Get the right price every day

Gerardo Javier Garcia Sotelo

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

Part of the Software Engineering Commons

Recommended Citation
Garcia Sotelo, Gerardo Javier, "Get the right price every day" (2005). Theses Digitization Project. 2729. https://scholarworks.lib.csusb.edu/etd-project/2729

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.
GET THE RIGHT PRICE EVERY DAY

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

In Partial Fulfillment
of the Requirements for the Degree
Master of Science
in
Computer Science

by
Gerardo Javier Garcia Sotelo
December 2005
GET THE RIGHT PRICE EVERY DAY

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

by
Gerardo Javier Garcia Sotelo

December 2005

Approved by:

Dr. Kerstin Voigt, Chair, Computer Science

Dr. Richard Botting

Dr. David Turner

11/29/05
Date
ABSTRACT

Do you know that the restaurant industry has the highest rates of failure in the US economy? From 100 new restaurants to open this year, 50 to 80 will fail on the first year and just 10 of them will be able to celebrate a third anniversary. So, in other words, just 10% of new restaurants will succeed in the restaurant industry. This situation is due to the fact that many of these restaurants neglect the financial side of the business. Controlling restaurant cost is as important as serving good quality food. To keep food cost controlled, it is very important to have consistency in portion control. To achieve this, you will need an efficient tool.

GRIPED which stands for "Get the Right Price Every day" will help restaurants to achieve this goal.

GRIPED has been implemented in PHP, MySQL database, and Apache Web Server on Windows System and provides cross-platform and extensibility for any future development.
ACKNOWLEDGMENTS

GRIPED was developed taking and considering ideas from people close to me.

I wish to acknowledge the help from my parents and wife which basically helped me out to come up with the idea to price the products out. In addition, I wish to acknowledge SYSCO and US Food Services vendors that helped me out numerous times with product information. I also wish to acknowledge the contribution of my advisor Dr. Kerstin Voigt and committee members Dr. Richard Botting and Dr. David Turner from California State University of San Bernardino, Computer Science Department, who reviewed this report and whose suggestions and commendations have been incorporated.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT</td>
<td>iii</td>
</tr>
<tr>
<td>ACKNOWLEDGMENTS</td>
<td>iv</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>viii</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>x</td>
</tr>
<tr>
<td><strong>CHAPTER ONE: SOFTWARE REQUIREMENTS SPECIFICATION</strong></td>
<td></td>
</tr>
<tr>
<td>1.1 Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Product Overview Summary</td>
<td>3</td>
</tr>
<tr>
<td>1.2.1 Perspective of Administrator</td>
<td>4</td>
</tr>
<tr>
<td>1.2.2 Perspective of User</td>
<td>5</td>
</tr>
<tr>
<td>1.3 Technologies</td>
<td>5</td>
</tr>
<tr>
<td>1.3.1 Hypertext Preprocessor</td>
<td>5</td>
</tr>
<tr>
<td>1.3.2 Database Management System</td>
<td>6</td>
</tr>
<tr>
<td>1.3.3 Apache</td>
<td>6</td>
</tr>
<tr>
<td>1.4 Development Operating and Maintenance Environment</td>
<td>7</td>
</tr>
<tr>
<td>1.4.1 Development Environment</td>
<td>7</td>
</tr>
<tr>
<td>1.4.2 Operating Environment</td>
<td>8</td>
</tr>
<tr>
<td>1.4.3 Maintenance Environment</td>
<td>8</td>
</tr>
<tr>
<td>1.4.4 Graphical User Interface</td>
<td>8</td>
</tr>
<tr>
<td>1.5 Assumptions</td>
<td>8</td>
</tr>
<tr>
<td>1.6 Definition of Terms</td>
<td>9</td>
</tr>
<tr>
<td>1.7 Organization of the Thesis</td>
<td>9</td>
</tr>
<tr>
<td><strong>CHAPTER TWO: SOFTWARE DESIGN</strong></td>
<td></td>
</tr>
<tr>
<td>2.1 Introduction</td>
<td>11</td>
</tr>
</tbody>
</table>
2.2 Overall Schema .................................................. 11
2.3 Function Schema ............................................... 12
2.4 Database Design ............................................... 13
2.5 Logical Model Table Schema .............................. 14
2.5 Performance Requirements .............................. 16
   2.5.1 Reliability .............................................. 16
   2.5.2 Efficiency .............................................. 16
   2.5.3 Testability .............................................. 16
   2.5.4 Exception Handling ................................. 16
2.6 Summary ......................................................... 17

CHAPTER THREE: SOFTWARE QUALITY ASSURANCE

3.1 Introduction .................................................. 18
3.2 Unit Test Plan ............................................... 18
3.3 Integration Test Plan ...................................... 18
3.4 Test Cases ..................................................... 19

CHAPTER FOUR: USERS MANUAL

4.1 Introduction .................................................. 32
4.2 General Layout .............................................. 32
4.3 Home Link Information .................................. 34
4.4 New Item Link Information .............................. 34
4.5 View Items Option ........................................ 37
4.6 Add Provider Option ........................................ 39
4.7 View Provider Option ...................................... 41
4.8 New User Option ............................................ 42
4.9 View Users Option on Administrator Menu ........ 44
4.10 Search Option on Administrator Menu ........... 45
4.11 Logout Option on Administrator Menu ........... 54
4.12 User’s Menu ........................................ 54
4.13 New Product Option on User’s Menu .......... 55
4.14 New Combo Option in User’s Menu ............ 56
4.15 Products/Combo Option on User’s Menu ...... 57
4.16 Specials Option on User’s Menu ............... 58
4.17 Logout Option on User’s Menu ................. 59

CHAPTER FIVE: CONCLUSIONS

5.1 Conclusions ............................................ 60
5.2 Proposed Future Development ..................... 61

APPENDIX: PROGRAM .................................... 63

REFERENCES .............................................. 106
LIST OF TABLES

Table 1. Item Table .......................................................... 14
Table 2. Category Table ................................................... 14
Table 3. Provider Table .................................................... 15
Table 4. Product Table ..................................................... 15
Table 5. Combo Table ...................................................... 15
Table 6. User Table ........................................................ 16
Table 7. Test Case for Login Page ................................. 19
Table 8. Test Case for Administrator’s Menu ................. 20
Table 9. Test Case for New Item Option on Administrator Menu .................................................. 21
Table 10. Test Case for View Item Option on Administrator Menu .............................................. 22
Table 11. Test Case for Add Provider Option on Administrator Menu ........................................... 23
Table 12. Test Case for View Provider Option on Administrator Menu ......................................... 24
Table 13. Test Case for New User Option on Administrator Menu ............................................... 25
Table 14. Test Case for View User Option on Administrator Menu .............................................. 26
Table 15. Test Case for Search Option on Administrator Menu .................................................... 27
Table 16. Test Case for Logout Option on Administrator Menu ................................................... 28
Table 17. Test Case for Login Page ................................. 28
Table 18. Test Case for Home Link ................................. 29
Table 19. Test Case for New Product Link ..................... 29

viii
Table 20. Test Case for New Combo Link ................. 30
Table 21. Test Case for Product/Combos Link .......... 30
Table 22. Test Case for Specials Link ................. 31
Table 23. Administrator Inputs and Corresponding
          Actions ...................................................... 34
Table 24. New Item Link on Administrator Menu ....... 36
Table 25. Add Provider link on Administrator
          Menu .......................................................... 40
Table 26. New User Link on Administrator Menu ...... 43
Table 27. Search Link on Administrator Menu ........... 46
Table 28. User's Menu Links and Corresponding
          Actions ...................................................... 55
LIST OF FIGURES

Figure 1. Overall Schema ................................................. 11
Figure 2. Functions Schema .............................................. 12
Figure 3. Entity Relation Diagram ................................. 13
Figure 4. Login Page ...................................................... 32
Figure 5. Administrator’s Menu ................................. 33
Figure 6. Home Link Information .................................. 34
Figure 7. New Item ...................................................... 35
Figure 8. Products Displayed After You Save a New Item ........................................ 37
Figure 9. View Items Display .......................................... 38
Figure 10. Item Edit Form .............................................. 39
Figure 11. Add a New Provider ........................................ 40
Figure 12. View Providers Information .......................... 41
Figure 13. Edit or Update a Provider ............................. 42
Figure 14. Create a New User ......................................... 43
Figure 15. View Existing Users ........................................ 44
Figure 16. Edit an Existing User ..................................... 45
Figure 17. Search an Item Using the Item Name ............... 46
Figure 18. Search Results Using the Item Name ................ 47
Figure 19. Search an Item Using its Item Number ............. 48
Figure 20. Search Results Using the Item Number .......... 49
Figure 21. Search Using Invoice Number ......................... 50
Figure 22. Search Using the Invoice Number .................. 51
Figure 23. Search an Item Using Provider’s Name .......... 52
Figure 24. Search Results Using the Provider’s Name ............................................................ 53

Figure 25. Display Message when Name, Number or Provider Can Not Be Found On The System ............................................................... 54

Figure 26. User’s Menu ........................................................................................................ 55

Figure 27. New Product ........................................................................................................ 56

Figure 28. New Combo Link ............................................................................................... 57

Figure 29. List of Products/Combos ................................................................................. 58

Figure 30. Items and Products on Special ........................................................................... 59
CHAPTER ONE
SOFTWARE REQUIREMENTS SPECIFICATION

1.1 Introduction

How many times have you bought an expensive meal that seems to cost a quarter of the price to be made? Or, have you bought one of those really cheap meals that seems like the company is not making any profit out of it? I am sure you have been in one of these situations at least once. To illustrate this condition, I will mention some examples of two items that you may find in a cafeteria. A salad that costs $4.50 USD includes: plastic container, portion of green salad and package of dressing. A hamburger combo that costs $2.99 USD includes hamburger bread, beef patty, 2 slices of tomato, 2 slices of onion, green lettuce, 1 pickle, 0.5 oz of mayonnaise or sauce, to-go box, wrapping tissue paper, condiments, soda, cup, lid, straw, paper bag for fries and 7 oz of French fries. Obviously, these final prices were not the result of a good food cost analysis.

