BR>"

' Write web detail to client's browser

%>
<CENTER><H1><FONT COLOR=#22FF22>Supplier Detail</FONT></H1></CENTER>
<form method=POST name=frmSupplier>
<table border=0 width=500>
<tr><td width=150><b>Supplier ID:</b></td><td>%=objRS("SupID")%</td></tr>
<tr><td colspan=2 align=center><a href="Product.asp?SupId=%=objRS("SupID")%">Product List</a></td></tr>
<tr><td width=150><b>Company Name:</b></td><td>
<input type=text name=CompanyName size=50 value="%=objRS("CompanyName")%"
<tr>
<tr><td width=150><b>Address:</b></td><td colspan=3>
<input type=text name=Address1 size=50 value="%=objRS("Address1")%"
<input type=text name=Address2 size=50 value="%=objRS("Address2")%"
<br>
<tr><td width=150><b>City:</b></td><td colspan=3>
<input type=text name=City size=50 value="%=objRS("City")%"
<input type=text name=Region size=50 value="%=objRS("Region")%"
<br>
<tr><td width=150><b>Postal Code:</b></td><td colspan=3>
<input type=text name=PostalCode size=50 value="%=objRS("PostalCode")%"
<br>
<tr><td width=150><b>Phone:</b></td><td colspan=3>
<input type=text name=Phone size=50 value="%=objRS("Phone")%"
<br>
<tr><td width=150><b>Fax:</b></td><td colspan=3>
<input type=text name=Fax size=50 value="%=objRS("Fax")%"
<br>
<tr><td width=150><b>Website:</b></td><td colspan=3>
<input type=text name=WebSite size=50 value="%=objRS("WebSite")%"
<br>
<tr><td width=150><b>Contactor:</b></td><td colspan=3>
<input type=text name=Title size=50 value="%=objRS("Title")%"
<br>
</form>
' Make sure that only employee login
If Session("EmpID") = "" Then
    Response.Redirect "EmpLogIn.asp"
End If

' Write head to client's browser
Dim strMessage
strMessage = "<CENTER><H1><FONT COLOR=#22FF22>Supplier</FONT></H1></CENTER>"

Set objRS = server.CreateObject("ADODB.Recordset")
' Select only information needed
strSQL = "SELECT SupID, CompanyName, Phone, Fax, Email " & _
    "FROM Suppliers "
If Len(Request("SearchMethod")) Then
    'Response.Write Request("SearchMethod")
    Select Case Request.Form("SearchMethod")
    Case "ID"
        strSQL = strSQL & "WHERE SupID Like '" & Request("Keyword") & ";"
    Case "Email"
        strSQL = strSQL & "WHERE Email Like '" & Request("Keyword") & ";"
    Case "CompanyName"
        strSQL = strSQL & "WHERE CompanyName Like '" & Request("Keyword") & ";"
    End Select
Else
    strSQL = strSQL & ";"
End If
' Connect to database

objRS.Open strSQL, objConn, adOpenStatic, adLockReadOnly
objRS.PageSize = 10

If Not objRS.EOF Then
' Set Page Number
    Dim PageMove
    PageMove = Request("PageMove")
    If Request("PageNO") = "" Then
        PageNo = 1
    Else
        PageNo = Request("PageNO")
    End If

    ' Set Navigation button
    If PageMove = "Previous" or PageMove = "Next" Then
        Select Case PageMove
        Case "Previous"
            If PageNo > 1 Then
                PageNo = PageNo - 1
            Else
                PageNo = 1
            End If
        Case "Next"
            PageNo = PageNo + 1
    End If
Case "Next"
    If objRS.AbsolutePage < objRS.PageCount
        PageNo = PageNo + 1
    Else
        PageNo = objRS.PageCount
    End If
End Select