The process of pricing out a product involves many factors such as season, price of gas and the percentage added due to the cost of operation of the place. Among food industry, setting up prices for sale is becoming a major problem. This is due to the price fluctuation among
brands, and different presentations of the same product from different providers. For example, lettuce costs $12.00 per case today and a week later you get it for $26.00. The tuna you special ordered at $6.00 a pound, comes in at $6.50. The foam cups are a petroleum-base product, and they mirror pump prices. So, how can restaurant management keep control and re-price the products every time this happens? As you can see, this makes it difficult for food industry staff to be setting up prices for items on the menu. That is why every time that new products are bought, prices need to be reviewed for consistency and we also need to verify that we are getting the best price on such products.

GRIPED will help Food industry staff to set a more accurate price of their products based on real prices at which the ingredients were bought every time.

GRIPED will include the following functionality:

• Access the most recent price on each product.
• Help owners to compare prices when a new vendor visits.
• Help staff to decide when is the right time to set a special price of a product and what can be included.
• Manage and control the inventory prices.
• Create new items for the menu based on the purchased items registered on the database.
• Allow the user to calculate and create prices of new items, saving them for future use as a single or combination.

GRIPED will benefit food industry staff as well as customers. Customers will have the possibility to purchase products just at the right price whereas the staff will have GRIPED as a powerful tool to keep track of their product prices. Also, GRIPED will allow the user to create the prices of new items, saving them in a section called “View Items”, where the list of all new existing items can be accessed at any time to check cost or make any update to them.

1.2 Product Overview Summary

GRIPED Software application is designed to store information about food industry products from several providers on a single database. This information about food industry products is found by the staff in charge (Admin) of add, update, revise and delete such information with the help of links provided by GRIPED.
The objective of this project is to create an application capable of storing information about food industry products from several providers in its own database; with the purpose of showing it later to the interested staff on create a new products, or simply keep track of the fluctuation of market prices.

Users can have access to GRIPED via Internet or locally by using a web browser like Internet Explorer. The interface used as a communication bridge between users and server is PHP, which helps to display the information on the site. To input information to the database from Internet, there are HTML forms in combination with PHP to perform this task like create a new item, edit an item, and create a new user.

The web server used for GRIPED is Apache, running on MS windows.

1.2.1 Perspective of Administrator

GRIPED formal Administrator will be able to add information regarding to food industry products, create and view users, search an item, view and add provider’s information and access existing items. Another option for the administrators is to update and delete information. These options are essential for database maintenance. The
administrator also has the option to create new items including the prices from the invoices.

1.2.2 Perspective of User

The available options for GRIPED users are: create a new product, create a new combo or view the existing products and combos.

All this information is located in one single main menu for the users. The links have the following functionality. New Product link on the user’s main menu will allow the user to create a new product using all the items previously entered by the administrator to the database. The user will have the option to save the new product. The link “new combo” on the user’s main menu will allow the user to create different combinations using the products created using the link “new product” on the user’s main menu. Another option that users will have is to see all the products and combinations that had been created. This feature is on the link products/combos also on user’s main menu.

1.3 Technologies

1.3.1 Hypertext Preprocessor

PHP has been chosen to develop GRIPED because can be used on all major operating systems, including Linux,
Microsoft Windows, Mac OS X and RISC OS, among others. PHP has also support for most of the web servers today. This includes Apache, Microsoft Internet Information Server, Netscape and iPlanet, and other servers. In addition, another reason to choose PHP to develop GRIPED is because PHP supports a wide range of databases including MySQL, which is free. Also PHP can interact well with HTML in web pages designing and has the advantage of connection to databases.

1.3.2 Database Management System

The Database Management system used to develop GRIPED was MySQL. MySQL was chosen because MySQL is a multithreaded, multi-user, SQL (Structured Query Language) Database Management System (DBMS). In addition, MySQL works on many different platforms—including Linux, Mac OS X, Net BSD, Netware, OS/2 Warp, Solaris, SunOS, SCO OpenServer, Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP and more recent versions of Windows. Also, MySQL is available as an open source software (free of charge).

1.3.3 Apache

Apache HTTP Server was chosen to develop GRIPED due that support common language interfaces such as Perl, Python, Tcl, and PHP. Apache supports a variety of
features, many implemented as compiled modules. These can range from server-side programming language support to authentication schemes. In addition, Apache features highly configurable error messages, DBMS-based authentication databases, and content negotiation. Another advantage to use Apache was the fact that is an open source HTTP web server for Unix-like systems (BSD, Linux, and UNIX systems), Microsoft Windows, Novell Netware and other platforms.

1.4 Development Operating and Maintenance Environment

1.4.1 Development Environment

The next hardware and software were used to develop GRIPED.

1. Hardware:
   a. Toshiba Satellite m35-s359 with the following specs:
   b. 60 GB HD
   c. 1.4 GHz Pentium M Processor
   d. 512 MB RAM
   e. 256 MB Memory stick
   f. CD / DVD Read Speed 24x (CD) / 8x (DVD)
   g. 19” external monitor.
   h. External keyboard and Mouse.
2. **Software:**
   b. Apache Web Server. 2.0.53
   c. MySQL 4.1.9
   d. Matrix Y2K 2005 HTML Editor.

3. **Languages:**
   a. PHP 5.0.3
   b. HTML

1.4.2 **Operating Environment**

GRIPED can be used from almost all web browsers like MS Internet Explorer, Mozilla Firefox, Safari, or Netscape.

1.4.3 **Maintenance Environment**

The maintenance of GRIPED should be performed using the elements mentioned in section 1.4.1 and 1.4.2

1.4.4 **Graphical User Interface**

The GRIPED GUI provide users with and easy way to locate products from many providers on a single database.

1.5 **Assumptions**

The following assumptions were made regarding the project:

1. User will be able to use the program locally or over the web.
2. User will provide necessary information to load GRIPED database.

1.6 Definition of Terms

The following terms are defined as they apply to the project:

PHP - Server-side language to access a database, and interacts with HTML.

Apache - Open source HTTP web server for Unix-like systems (BSD, Linux, and UNIX systems), Microsoft Windows, Novell Netware and other platforms.

MySQL - Open source database management system that allows interaction with apache server and php language. MySQL (Structured Query Language) support different platforms including Linux and Windows.

1.7 Organization of the Thesis

The thesis portion of the project was divided into five chapters. Chapter One provides software requirements specification, an introduction to the context of the problem, product overview summary, Technologies used on the project, development, operating and maintenance environment, assumptions and definitions of terms. Chapter Two consists of the software design. Chapter Three documents the steps used in testing the project. Chapter
Four presents the users manual from the project. Chapter Five presents conclusions drawn from the development of the project and proposed future development. The Appendices containing the project follows Chapter Five. Finally, the references for the project are presented.
CHAPTER TWO
SOFTWARE DESIGN

2.1 Introduction

GRIPED software was designed using PHP, HTML, MySQL and Apache server. These components allow the system full functionality and at the same time provide future extensibility under almost any platform.

2.2 Overall Schema

Components that were required by GRIPED to operate correctly (Figure 1).

![Overall Schema Diagram]

Figure 1. Overall Schema
2.3 Function Schema

Figure 2. Functions Schema
2.4 Database Design

Figure 3. Entity Relation Diagram
2.5 Logical Model Table Schema

The Tables used on GRIPED are shown on this section.

Table 1. Item Table

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Id</td>
<td>Item id generated by DB</td>
</tr>
<tr>
<td>item_name</td>
<td>Item name (text)</td>
</tr>
<tr>
<td>Pack</td>
<td>Item pack (alphanumeric)</td>
</tr>
<tr>
<td>Unit</td>
<td>Unit used (alphanumeric)</td>
</tr>
<tr>
<td>item_number</td>
<td>Item number (alphanumeric)</td>
</tr>
<tr>
<td>Invoice_number</td>
<td>Invoice number (alphanumeric)</td>
</tr>
<tr>
<td>*category</td>
<td>Select from drop down box</td>
</tr>
<tr>
<td>Subcategory</td>
<td>Enter subcategory (text)</td>
</tr>
<tr>
<td>*Provider</td>
<td>Select from drop down box</td>
</tr>
<tr>
<td>Comments</td>
<td>Enter comment (alphanumeric)</td>
</tr>
<tr>
<td>Price</td>
<td>Enter product price (number)</td>
</tr>
<tr>
<td>Special</td>
<td>Check Box</td>
</tr>
</tbody>
</table>

*These fields are linked to another table

Table 2. Category Table

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Select a category from drop down box.</td>
</tr>
</tbody>
</table>
Table 3. Provider Table

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Provider</td>
<td>Select a Provider from drop down box.</td>
</tr>
</tbody>
</table>

Table 4. Product Table

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pname</td>
<td>Enter product name (alphanumeric)</td>
</tr>
<tr>
<td>product_id</td>
<td>Enter product number (alphanumeric)</td>
</tr>
<tr>
<td>*item</td>
<td>List of Items on the DB are displayed</td>
</tr>
<tr>
<td>price</td>
<td>Price of the product is displayed.</td>
</tr>
</tbody>
</table>

*These field is linked to item table.

Table 5. Combo Table

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cname</td>
<td>Enter combo name (alphanumeric).</td>
</tr>
<tr>
<td>combo_id</td>
<td>Enter combo number (number).</td>
</tr>
<tr>
<td>*product</td>
<td>List of Items on the DB are displayed</td>
</tr>
<tr>
<td>price</td>
<td>Price of the product is displayed.</td>
</tr>
</tbody>
</table>

*These field is linked to product table.
Table 6. User Table

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Userid</td>
<td>Enter user id (alphanumeric).</td>
</tr>
<tr>
<td>Fname</td>
<td>Enter first name (text).</td>
</tr>
<tr>
<td>Lname</td>
<td>Enter last name (text)</td>
</tr>
<tr>
<td>Password</td>
<td>Enter password.</td>
</tr>
<tr>
<td>AdminCheck</td>
<td>Checkbox.</td>
</tr>
</tbody>
</table>

2.5 Performance Requirements

2.5.1 Reliability

Simple design, user-friendly interface and standard components make GRIPED reliable. To ensure the functionality of the system, all the available options on the system were tested.

2.5.2 Efficiency

The efficiency of GRIPED rely on the speed and accuracy of the task execution. This efficiency has been optimized using simple and efficient program on php that helps to operate the system fast and accurately.

2.5.3 Testability

Every option of the system was tested individually and as a part of the whole system. This ensures that every part of the system works correctly. (Refer to chapter 3.)

2.5.4 Exception Handling

Error messages, product of the system's validation are displayed on the browser when a step is missing or entered information is wrong. GRIPED alerts the user, so
that they can correct the information and keep using the system.

2.6 Summary

The software design of GRIPED was presented in Chapter Two. GRIPED was designed to save time and money to the owners of restaurants by keeping accurate information of items used on the operation of any food place. GRIPED features can be accessed from anywhere where there is an available Internet terminal. In addition, GRIPED features can also be used from a local terminal using an internet browser. By gathering all the product information from different food providers, GRIPED provides to the user a fast and efficient way to price a product out assuring the right and lowest price between different providers. To achieve this, different technologies were used on GRIPED development such as PHP, Mysql and Apache server.
CHAPTER THREE
SOFTWARE QUALITY ASSURANCE

3.1 Introduction
To assure the quality and full functionality of the software a testing cases were designed.

3.2 Unit Test Plan
The unit testing has focused on the verification of all the components of GRIPED. The unit tested will be showed on section 3.4.

3.3 Integration Test Plan
The integration test was done testing each one of the links connecting all the pages. The results are shown below on section 3.4.
3.4 Test Cases

Test cases are presented below.

Table 7. Test Case for Login Page

<table>
<thead>
<tr>
<th>Step No</th>
<th>Description</th>
<th>Used Data</th>
<th>Actual Results</th>
<th>Pass</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Enter user name and press Login</td>
<td>Username: gl</td>
<td>Access Denied</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>Enter password and press Login</td>
<td>Password: gll</td>
<td>Access Denied</td>
<td>Yes</td>
</tr>
<tr>
<td>3</td>
<td>Enter User name and password</td>
<td>Username: gl</td>
<td>Access to administrator menu</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Table 8. Test Case for Administrator’s Menu

<table>
<thead>
<tr>
<th>Step No</th>
<th>Description</th>
<th>Used Data</th>
<th>Actual Results</th>
<th>Pass</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Click on Home option</td>
<td>Click on link</td>
<td>Display first Admin Menu</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>Click on New Item</td>
<td>Click on link</td>
<td>Open a form to create a New Item</td>
<td>Yes</td>
</tr>
<tr>
<td>3</td>
<td>Click on View Item</td>
<td>Click on link</td>
<td>Display existing Items</td>
<td>Yes</td>
</tr>
<tr>
<td>4</td>
<td>Click on Add Provider</td>
<td>Click on link</td>
<td>Open a new form to add a New Provider.</td>
<td>Yes</td>
</tr>
<tr>
<td>5</td>
<td>Click on View Providers</td>
<td>Click on link</td>
<td>Display the providers on the System</td>
<td>Yes</td>
</tr>
<tr>
<td>6</td>
<td>Click on New User</td>
<td>Click on link</td>
<td>Open a Form to enter a new user.</td>
<td>Yes</td>
</tr>
<tr>
<td>7</td>
<td>Click on View Users</td>
<td>Click on link</td>
<td>Display a page with existing users.</td>
<td>Yes</td>
</tr>
<tr>
<td>8</td>
<td>Click on Search</td>
<td>Click on link</td>
<td>Display the search page.</td>
<td>Yes</td>
</tr>
<tr>
<td>9</td>
<td>Click on Logout</td>
<td>Click on link</td>
<td>End the current logging session.</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Table 9. Test Case for New Item Option on Administrator Menu

<table>
<thead>
<tr>
<th>Step No</th>
<th>Description</th>
<th>Used Data</th>
<th>Actual Results</th>
<th>Pass</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Enter item name</td>
<td>Paper plate 10 in</td>
<td>Form is being filled</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>Enter pack</td>
<td>500</td>
<td>Form is being filled</td>
<td>Yes</td>
</tr>
<tr>
<td>3</td>
<td>Enter unit</td>
<td>pz</td>
<td>Form is being filled</td>
<td>Yes</td>
</tr>
<tr>
<td>4</td>
<td>Enter item number</td>
<td>4000000</td>
<td>Form is being filled</td>
<td>Yes</td>
</tr>
<tr>
<td>5</td>
<td>Enter invoice number</td>
<td>1260000</td>
<td>Form is being filled</td>
<td>Yes</td>
</tr>
<tr>
<td>6</td>
<td>Select category</td>
<td>Paper</td>
<td>Form is being filled</td>
<td>Yes</td>
</tr>
<tr>
<td>7</td>
<td>Enter subcategory</td>
<td>Plates</td>
<td>Form is being filled</td>
<td>Yes</td>
</tr>
<tr>
<td>8</td>
<td>Select provider</td>
<td>SYSCO</td>
<td>Form is being filled</td>
<td>Yes</td>
</tr>
<tr>
<td>9</td>
<td>Add comment</td>
<td>Can be delivered same day</td>
<td>Form is being filled</td>
<td>Yes</td>
</tr>
<tr>
<td>10</td>
<td>Item price</td>
<td>.028</td>
<td>Form is being filled</td>
<td>Yes</td>
</tr>
<tr>
<td>11</td>
<td>Special</td>
<td>No check mark</td>
<td>Form is being filled</td>
<td>Yes</td>
</tr>
<tr>
<td>12</td>
<td>Save</td>
<td>Click on save button</td>
<td>Save to DB all the info on the form</td>
<td>Yes</td>
</tr>
<tr>
<td>13</td>
<td>Reset</td>
<td>Click on reset button</td>
<td>Clear all the info on the form</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Table 10. Test Case for View Item Option on Administrator Menu

<table>
<thead>
<tr>
<th>Step No</th>
<th>Description</th>
<th>Used Data</th>
<th>Actual Results</th>
<th>Pass</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Click the link View Items</td>
<td>Click Link</td>
<td>Table with all saved products is displayed</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>Remove option on view items link</td>
<td>Press the remove button of any of the items on the list.</td>
<td>The item is deleted from the DB.</td>
<td>Yes</td>
</tr>
<tr>
<td>3</td>
<td>Edit option on view items link</td>
<td>Press the edit button of any of the items on the list.</td>
<td>A form is displayed with the info of the product that need to be edited</td>
<td>Yes</td>
</tr>
<tr>
<td>4</td>
<td>Edit actual entree</td>
<td>Edit the existing data for paper bowl 10 in</td>
<td>Information on the form is being changed</td>
<td>Yes</td>
</tr>
<tr>
<td>5</td>
<td>Change name, category, item number, cost price</td>
<td>New name: paper bowl</td>
<td>New Information is on the form</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>New category: paper</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>New item number:4002</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>New Cost Price:0.021</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Save option</td>
<td>New information on form.</td>
<td>The item was edited. New info is saved on DB.</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Table 11. Test Case for Add Provider Option on Administrator Menu