objRS.AbsolutePage = PageNo
' Write detail to client's browser
strMessage = strMessage & "<TABLE BORDER=0 WIDTH= 550><TR><TD WIDTH=450>" & _
"<A HREF=AddSup.asp><B>Add New Supplier</B></A><TD>
&nbsp;&nbsp;" & _
"<TD ALIGN=RIGHT WIDTH=100><SMALL>Page &nbsp;&nbsp; PageNo & * of " & objRS.PageCount & _
"<</SMALL><TD><TR><TABLE>" & _
"<TABLE BORDER=1 WIDTH=625><TR>" & _
"<TD WIDTH=150 ALIGN=RIGHT><B>ID</B></TD>
& _
"<TD WIDTH=30 ALIGN=CENTER><B>Company Name</B></TD>
& _
"<TD WIDTH=200 ALIGN=CENTER><B>Phone</B></TD>" & _
"<TD WIDTH=150 ALIGN=CENTER><B>Fax</B></TD>" & _
"<TD WIDTH=80 ALIGN=CENTER><B>Email</B></TD>" & _
"<TD WIDTH=150 ALIGN=CENTER><B>Email</B></TD><TR>"
    Dim intItemCount
    For intItemCount = 1 to objRS.PageSize
        ' Write table body
        strMessage = strMessage & "<TR><TD WIDTH=30 ALIGN= CENTER><SMALL>" & _
objRS("SupID") & "">" & _
objRS("SupID") & "</A><SMALL></TD>" & _
"<TD WIDTH=200><SMALL>" & _
"<A HREF=SupDetail.asp?SupID=" & _
objRS("SupID") & "</A><!--<A>HREF=SupDetail.asp?SupID=" & _
objRS("CompanyName") & "</A></TD>" & _
"<TD WIDTH=150><SMALL>" & objRS("Phone") & _
"<TD WIDTH=150><SMALL>" & objRS("Fax") & _
"<TD WIDTH=80><SMALL>" & _
"<A HREF=mailto:" & objRS("Email") & "">" & _
objRS("Email") & "</A></TD></TR>"
objRS.MoveNext
No more record
If objRS.EOF Then
  Exit For
End If

Next
strMessage = strMessage & "</TABLE>"

Response.Write strMessage

strMessage = "PageNo=" & PageNo
If PageNo > 1 or PageNo < objRS.PageCount Then
  Response.Write "<TABLE BORDER=0><TR><TD WIDTH=100>&nbsp;</TD><TD WIDTH=100 HEIGHT=30>"
  window.status='Previous Record';return true;
  <img border="0" src="image/Back.gif" width="100" height="30"
  id="fpAnimswapImgPre" name="fpAnimswapImgPre" dynamicanimation="fpAnimswapImgPre"
  lowsrc="image/BackOV.gif">
</A>
%A
End If
If PageNo > 1 Then
  'Write Animation button for the Previous record navigator
%A
<A HREF="Supplier.asp?c%=&PageMove=Previous" onmouseover="document['fpAnimswapImgPre'].imgRolln=document['fpAnimswapImgPre'].src;document['fpAnimswapImgPre'].src=document['fpAnimswapImgPre'].lowsrc;window.status='Previous Record';return true;
  onmouseout="document['fpAnimswapImgPre'].src=document['fpAnimswapImgPre'].imgRolln;
  window.status='';return true">
  <img border="0" src="image/Back.gif" width="100" height="30"
  id="fpAnimswapImgPre" name="fpAnimswapImgPre" dynamicanimation="fpAnimswapImgPre"
  lowsrc="image/BackOV.gif">
</A>

%A
End IF
If PageNo < objRS.PageCount Then
  'Write Animation button of Next record navigator
%A
<A HREF="Supplier.asp?c%=&PageMove=Next" onmouseover="document['fpAnimswapImgNext'].imgRolln=document['fpAnimswapImgNext'].src;document['fpAnimswapImgNext'].src=document['fpAnimswapImgNext'].lowsrc;
  window.status='Next Record';return true;
  onmouseout="document['fpAnimswapImgNext'].src=document['fpAnimswapImgNext'].imgRolln;
  window.status='';return true">
  <img border="0" src="image/Next.gif" width="100" height="30"
  id="fpAnimswapImgNext" name="fpAnimswapImgNext" dynamicanimation="fpAnimswapImgNext"
  lowsrc="image/NextOV.gif">
</A>

%A
End If
Response.Write "</TD><TD WIDTH=100
HEIGHT=30></TD></TR></TABLE></TD></TR></TABLE>

Else
' Find not found
Response.Write "<BR><BR><CENTER>" & _
"<P>Sorry We are currently do not have any
supplier you are looking for." & _
"<BR>Please try again.<BR></CENTER><BR><BR>
End If
objRS.Close
Set objRS = Nothing
%
</FORM METHOD=POST>
<B>Search Supplier Detail</B><BR>
Please select one of the following searching methods:<BR>
.INPUT TYPE=RADIO NAME="SearchMethod" VALUE="ID" CHECKED>Supplier ID
&nbsp; &nbsp;
.INPUT TYPE=RADIO NAME="SearchMethod" VALUE="CompanyName">Company Name
&nbsp; &nbsp;
.INPUT TYPE=RADIO NAME="SearchMethod" VALUE="Email">Email &nbsp;
Search Keyword: &nbsp;
.INPUT TYPE=TEXT NAME="Keyword">
</FORM>
</TABLE WIDTH=500></TR><TD VALIGN=TOP></B>WildCard:</B></TD>
<TD><B>Use</B> _</B>
(underscore) for one character or &nbsp; &nbsp;
for many characters</TD></TR>
</TABLE WIDTH=500></TR><TD VALIGN=TOP></B>Example :</B></TD>
<TD>&nbsp;&nbsp;</TD>
<TD>&nbsp;&nbsp;</TD><B>%Computer% for every company with the word
Computer</B></TD></TR>
</TABLE
</TABLE>

<!-- #INCLUDE FILE="Footer.asp" -->

<!**********************************>
Template.asp

<%@ Language=VBScript %>
<!******************************
* ProgramName.asp
* Program by Sumuscha Teesri
* May 11, 2000
******************************>

<! Include file needed >
<!-- #INCLUDE FILE="Head.asp" -->
<!-- #INCLUDE FILE="NavigatorBar.asp" -->
<!--**********************************-->
<!--* Begin main -->
<!-------------
$ Response.Buffer = True
If Session("EmpID") = "" Then
   Response.Redirect "EmpLogIn.asp"
End If
$>
<CENTER><H1><FONT COLOR="#22FF22">Title Here</FONT></H1></CENTER>
<!--This is template file please delete and enter code here-->  
<!-- #INCLUDE FILE="Footer.asp" -->
<!----------------------------------------------------------------------

**ViewMessage.asp**

<%@ Language=VBScript %>
<!-------------------------------
  * ProgramName.asp
  * Program by Sumuscha Teesri
  * May 11, 2000
------------------------------->
<! Include file needed >
<!-- #INCLUDE FILE="Head.asp" -->
<!-- #INCLUDE FILE="NavigatorBar.asp" -->

<!-------------------------------
  * Begin main -->
<!-------------
<%
  ' Make sure that only employee is allow to access the site
  Response.Buffer = True
  If Session("EmpID") = "" Then
     Response.Redirect "EmpLogIn.asp"
  End If

  ' Declaration statement
  Dim colMsgs
  Dim objCurMessage
  Dim objSession
  Dim intIndex

  ' Connect to mail service
  Set objSession = Session("CurSession")
  Set colMsgs = objSession.Inbox.Messages
  intIndex = Request.QueryString("MsgID")
%>
<CENTER><H1><FONT COLOR="#22FF22">View Message</FONT></H1></CENTER>
<P>
<TABLE BORDER=0 CELLPADDING=1 CELLSspacing=1 WIDTH="75%">
<TR>
    <TD><STRONG>From:</STRONG></TD>
</TR>


290
<TR>
  <TD><LABEL><%= colMsgs(intIndex).Sender %></LABEL></TD>
</TR>

<TR>
  <TD><STRONG>Subject:</STRONG></TD>
  <TD><LABEL><%= colMsgs(intIndex).Subject %></LABEL></TD>
</TR>

<TR>
  <TD><TEXTAREA ID=TEXTAREA1 NAME=TEXTAREA1 STYLE="HEIGHT: 200PX; WIDTH:500PX">
  <%= colMsgs(intIndex).Text %>
  </TEXTAREA></TD>
</TR>