<table>
<thead>
<tr>
<th>Step No</th>
<th>Description</th>
<th>Used Data</th>
<th>Actual Results</th>
<th>Pass</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Click the link Add Provider</td>
<td>Click Link</td>
<td>A page with an empty table is open.</td>
<td>Yes</td>
</tr>
</tbody>
</table>
| 5       | Enter data requested for new provider | Provider: KFC  
  Sales Rep: P.Mcdonald  
  Prov's phone: 909-312-2423  
  Prov's Fax: 909-312-2423  
  Prov's mobile: 909-312-2423  
  Prov's email: mcdonald@kfc.com  
  Prov's website: www.kfc.com | New Information is on the form.                                           | Yes  |
| 6       | Add option                       | New information on form.                                                 | The new provider is saved on DB.                    | Yes  |
| 7       | Clear Option                     | All the information on the fields is removed without being saved.        | Form is ready for new information                   | Yes  |
Table 12. Test Case for View Provider Option on Administrator Menu

<table>
<thead>
<tr>
<th>Step No</th>
<th>Description</th>
<th>Used Data</th>
<th>Actual Results</th>
<th>Pass</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Click the link view Provider</td>
<td>Click Link</td>
<td>A list with all the current providers is displayed.</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>Remove option on view providers link</td>
<td>Press the remove button of any of the providers on the list.</td>
<td>The provider is deleted from the DB.</td>
<td>Yes</td>
</tr>
<tr>
<td>3</td>
<td>Edit option on view providers link</td>
<td>Press the edit button of any of the providers on the list.</td>
<td>A form is displayed with the info of the provider that need to be edited</td>
<td>Yes</td>
</tr>
<tr>
<td>4</td>
<td>Edit a Provider</td>
<td>KFC provider existing data</td>
<td>Information on the form is being changed</td>
<td>Yes</td>
</tr>
</tbody>
</table>
| 5       | Change email, name, phone, fax, mobile| New e mail: pmcd@kfc.com  
New phone: 909-312-2424  
New fax: 909-312-2525  
New mobile: 909-312-2626 | New Information is on the form                              | Yes  |
| 6       | Add option                            | New information on form.                                   | The new information was edited; new info is saved on DB.        | Yes  |
Table 13. Test Case for New User Option on Administrator Menu

<table>
<thead>
<tr>
<th>Step No</th>
<th>Description</th>
<th>Used Data</th>
<th>Actual Results</th>
<th>Pass</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Click the link New User</td>
<td>Click Link</td>
<td>A page with an empty table is open.</td>
<td>Yes</td>
</tr>
</tbody>
</table>
| 2       | Enter data requested for new user | username: g1  
First Name: Gerardo  
Last Name:  
Administrator  
Password: gll  
Administrator: Yes | New Information is on the form               | Yes  |
| 3       | Add option                   | New information on form.               | The new user is added on DB as an Administrator. | Yes  |
| 4       | Clear Option                 | All the information on the fields is removed without been saved. | Form is ready for new information              | Yes  |
| 5       | Enter data requested for new user | username: g2  
First Name: Gerardo  
Last Name: Staff  
Password: g2l  
Administrator: No | New Information is on the form               | Yes  |
| 6       | Add option                   | New information on form.               | The new user is added on DB as a Staff.        | Yes  |
| 7       | Clear Option                 | All the information on the fields is removed without been saved. | Form is ready for new information              | Yes  |
Table 14. Test Case for View User Option on Administrator Menu

<table>
<thead>
<tr>
<th>Step No</th>
<th>Description</th>
<th>Used Data</th>
<th>Actual Results</th>
<th>Pass</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Click the link view users</td>
<td>Click Link</td>
<td>A list with all the current users is displayed.</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>Remove option on view users link</td>
<td>Press the remove button of any of the user on the list.</td>
<td>The user is deleted from the DB.</td>
<td>Yes</td>
</tr>
<tr>
<td>3</td>
<td>Edit option on view users link</td>
<td>Press the edit button of any of the users on the list.</td>
<td>A form is displayed with the info of the user that can be edited</td>
<td>Yes</td>
</tr>
<tr>
<td>4</td>
<td>Edit a user</td>
<td>G2 user existing data</td>
<td>Information on the form is being changed</td>
<td>Yes</td>
</tr>
<tr>
<td>5</td>
<td>Change username, first name, last name, password and user type.</td>
<td>New username: g21 New First Name: Jerry New Last name: Admin New Password: g2121 New User type: Admin</td>
<td>New Information is on the form</td>
<td>Yes</td>
</tr>
<tr>
<td>6</td>
<td>Add option</td>
<td>New information on form.</td>
<td>The new information was edited; new info is saved on DB.</td>
<td>Yes</td>
</tr>
<tr>
<td>7</td>
<td>Clear option</td>
<td>All the information on the fields is removed without being saved.</td>
<td>Form is ready for new information</td>
<td>Yes</td>
</tr>
<tr>
<td>8</td>
<td>Access login page with new user data</td>
<td>Username: g21 Password: g2121 Admin: Yes</td>
<td>Granted access</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Table 15. Test Case for Search Option on Administrator Menu

<table>
<thead>
<tr>
<th>Step No</th>
<th>Description</th>
<th>Used Data</th>
<th>Actual Results</th>
<th>Pass</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Click the link Search</td>
<td>Click Link</td>
<td>An empty form is displayed.</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>Search using Item name</td>
<td>French Fries and search option</td>
<td>French Fries product was found.</td>
<td>Yes</td>
</tr>
<tr>
<td>3</td>
<td>Search using Item Number</td>
<td>6000001 and search option</td>
<td>French Fries product was found.</td>
<td>Yes</td>
</tr>
<tr>
<td>4</td>
<td>Search Item using invoice number</td>
<td>128000 and search option</td>
<td>All the products from that invoice number were displayed, including French fries</td>
<td>Yes</td>
</tr>
<tr>
<td>5</td>
<td>Select Provider from the drop down box.</td>
<td>SYSCO and search option</td>
<td>All the products from SYSCO were displayed, including French fries</td>
<td>Yes</td>
</tr>
<tr>
<td>6</td>
<td>Click on Reset link</td>
<td>Click reset Link</td>
<td>All the info is clear off giving you the option to start a new search.</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Table 16. Test Case for Logout Option on Administrator Menu

<table>
<thead>
<tr>
<th>Step No</th>
<th>Description</th>
<th>Used Data</th>
<th>Actual Results</th>
<th>Pass</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Click the link If logout</td>
<td>Click logout link</td>
<td>Session is close. Redirected to login page.</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Test Case | Logout option on Administrator menu.
Test Objective | To see whether or not the logout link on Administrator’s menu work properly.
Test Data | Click the link logout on Admin menu.

Table 17. Test Case for Login Page

<table>
<thead>
<tr>
<th>Step No</th>
<th>Description</th>
<th>Used Data</th>
<th>Actual Results</th>
<th>Pass</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Enter user name and press Login</td>
<td>Username:g2</td>
<td>Access Denied</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>Enter password and press Login</td>
<td>Password:g21</td>
<td>Access Denied</td>
<td>Yes</td>
</tr>
<tr>
<td>3</td>
<td>Enter User name and password</td>
<td>Username:g2</td>
<td>Access to user's menu</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Password:g21</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Table 18. Test Case for Home Link

<table>
<thead>
<tr>
<th>Step No</th>
<th>Description</th>
<th>Used Data</th>
<th>Actual Results</th>
<th>Pass</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Test on home link</td>
<td>Click on Home Link</td>
<td>Home page is refreshed</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>Click on Specials link and then on Home link.</td>
<td>Click on Special link / Click on Home Link.</td>
<td>List of products on special were displayed. Once we press Home link Staff user was redirected to Home page.</td>
<td>Yes</td>
</tr>
</tbody>
</table>

### Table 19. Test Case for New Product Link

<table>
<thead>
<tr>
<th>Step No</th>
<th>Description</th>
<th>Used Data</th>
<th>Actual Results</th>
<th>Pass</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Click on New Product Link</td>
<td>Click on link</td>
<td>New page is displayed, ready to create a new product.</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>Create a New Product</td>
<td>Select the items from the list to create the new product, give a name and number to the product and save it.</td>
<td>Product was successfully added to the database, including the price.</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Table 20. Test Case for New Combo Link

<table>
<thead>
<tr>
<th>Step No</th>
<th>Description</th>
<th>Used Data</th>
<th>Actual Results</th>
<th>Pass</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Click on New Combo Link</td>
<td>Click on link</td>
<td>New page is displayed, showing all the items ready to create a new combo.</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>Create a New Product</td>
<td>Select the items from the list to create the new product, give a name and number to the product and save it.</td>
<td>Yes</td>
<td></td>
</tr>
</tbody>
</table>

Table 21. Test Case for Product/Combos Link

<table>
<thead>
<tr>
<th>Step No</th>
<th>Description</th>
<th>Used Data</th>
<th>Actual Results</th>
<th>Pass</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Click on Product/Combo Link</td>
<td>Click on link</td>
<td>New page is displayed showing all the products and combos on the database.</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Table 22. Test Case for Specials Link

<table>
<thead>
<tr>
<th>Step No</th>
<th>Description</th>
<th>Used Data</th>
<th>Actual Results</th>
<th>Pass</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Click on Specials</td>
<td>Click on link</td>
<td>New page is displayed showing all the products checked as a special</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Link</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
4.1 Introduction

Included in Chapter Four a presentation of the user's manual of the project.

4.2 General Layout

GRIPED has two different main menus, one for administrators and a second one for users. Both of them are accessed using the same logging page. (Figure 4)

![Welcome to GRIPED]

Figure 4. Login Page

Login as an Administrator. The first Admin Menu (Figure 5) contains nine available options which are: Home, New Items, View Items, Add Provider, View Provider, New User, View User, Search, and Logout. Each of the links
gives to the administrator different functions and services.

Welcome to GRIPED

Gerardo Administrator

GRIPED

Welcome to "Get the Right Price Everyday". GRIPED was designed for the purpose of enabling a systematic approach to the setting of accurate product pricing in the food industry. With GRIPED, food industry professionals are able to ensure that their inventory is always properly valued as GRIPED ensures that their prices are based on their original purchase price. Specifically, GRIPED includes all of the following functionality:

- Access to the most recent prices of each item, product and combo
- Price comparisons between vendors
- Helps staff decide on when is the right time to set a special price of a product and which products should be included
- Manage and control the inventory and prices
- Create new menu products based on the purchased items registered in the database
- Allows users to calculate and create prices of new items, and products
- Allows access to create products and combinations from existing items

Figure 5. Administrator's Menu

All the options are hyperlinks to different pages where the administrator will be able to review, execute or visit a different page. The links on the menu will allow the administrator to manage the information. The Administrator will be able to input, add, remove, view, search and edit the information using these links on the main menu.

33
Table 23. Administrator Inputs and Corresponding Actions

<table>
<thead>
<tr>
<th>User Input</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click on Home</td>
<td>Display the first Admin Menu (Fig 5)</td>
</tr>
<tr>
<td>Click on New Item</td>
<td>Open a form to create a New Item</td>
</tr>
<tr>
<td>Click on View Item</td>
<td>Display Existing Products.</td>
</tr>
<tr>
<td>Click on Add Provider</td>
<td>Open a query to add a New Provider.</td>
</tr>
<tr>
<td>Click on View Providers</td>
<td>Display the providers on the System</td>
</tr>
<tr>
<td>Click on New User</td>
<td>Open a Form to enter a new user.</td>
</tr>
<tr>
<td>Click on View Users</td>
<td>Display a page with existing users.</td>
</tr>
<tr>
<td>Click on Search</td>
<td>Display a query search page.</td>
</tr>
<tr>
<td>Click on Logout</td>
<td>End the current logging session.</td>
</tr>
</tbody>
</table>

4.3 Home Link Information

This link will display the general information about GRIPED. So, user and administrator can have an idea of what the system can do (Figure 6).

Welcome to GRIPED
Gerardo Administrator

Welcome to "Get the Right Price Everyday". GRIPED was designed for the purpose of enabling a systematic approach to the setting of accurate product pricing in the food industry. With GRIPED food industry professionals are able to ensure that their inventory is always properly valued as GRIPED ensures that their prices are based on their original purchase price. Specifically, GRIPED includes all of the following functionality:

- Access to the most recent prices of each item, product, and combo
- Price comparisons between vendors
- Helps staff decide on when is the right time to set a special price of a product and which products should be included
- Manage and control the inventory and prices
- Create new menu products based on the purchased items registered in the database
- Allows users to calculate and create prices of new items, and products
- Allows users to create products and combinations from existing items

Figure 6. Home Link Information

4.4 New Item Link Information

This link offers the option to the Administrator to create a New Item in the database. So once the Administrator enters all the information for the new item, it can be saved. Every new item that is created using this
option will start forming part of the whole inventory of items. In this way the Administrator can populate the database with new products from invoices (Figure 7).

A description of every required input on new item form is provided on the table.
Table 24. New Item Link on Administrator Menu

<table>
<thead>
<tr>
<th>Administrator Input</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item Name</td>
<td>Use the Item Name</td>
</tr>
<tr>
<td>Pack</td>
<td>Use item pack description.</td>
</tr>
<tr>
<td>Unit</td>
<td>Display the unit amount that is going to be price out.</td>
</tr>
<tr>
<td>Item Number</td>
<td>Enter the item number from invoice.</td>
</tr>
<tr>
<td>Invoice Number</td>
<td>Enter the invoice number.</td>
</tr>
<tr>
<td>Category</td>
<td>Choose a category for the new item from the drop down box.</td>
</tr>
<tr>
<td>Subcategory</td>
<td>Enter a subcategory for the new item.</td>
</tr>
<tr>
<td>Provider</td>
<td>Choose the provider from the drop down box.</td>
</tr>
<tr>
<td>Comments</td>
<td>Enter a comment about this product. Deliver day, etc.</td>
</tr>
<tr>
<td>Item Price</td>
<td>Enter the price for the product described.</td>
</tr>
<tr>
<td>Special</td>
<td>Check this box for special price on the product.</td>
</tr>
<tr>
<td>Save</td>
<td>Save the actual information on the screen.</td>
</tr>
<tr>
<td>Reset</td>
<td>Clear the form without saving anything.</td>
</tr>
</tbody>
</table>

Once the Save button has been pushed, the information will be saved in the database and a table with all the saved products will be displayed every time that we save a new item (Figure 8).
Your New Entry Has Been Created

<table>
<thead>
<tr>
<th>Item Name</th>
<th>Pack Unit</th>
<th>Item Number</th>
<th>Invoice Category</th>
<th>Sub-Category</th>
<th>Provider</th>
<th>Comments</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>Paper Plate 10 in</td>
<td>5000 pz</td>
<td>6000001</td>
<td>Paper Plates</td>
<td>American Paper &amp; Plastic</td>
<td>Can be delivered same day</td>
<td>$0.081</td>
<td></td>
</tr>
<tr>
<td>Straw Wrapped</td>
<td>5000003</td>
<td>27000007</td>
<td>Plastic Straw</td>
<td>American Paper &amp; Plastic</td>
<td>Coca-Cola also delivers this product</td>
<td>$0.003</td>
<td></td>
</tr>
<tr>
<td>French Fries 1 oz</td>
<td>4000001</td>
<td>22000003</td>
<td>Sides Fries</td>
<td>SYSCO</td>
<td>Cerre comes with 5 boxes of 25s</td>
<td>$0.164</td>
<td></td>
</tr>
<tr>
<td>French Fries 5 oz</td>
<td>4000001</td>
<td>22000003</td>
<td>Sides Fries</td>
<td>SYSCO</td>
<td>Cerre come with 5 boxes of 25s</td>
<td>$0.164</td>
<td></td>
</tr>
<tr>
<td>Hamburger Bun 6 ea</td>
<td>1000002</td>
<td>22000003</td>
<td>Bread Buns</td>
<td>Globe</td>
<td>Delivery every day</td>
<td>$0.25</td>
<td></td>
</tr>
<tr>
<td>Onion Rings 1 oz</td>
<td>4000001</td>
<td>22000003</td>
<td>Sides Onion Rings</td>
<td>SYSCO</td>
<td>The above price is for blue label OR</td>
<td>$0.53</td>
<td></td>
</tr>
<tr>
<td>Hamburger Patty 8 of 2 oz</td>
<td>1000001</td>
<td>22000003</td>
<td>Patty Hamburger</td>
<td>SYSCO</td>
<td>Delivery Fri</td>
<td>$0.25</td>
<td></td>
</tr>
<tr>
<td>French Fries 20 oz</td>
<td>2000004</td>
<td>22000003</td>
<td>Plastic Fries</td>
<td>SYSCO</td>
<td>Delivery every week</td>
<td>$0.002</td>
<td></td>
</tr>
<tr>
<td>Paper Cup 32 Oz</td>
<td>1000001</td>
<td>22000003</td>
<td>Paper Cups</td>
<td>American Paper &amp; Plastic</td>
<td>$0.018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pickle Del</td>
<td>5000001</td>
<td>22000003</td>
<td>Vegetable Pickles</td>
<td>AAA</td>
<td>Ridged spear on pack</td>
<td>$0.12</td>
<td></td>
</tr>
<tr>
<td>Tomato Strips 2 oz</td>
<td>5000002</td>
<td>22000003</td>
<td>Vegetable Slices</td>
<td>AAA</td>
<td>Each slice is about 1 oz.</td>
<td>$0.185</td>
<td></td>
</tr>
<tr>
<td>Paper Cap 20 Oz</td>
<td>1000001</td>
<td>22000003</td>
<td>Paper Caps</td>
<td>AP&amp;F</td>
<td></td>
<td>$0.015</td>
<td></td>
</tr>
<tr>
<td>Green Leaf</td>
<td>1000003</td>
<td>22000003</td>
<td>Vegetable Leaves</td>
<td>AAA</td>
<td>Considering 6 leaves from every head which is the standard.</td>
<td>$0.041</td>
<td></td>
</tr>
</tbody>
</table>