</TABLE></P>
</FORM>
</TABLE>
</BODY>
</HTML>
APPENDIX B. DATABASE SOURCE CODE

Step.txt

/******************************************************************************
/* Stepts to Create MacrotoysSQL Database */
/* Run the following files by this step to create */
/* Macrotoys's Database */
******************************************************************************
/* By Sumuscha Teesri      Date April 28, 2000 */
/* Project: B2BCEA */
******************************************************************************

1 CreateDatabase.sql
2 AddUser.sql
3 CreateEMPLOYEES.sql
4 CreateSUPPLIERS.sql
5 CreateCategories.sql
6 CreateSubcategories.sql
7 CreateCustomerTypes.sql
8 CreateStateZone.sql
9 CreateStates.sql
10 CreateCarriers.sql
11 CreateWeightZone.sql
12 createShippingRates.sql
13 CreateSizeZone.sql
14 CreateCUSTOMERS.sql
15 CreatePRODUCTS.sql
16 CreateProductCats.sql
17 CreateOrderStatus.sql
18 CreatePaymentTypes.sql
19 CreateORDERS.sql
20 CreateTRANSACTS.sql
21 createWebStats.sql

292
**AddUser.sql**

/* Add an Internet user to MacrotoysSQL Database */
/* Then assign the necessary permit to the user */
/* - IUSR is the default user to connect to the MacrotoysSQL Database, change the name if necessary */
/* By Sumuscha Teesri Date April 26, 2000 */
/* Project: B2BECA */
/* Database: MacrotoysSQL Table: */

USE MacrotoysSQL
GO

/* Add user to the database */
EXEC sp_addlogin @loginame='IUSR', @passwd='', @defdb='MacrotoysSQL'
GO

/* Assign read and write permit */
EXEC sp_grantdbaccess @loginame='IUSR'
GO
EXEC sp_addrolemember @rolename = 'db_datareader', @membername = 'IUSR'
GO
EXEC sp_addrolemember @rolename = 'db_datawriter', @membername = 'IUSR'
GO

**CreateCarriers.sql**

/* Create Carriers Table in MacrotoysSQL Database */
/* Carriers is a system database. */
/* Primary key: CarrierID Foreign key: */
/* */

USE MacrotoysSQL
CREATE TABLE dbo.Carriers
(
    CarrierID int IDENTITY(1,1) NOT NULL
    PRIMARY KEY,
    Description varchar(50) NOT NULL
)
GO
**CreateCategories.sql**

```sql
USE MacrotoysSQL
CREATE TABLE dbo.Categories
(
  CatID int IDENTITY(1,1) NOT NULL
  PRIMARY KEY,
  CatName varchar(20) NOT NULL
  UNIQUE,
  Description varchar(50) NULL
)
GO
```