Figure 8. Products Displayed After You Save a New Item

4.5 View Items Option

With this option the Administrator will have the opportunity to see all the items in the system. Also, there will be the chance to edit or remove any product from the system (Figure 9).
### EXISTING ENTRIES

<table>
<thead>
<tr>
<th>Category</th>
<th>Name of Item</th>
<th>Item Number</th>
<th>Cost Price</th>
<th>Edit</th>
<th>Remove</th>
</tr>
</thead>
<tbody>
<tr>
<td>Beverages</td>
<td>BIB Coca Cola 32 oz</td>
<td>1000003</td>
<td>.256</td>
<td>E</td>
<td>X</td>
</tr>
<tr>
<td>Beverages</td>
<td>BIB Coca Cola 20 oz</td>
<td>1000002</td>
<td>.160</td>
<td>E</td>
<td>X</td>
</tr>
<tr>
<td>Beverages</td>
<td>Soft Drink</td>
<td>1000004</td>
<td>.35</td>
<td>E</td>
<td>X</td>
</tr>
<tr>
<td>Bread</td>
<td>Hamburger Bun 6 oz</td>
<td>2000002</td>
<td>.25</td>
<td>E</td>
<td>X</td>
</tr>
<tr>
<td>Bread</td>
<td>Sliced Croissant 4 oz</td>
<td>2000003</td>
<td>.15</td>
<td>E</td>
<td>X</td>
</tr>
<tr>
<td>Bread</td>
<td>Sour Dough 2 Slices</td>
<td>2000007</td>
<td>.18</td>
<td>E</td>
<td>X</td>
</tr>
<tr>
<td>Bread</td>
<td>Chalet Bread 2 pz</td>
<td>2000004</td>
<td>.25</td>
<td>E</td>
<td>X</td>
</tr>
<tr>
<td>Dairy</td>
<td>Yellow Cheese 2 Slices</td>
<td>8000001</td>
<td>.25</td>
<td>E</td>
<td>X</td>
</tr>
<tr>
<td>Meats</td>
<td>Hamburger Patty 8 oz</td>
<td>3000001</td>
<td>.75</td>
<td>E</td>
<td>X</td>
</tr>
<tr>
<td>Meats</td>
<td>Bacon 4 Slices</td>
<td>3000004</td>
<td>.30</td>
<td>E</td>
<td>X</td>
</tr>
<tr>
<td>Meats</td>
<td>Smoked Ham 4 oz Portion</td>
<td>3000008</td>
<td>.35</td>
<td>E</td>
<td>X</td>
</tr>
<tr>
<td>Meats</td>
<td>Hamburger Patty 4 oz</td>
<td>3000005</td>
<td>.25</td>
<td>E</td>
<td>X</td>
</tr>
<tr>
<td>Paper</td>
<td>Paper Plate 10 in</td>
<td>4000001</td>
<td>.028</td>
<td>E</td>
<td>X</td>
</tr>
<tr>
<td>Paper</td>
<td>Paper Cup 32 Oz</td>
<td>7000001</td>
<td>.018</td>
<td>E</td>
<td>X</td>
</tr>
<tr>
<td>Paper</td>
<td>Paper Cup 20 Oz</td>
<td>7000002</td>
<td>.015</td>
<td>E</td>
<td>X</td>
</tr>
<tr>
<td>Plastic</td>
<td>Straw Wrapped</td>
<td>4000003</td>
<td>.003</td>
<td>E</td>
<td>X</td>
</tr>
<tr>
<td>Plastic</td>
<td>Plastic Lid 20 oz</td>
<td>4000004</td>
<td>.002</td>
<td>E</td>
<td>X</td>
</tr>
<tr>
<td>Plastic</td>
<td>Plastic Lid 32 oz</td>
<td>4000002</td>
<td>.003</td>
<td>E</td>
<td>X</td>
</tr>
<tr>
<td>Sides</td>
<td>French Fries 7oz</td>
<td>6000001</td>
<td>.164</td>
<td>E</td>
<td>X</td>
</tr>
</tbody>
</table>

**Figure 9. View Items Display**

If administrators click the edit button next to the cost price column of any item they can change its price, name, category, etc. using the edit form (Figure 10). Also if he/she clicks the remove button on this page, the one with the red "x" next to the edit button, the administrator can delete it from the database.
In this section, the administrator can change any information in the fields and save it to be used later on.

4.6 Add Provider Option

This option will let the administrator add a new provider into the database (Figure 11).
Welcome to GRIPED
Gerardo Administrator

ADD A PROVIDER
Please fill out the form to add a provider.
* Are required fields.
*Provider: 
*Sales Representative: 
*Provider’s Phone: 
Provider’s Fax: 
Provider’s Mobile: 
Provider’s Email: 
Provider’s Website: 

Add  Clear

Figure 11. Add a New Provider

Table 25. Add Provider link on Administrator Menu

<table>
<thead>
<tr>
<th>Administrator Input</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Provider</td>
<td>Use the Provider’s Name. eg. SYSCO.</td>
</tr>
<tr>
<td>Sales Representative</td>
<td>Use the name of the sales person.</td>
</tr>
<tr>
<td>Provider’s phone</td>
<td>Enter the provider’s phone number.</td>
</tr>
<tr>
<td>Provider’s fax</td>
<td>Enter the provider’s fax number.</td>
</tr>
<tr>
<td>Provider’s mobile</td>
<td>Enter the provider’s mobile number.</td>
</tr>
<tr>
<td>Provider’s e-mail</td>
<td>Enter the provider’s e-mail.</td>
</tr>
<tr>
<td>Provider’s website</td>
<td>Enter the provider’s website.</td>
</tr>
<tr>
<td>Add Button</td>
<td>Save the current information.</td>
</tr>
<tr>
<td>Clear</td>
<td>Clear all the inputs.</td>
</tr>
</tbody>
</table>

Once the administrator presses the add button, the provider is added to the database. The list of providers can be seen with the view provider option.
4.7 View Provider Option

Here the administrator can see all the existing providers in the database (Figure 12). The information of the providers on the display table is as follows: Provider name, Sales representative, provider's phone, provider's fax, provider's mobile provider's email and provider's website. In addition, the administrator will have the options to edit and remove the existing information about the providers.

<table>
<thead>
<tr>
<th>Provider</th>
<th>Sales Representative</th>
<th>Provider's Phone</th>
<th>Provider's Fax</th>
<th>Provider's Email</th>
<th>Provider's Website</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dairy</td>
<td>Rosy Clark</td>
<td>909-210-3454</td>
<td>909-210-3455</td>
<td>clarke@am dairy.com</td>
<td><a href="http://www.american-dairy.com">www.american-dairy.com</a></td>
</tr>
<tr>
<td>KFC</td>
<td>Paul McDonald</td>
<td>909-312-2423</td>
<td>909-312-2424</td>
<td><a href="mailto:mcdonald@kfc.com">mcdonald@kfc.com</a></td>
<td><a href="http://www.kfc.com">www.kfc.com</a></td>
</tr>
<tr>
<td>Pizza Hut</td>
<td>Roxana Peterson</td>
<td>909-081-2104</td>
<td>909-081-2125</td>
<td><a href="mailto:peterson@pizzahut.com">peterson@pizzahut.com</a></td>
<td><a href="http://www.pizzahut.com">www.pizzahut.com</a></td>
</tr>
<tr>
<td>Jcrrys</td>
<td>Rachel Dixon</td>
<td>909-221-3312</td>
<td>909-221-3313</td>
<td><a href="mailto:Dixon@jcrrys.com">Dixon@jcrrys.com</a></td>
<td><a href="http://www.jcrrys.com">www.jcrrys.com</a></td>
</tr>
<tr>
<td>Arrowhead</td>
<td>Rachel Dixon</td>
<td>909-221-2333</td>
<td>909-221-3313</td>
<td><a href="mailto:Dixon@jcrrys.com">Dixon@jcrrys.com</a></td>
<td><a href="http://www.jcrrys.com">www.jcrrys.com</a></td>
</tr>
<tr>
<td>Water</td>
<td>James Smith</td>
<td>909-213-2325</td>
<td>909-213-2326</td>
<td><a href="mailto:smith@water.com">smith@water.com</a></td>
<td><a href="http://www.water.com">www.water.com</a></td>
</tr>
<tr>
<td>Jerry</td>
<td>Robert Robles</td>
<td>213-953-5125</td>
<td>213-953-5135</td>
<td><a href="mailto:Robles@jerryd.com">Robles@jerryd.com</a></td>
<td><a href="http://www.jerryd.com">www.jerryd.com</a></td>
</tr>
<tr>
<td>Deli</td>
<td>Robert Robles</td>
<td>213-953-5125</td>
<td>213-953-5135</td>
<td><a href="mailto:robles@deli.com">robles@deli.com</a></td>
<td><a href="http://www.deli.com">www.deli.com</a></td>
</tr>
</tbody>
</table>

Figure 12. View Providers Information

If the administrator clicks the edit button which is next to the provider's website, he will have the option to modify the existing information from that specific provider such as providers name, phone numbers, and email among others (Figure 13). On the other hand, if he clicks the red "x" next to edit, the administrator will
permanently remove the information of that specific provider from the database.