**CreateCUSTOMERS.sql**

```sql
USE MacrotoysSQL
CREATE TABLE dbo.CUSTOMERS
(
  CusID int IDENTITY(1,1) NOT NULL
  PRIMARY KEY,
  EmpID int NOT NULL
  FOREIGN KEY REFERENCES Employees(EmpID),
  CompanyName varchar(40) NOT NULL
  UNIQUE,
  Address1 varchar(30) NOT NULL,
  Address2 varchar(30) NULL,
  City varchar(20) NOT NULL,
  Region varchar(20) NULL,
  PostalCode varchar(10) NULL,
  Country varchar(20) NOT NULL,
  Phone varchar(15) NOT NULL,
  Fax varchar(15) NULL,
  LogInName varchar(15) NOT NULL
)
CREATE DATABASE.sql

/******************************************************************************
* Create MacrotoysSQL Database
* - MacrotoysSQL is the major database used in B2BECA
* - Change the path if necessary
* By Sumuscha Teesri     Date April 26, 2000
* Project: B2BECA
* Database: MacrotoysSQL Table:  
/******************************************************************************

CREATE DATABASE MacrotoysSQL

---

CreateCustomerTypes.sql

/******************************************************************************
* Create CustomerTypes Table in MacrotoysSQL Database
* - CustomerTypes is a system database.
* Primary key: ID    Foreign key:  
* By Sumuscha Teesri     Date April 26, 2000
* Project: B2BECA
* Database: MacrotoysSQL Table: CustomerTypes  
******************************************************************************

USE MacrotoysSQL
CREATE TABLE dbo.CustomerTypes
(
    ID varchar(2) NOT NULL,
    PRIMARY KEY,
    Description varchar(50) NOT NULL,
    Discount real NOT NULL
)
GO
CREATE DATA FILE
ON
(NAME = 'MacrotoysSQL',
FILENAME = 'D:\MSSQL7\Data\MacrotoysSQL_Data.mdf',
SIZE = 5MB,
MAXSIZE = 50MB,
FILEGROWTH = 5MB)

CREATE LOG FILE
LOG ON
(NAME = 'MacrotoysSQLLog',
FILENAME = 'D:\MSSQL7\Data\MacrotoysSQL_log.idf',
SIZE = 5MB,
MAXSIZE = 25MB,
FILEGROWTH = 5 MB)

GO

CREATE EMPLOYEES.sql

USE MacrotoysSQL
CREATE TABLE dbo.EMPLOYEES
(
    EmpID int IDENTITY(1,1) NOT NULL
    PRIMARY KEY,
    SSN varchar(11) NOT NULL
    UNIQUE,
    FirstName varchar(30) NOT NULL,
    LastName varchar(30) NOT NULL,
    Address1 varchar(30) NULL,
    Address2 varchar(30) NULL,
    City varchar(20) NOT NULL,
    State varchar(2) NOT NULL,
    PostalCode varchar(10) NOT NULL,
    Phone varchar(15) NOT NULL,
    Email varchar(35) NOT NULL
    UNIQUE,
    Password varchar(15) NOT NULL,
    HireDate smalldatetime NOT NULL
    DEFAULT GETDATE() )

GO
**CreateORDERS.sql**

```sql
USE MacrotoysSQL
CREATE TABLE dbo.ORDERS
(
    OrderID int IDENTITY(1,1) NOT NULL,
    PRIMARY KEY,
    CusID int NOT NULL,
    FOREIGN KEY REFERENCES CUSTOMERS(CusID),
    OrderDate smalldatetime NOT NULL DEFAULT GETDATE(),
    Status varchar(2) NOT NULL REFERENCES ORDERSTATUS(ID),
    Payment varchar(2) NOT NULL REFERENCES PaymentTypes(ID),
    AccNum varchar(16) NULL,
    CarrierID int NOT NULL REFERENCES Carriers(CarrierID),
    Tracking Varchar(20) NULL
)
GO
```

**CreateOrderStatus.sql**

```sql
USE MacrotoysSQL
CREATE TABLE dbo.OrderStatus
(
    ID varchar(2) NOT NULL PRIMARY KEY,
    Description varchar(30) NOT NULL
)
GO
```
CreatePaymentTypes.sql

 USE MacrotoysSQL
 CREATE TABLE dbo.PaymentTypes
 ( 
   ID PRIMARY KEY, varchar(2) NOT NULL
   Description varchar(30) NOT NULL
 )
 GO

CreateProductCats.sql

 USE MacrotoysSQL
 CREATE TABLE dbo.ProductCats
 ( 
   ProdID int NOT NULL
   FOREIGN KEY REFERENCES PRODUCTS(ProdID),
   SubCatID int NOT NULL
   FOREIGN KEY REFERENCES Subcategories(SubCatID),
   StartDate smalldatetime NULL
   DEFAULT GETDATE(),
   EndDate smalldatetime NULL,
   Promotion money NULL
 )
 GO
Create PRODUCTS.sql

USE MacrotoysSQL
CREATE TABLE dbo.PRODUCTS
(
    ProdID int IDENTITY(1,1) NOT NULL,
    PRIMARY KEY,
    SupID int NOT NULL,
    FOREIGN KEY REFERENCES SUPPLIERS(SupID),
    ProdName varchar(50) NOT NULL,
    Description text NULL,
    PurchasePrice money NOT NULL,
    CHECK (PurchasePrice > 0),
    SalePrice money NOT NULL,
    CHECK (SalePrice > 0),
    StockQuantity int NOT NULL,
    CHECK (StockQuantity > 0),
    ReorderPoint int NOT NULL,
    CHECK (ReorderPoint > 0),
    SaleUnit varchar(20) NULL,
    PurchaseUnit varchar(20) NULL,
    Age varchar(20) NULL,
    Gender varchar(1) NULL,
    CHECK (Gender in ( 'M', 'F', 'U' ) ),
    Picture varchar(20) NULL,
    Measure int NOT NULL,
    WEIGHT real NOT NULL,
    FOREIGN KEY REFERENCES SizeZones(SizeZoneID),
    SizeZoneID int NOT NULL
)
GO
CreateShippingRates.sql

USE MacrotoysSQL
CREATE TABLE dbo.ShippingRates
(
    CarrierID int NOT NULL,
    FOREIGN KEY REFERENCES Carriers(CarrierID),
    WeightZoneID int NOT NULL,
    FOREIGN KEY REFERENCES WeightZones(WeightZoneID),
    ShippingCost money NOT NULL,
    CHECK (ShippingCost > 0),
    CONSTRAINT transact_key PRIMARY KEY (CarrierID, WeightZoneID)
)
GO

CreateSizeZone.sql

USE MacrotoysSQL
CREATE TABLE dbo.SizeZones
(
    SizeZoneID int IDENTITY(1,1) NOT NULL,
    PRIMARY KEY,
    Height real NOT NULL,
    Width real NOT NULL,
    Length real NOT NULL,
    ChargeRate real NOT NULL
)
GO
CreateStates.sql

USE MacrotoysSQL
CREATE TABLE dbo.States
(
    ID varchar(2) NOT NULL
    PRIMARY KEY,
    StateName varchar(30) NOT NULL,
    StateZoneID int NOT NULL
    REFERENCES StateZones(StateZoneID)
)
GO

CreateStateZones.sql

USE MacrotoysSQL
CREATE TABLE dbo.StateZones
(
    StateZoneID int IDENTITY(1,1) NOT NULL
    PRIMARY KEY,
    ChargeRate real NOT NULL
)
GO
**CreateSubcategories.sql**

```sql
USE MacrotoysSQL
CREATE TABLE dbo.Subcategories
(
    SubCatID int IDENTITY(1,1) NOT NULL,
    PRIMARY KEY,
    CatID int NOT NULL,
    FOREIGN KEY REFERENCES Categories(CatID),
    SubCatName varchar(20) NOT NULL,
    Description varchar(50) NULL
)
GO
```

**CreateSUPPLIERS.sql**

```sql
USE MacrotoysSQL
CREATE TABLE dbo.SUPPLIERS
(
    SupID int IDENTITY(1,1) NOT NULL,
    PRIMARY KEY,
    CompanyName varchar(40) NOT NULL,
    UNIQUE,
    Address1 varchar(30) NOT NULL,
    Address2 varchar(30) NULL,
    City varchar(20) NOT NULL,
    Region varchar(20) NULL,
    PostalCode varchar(10) NULL,
    Country varchar(20) NOT NULL,
    Phone varchar(15) NOT NULL,
    Fax varchar(15) NULL,
    Website varchar(40) NULL,
    Title varchar(35) NULL
)
```
CREATE TABLE dbo.TRANSACTS
(
    OrderID int NOT NULL,
    FOREIGN KEY REFERENCES ORDERS(OrderID),
    ProdID int NOT NULL,
    FOREIGN KEY REFERENCES PRODUCTS(ProdID),
    Quantity int NOT NULL,
    CHECK (Quantity > 0),
    CONSTRAINT transact_key PRIMARY KEY (OrderID, ProdID)
)
GO

CREATE TABLE dbo.WebStats
(
    WebStatID int IDENTITY(1,1) NOT NULL,
    PRIMARY KEY,
    Browser varchar(20) NULL,
    Version varchar(20) NULL,
)
GO
CREATE TABLE dbo.WeightZones
(
    WeightZoneID int IDENTITY(1,1) NOT NULL,
    PRIMARY KEY,
    MinWeight real NOT NULL,
    MaxWeight real NOT NULL
)
REFERENCES

[1] Baltazar, Henry and Chowdhry Pankaj: PC Week Labs's tests show what path Linux must take, eWeek, June 29,1999