Welcome to GRIPED
Gerardo Administrator

UPDATE A Provider

Please fill out the form to add a provider.
* Are required fields.

*Provider: American Dairy
*Sales Representative: Rosy Clark
*Provider's Phone: 909-212-3434
Provider's Fax: 909-212-3435
Provider's Mobile: 909-212-3435
Provider's Email: clarkr@amdairy.com
Provider's Website: www.americandairy.com

Add | Clear

Figure 13. Edit or Update a Provider

4.8 New User Option

This option will let the administrator add a new user into the database (Figure 14). The table 26 will provide an explanation of the required fields on this form.
Welcome to GRIPED
Gerardo Administrator

Figure 14. Create a New User

Table 26. New User Link on Administrator Menu

<table>
<thead>
<tr>
<th>Input</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Username</td>
<td>Enter a username you like it.</td>
</tr>
<tr>
<td>First Name</td>
<td>Use first name.</td>
</tr>
<tr>
<td>Last Name</td>
<td>Use last name.</td>
</tr>
<tr>
<td>Password</td>
<td>Enter a password.</td>
</tr>
<tr>
<td>Check box for administrator</td>
<td>Check the box to get admin rights.</td>
</tr>
<tr>
<td>rights</td>
<td></td>
</tr>
<tr>
<td>Add button</td>
<td>Add the username in to the database.</td>
</tr>
<tr>
<td>Clear Button</td>
<td>Clear the information on the fields.</td>
</tr>
</tbody>
</table>
4.9 View Users Option on Administrator Menu

This option will allow the administrator to view the list of all existing users on the system, including their user profile or user type. In addition from this option the administrator will be able to edit or remove any information from the existing users (Figure 15).

![Welcome to GRIPED](image)

<table>
<thead>
<tr>
<th>VIEW USERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>User Name</td>
</tr>
<tr>
<td>hsimpson</td>
</tr>
<tr>
<td>bsimpson</td>
</tr>
<tr>
<td>G2</td>
</tr>
<tr>
<td>G1</td>
</tr>
</tbody>
</table>

Figure 15. View Existing Users

If the administrator presses the button edit which is next to the user type column, he will have the option to modify the existing information from that specific user such as user name, first name, last name, password, and even give a different rights to that user (Figure 16). On the other hand, if he presses the red "x" next to edit,
the administrator will permanently remove the information of that specific user from the database.

**Figure 16. Edit an Existing User**

4.10 Search Option on Administrator Menu

This option will give the opportunity to the Administrator to search for an existing product in the database. The Administrator will have several choices to search for an item. Table 27 explains in detail the required fields to use the search option from the administrator menu.
Table 27. Search Link on Administrator Menu

<table>
<thead>
<tr>
<th>Administrator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item Name</td>
<td>Enter the item name (See Figure 17)</td>
</tr>
<tr>
<td>Item Number</td>
<td>Enter the item number (See Figure 19)</td>
</tr>
<tr>
<td>Invoice Number</td>
<td>Enter the invoice number (See Figure 21)</td>
</tr>
<tr>
<td>Provider</td>
<td>Enter provider's name (See Figure 23)</td>
</tr>
<tr>
<td>Search button</td>
<td>Start the query search.</td>
</tr>
<tr>
<td>Clear Button</td>
<td>Clear the information on the fields</td>
</tr>
</tbody>
</table>

For example, one of the choices to search is using the item name (Figure 17).

Welcome to GRIPED
Gerardo Administrator

![Welcome to GRIPED](image)

<table>
<thead>
<tr>
<th>Items</th>
<th>SEARCH</th>
</tr>
</thead>
<tbody>
<tr>
<td>Home</td>
<td>Item Name:</td>
</tr>
<tr>
<td>New Item</td>
<td>Item Number:</td>
</tr>
<tr>
<td>View Items</td>
<td>Invoice Number:</td>
</tr>
<tr>
<td>Add Provider</td>
<td>Provider:</td>
</tr>
<tr>
<td>View Providers</td>
<td></td>
</tr>
<tr>
<td>New User</td>
<td></td>
</tr>
<tr>
<td>View Users</td>
<td></td>
</tr>
<tr>
<td>Search</td>
<td></td>
</tr>
<tr>
<td>Logout</td>
<td></td>
</tr>
</tbody>
</table>

Figure 17. Search an Item Using the Item Name

All the products that match the description name will be displayed on the search results (Figure 18). In other words we can have several products in the search results by item name.
Welcome to GRIPED
Gerardo Administrator

<table>
<thead>
<tr>
<th>Item Name</th>
<th>French Fries 7oz</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pack</td>
<td>400 oz</td>
</tr>
<tr>
<td>Unit</td>
<td>7 oz</td>
</tr>
<tr>
<td>Item Number</td>
<td>6000001</td>
</tr>
<tr>
<td>Invoice Number</td>
<td>128000</td>
</tr>
<tr>
<td>Category</td>
<td>Sides</td>
</tr>
<tr>
<td>SubCategory</td>
<td>Fries</td>
</tr>
<tr>
<td>Provider</td>
<td>SYSCO</td>
</tr>
<tr>
<td>Comments</td>
<td>Case comes with 5 bags of 5lbs</td>
</tr>
<tr>
<td>Price</td>
<td>164</td>
</tr>
</tbody>
</table>

Figure 18. Search Results Using the Item Name

In addition, other available option to search for an item is using the item number (Figure 19). By using this search option, the administrator has the opportunity to pull the information from a specific item number. This option will help to the administrator to review or compare prices when a new vendor visits or just to see the price fluctuation of that specific product. The search result is going to match just the specific item number (Figure 20).
Welcome to GRIPED
Gerardo Administrator

SEARCH

Item Name: 
Item Number: 6000001
Invoice Number: 
Provider: 

Search  Reset

Figure 19. Search an Item Using its Item Number
Welcome to GRIPED

Gerardo Administrator

<table>
<thead>
<tr>
<th>Item Name</th>
<th>French Fries 7oz</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pack</td>
<td>400 oz</td>
</tr>
<tr>
<td>Unit</td>
<td>7 oz</td>
</tr>
<tr>
<td>Item Number</td>
<td>6000001</td>
</tr>
<tr>
<td>Invoice Number</td>
<td>128000</td>
</tr>
<tr>
<td>Category</td>
<td>Sides</td>
</tr>
<tr>
<td>SubCategory</td>
<td>Fries</td>
</tr>
<tr>
<td>Provider</td>
<td>Sysco</td>
</tr>
<tr>
<td>Comments</td>
<td>Case comes with 5 bags of 5lbs</td>
</tr>
<tr>
<td>Price</td>
<td>1.64</td>
</tr>
</tbody>
</table>

Figure 20. Search Results Using the Item Number

Another available option for search is using the Invoice number (Figure 21). This search will help to see all the products information from a specific invoice number. With that information on hand, the administrator can see on the system all the products shipped on that specific invoice number.
The search results will display all the products that match that invoice number (Figure 22). With this search, we are going to see in the search result as many products as there are listed for that specific invoice number.
<table>
<thead>
<tr>
<th>Home</th>
<th>Item Name</th>
<th>French Fries 7oz</th>
</tr>
</thead>
<tbody>
<tr>
<td>New Item</td>
<td>Pack</td>
<td>400 oz</td>
</tr>
<tr>
<td>View Items</td>
<td>Unit</td>
<td>7 oz</td>
</tr>
<tr>
<td>Add Item</td>
<td>Item Number</td>
<td>6000001</td>
</tr>
<tr>
<td>View Items</td>
<td>Invoice Number</td>
<td>1280000</td>
</tr>
<tr>
<td>View Items</td>
<td>Category</td>
<td>Sides</td>
</tr>
<tr>
<td>View Items</td>
<td>SubCategory</td>
<td>Fries</td>
</tr>
<tr>
<td>View Items</td>
<td>Provider</td>
<td>SYSCO</td>
</tr>
<tr>
<td>View Items</td>
<td>Comments</td>
<td>Case comes with 5 bags of 5lbs</td>
</tr>
<tr>
<td>View Items</td>
<td>Price</td>
<td>164</td>
</tr>
<tr>
<td>Home</td>
<td>Item Name</td>
<td>Green Leaf</td>
</tr>
<tr>
<td>Home</td>
<td>Pack</td>
<td>1/25 lb</td>
</tr>
<tr>
<td>Home</td>
<td>Unit</td>
<td>Leaf</td>
</tr>
<tr>
<td>Home</td>
<td>Item Number</td>
<td>5000003</td>
</tr>
<tr>
<td>Home</td>
<td>Invoice Number</td>
<td>1280000</td>
</tr>
<tr>
<td>Home</td>
<td>Category</td>
<td>Vegetables</td>
</tr>
<tr>
<td>Home</td>
<td>SubCategory</td>
<td>Leefs</td>
</tr>
<tr>
<td>Home</td>
<td>Provider</td>
<td>AAA</td>
</tr>
<tr>
<td>Home</td>
<td>Comments</td>
<td>Considering 6 leaves from every head which is the standard</td>
</tr>
<tr>
<td>Home</td>
<td>Price</td>
<td>041</td>
</tr>
</tbody>
</table>

Figure 22. Search Using the Invoice Number

The last option that the administrator will have to search for an item on the system will be using the provider’s name (Figure 23). This option can be used primarily when the administrator does not have enough information about some products, specifically when he only knows the provider’s name. This option will help the administrator to narrow down the search from all the existing providers to the one he chooses. The search results are going to be all the items that match the provider’s name (Figure 24).
Welcome to GRIPED
Gerardo Administrator

SEARCH

Item Name: 
Item Number: 
Invoice Number: 
Provider: Choose a Provider

AAA
American Dairy
AP&P
Arrowhead Water
Ben & Jerrys
Coca Cola
Deli Packs
Galazzos
Hostess
KFC

Figure 23. Search an Item Using Provider’s Name
<table>
<thead>
<tr>
<th>Home</th>
<th>Item Name</th>
<th>French Fries 7oz</th>
</tr>
</thead>
<tbody>
<tr>
<td>New Item</td>
<td>Pack</td>
<td>400 oz</td>
</tr>
<tr>
<td>View Items</td>
<td>Unit</td>
<td>7 oz</td>
</tr>
<tr>
<td>Add Provider</td>
<td>Item Number</td>
<td>6000001</td>
</tr>
<tr>
<td>View Providers</td>
<td>Invoice Number</td>
<td>128000</td>
</tr>
<tr>
<td>New User</td>
<td>Category</td>
<td>Sides</td>
</tr>
<tr>
<td>View Users</td>
<td>Sub-Category</td>
<td>Fries</td>
</tr>
<tr>
<td>Search</td>
<td>Provider</td>
<td>SYSCO</td>
</tr>
<tr>
<td>Login</td>
<td>Comments</td>
<td>Case comes with 5 bags of 5 lbs</td>
</tr>
<tr>
<td></td>
<td>Price</td>
<td>1.16</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Item Name</th>
<th>Onion Rings 4 oz</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pack</td>
<td>240 oz</td>
</tr>
<tr>
<td>Unit</td>
<td>4 oz</td>
</tr>
<tr>
<td>Item Number</td>
<td>6000002</td>
</tr>
<tr>
<td>Invoice Number</td>
<td>125000</td>
</tr>
<tr>
<td>Category</td>
<td>Sides</td>
</tr>
<tr>
<td>Sub-Category</td>
<td>Onion Rings</td>
</tr>
<tr>
<td>Provider</td>
<td>SYSCO</td>
</tr>
<tr>
<td>Comments</td>
<td>The above price is for blue label OR</td>
</tr>
<tr>
<td>Price</td>
<td>0.53</td>
</tr>
</tbody>
</table>

Figure 24. Search Results Using the Provider's Name

When the Administrator is searching for a product using any of the available fields for search like item name, item number, invoice number or provider, and if the information entered does not match any of the existing entrees on the database, a message will be displayed to let the administrator know that such item is not currently in the system (Figure 25).
Welcome to GRIPED

Gerardo Administrator

Figure 25. Display Message when Name, Number or Provider Can Not Be Found On The System

4.11 Logout Option on Administrator Menu

This link will terminate the Administrator session and take him back to the logging page (Figure 4).

4.12 User’s Menu

The User’s menu contains the next options “Home”, “New Product”, “Product/Combos”, “Specials” and “Logout” (Figure 26)
Welcome to GRIPED

Gerardo Staff

Monday May 2005
10:32 am

GRIPED

Welcome to "Get the Right Price Everyday". GRIPED was designed for the purpose of enabling a systematic approach to the setting of accurate product pricing in the food industry. With GRIPED food industry professionals are able to ensure that their inventory is always properly valued as GRIPED ensures that their prices are based on their original purchase price. Specifically, GRIPED includes all of the following functionality:

- Access to the most recent prices of each item, product and combo
- Price comparisons between vendors
- Helps staff to decide on when is the right time to set a special price of a product and which products should be included
- Manage and control the inventory and prices
- Create new menu products based on the purchased items registered in the database
- Allows users to calculate and create prices of new items, and products
- Allows users to create products and combinations from existing items

Figure 26. User’s Menu

These are the options for the users:

Table 28. User’s Menu Links and Corresponding Actions

<table>
<thead>
<tr>
<th>User’s Link</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click on Home</td>
<td>Display the Users Menu (See Fig 26)</td>
</tr>
<tr>
<td>Click on New Product</td>
<td>Display page to create new product</td>
</tr>
<tr>
<td>Click on New Combo</td>
<td>Display page to create new combo</td>
</tr>
<tr>
<td>Click on Products/Combos</td>
<td>Display a list of Products/Combos created</td>
</tr>
<tr>
<td>Specials</td>
<td>Display list of the Products on special.</td>
</tr>
<tr>
<td>Logout</td>
<td>Display the login page.</td>
</tr>
</tbody>
</table>

4.13 New Product Option on User’s Menu

This option will allow the user to create a new product using all the items that the administrator already entered into the database. The products are going to be displayed by categories. So the only thing the user needs to do is check the boxes of the items that he will use for the new product. In addition, the user will also have the opportunity to name the product and assign it a number.
With this option the user can match the name and the number of any product on the existing paper menu (Figure 27).

**Figure 27. New Product**

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Product Number</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>Save</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Items</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Beverages**
- ☐ BIB Coca Cola 32 oz
- ☐ BIB Coca Cola 20 oz

**Bread**
- ☐ Hamburger Bun 6 oz
- ☐ Sliced Croissant 4 oz
- ☐ Sourdough 2 Slices
- ☐ Chalet Bread 2 pc

**Dairy**
- ☐ Yellow Cheese 2 Slices

**Meats**
- ☐ Hamburger Patty 8 oz
- ☐ Bacon 4 Slices
- ☐ Smoked Ham 4 oz Portion
- ☐ Hamburger Patty 4 oz

**Paper**
- ☐ Paper Plate 10 in
- ☐ Paper Cup 32 Oz
- ☐ Paper Cup 20 Oz

4.14 New Combo Option in User’s Menu

On this option, the user can combine all the products previously created. So, the user will have the option to combine all this products and create new combinations with them (Figure 28). In addition, a name and number can be assigned to every entrée. So users can match the name and the number of the combos on the existing paper menu.
Welcome to GRIPED
Gerardo Staff

**CREATE A COMBO**

<table>
<thead>
<tr>
<th>Description</th>
<th>Combo Number</th>
<th>Price</th>
</tr>
</thead>
</table>

**Products**
- □ Turbo Dog
- □ Classic Hamburger
- □ BLT Croissant
- □ Low Carb Burger
- □ Grilled Cheese
- □ Mother Load Burger
- □ 20 oz Beverage
- □ Boston Style Sandwich
- □ Corn Beef Hash Low Carb
- □ Power Burger
- □ Super Salad
- □ French Fries Side
- □ Onion Ring Side
- □ American Hamburger

Figure 28. New Combo Link

4.15 Products/Combo Option on User’s Menu

This option will allow the user to see a list with all the products and combos already in the database (Figure 29). Also the user will have the option to edit or remove these products or combos. So users can add or remove items or products, respectively, from them. Also the name of the product or combo can be edited. These options give the users a lot of flexibility on using all the existing products on the database. In other words, they will be able to use everything that is on inventory.
### EXISTING ENTRIES

<table>
<thead>
<tr>
<th>Name of Product</th>
<th>Product Number</th>
<th>Cost Price</th>
<th>Edit</th>
<th>Remove</th>
</tr>
</thead>
<tbody>
<tr>
<td>Turbo Dog</td>
<td>189456</td>
<td>0.436</td>
<td>$</td>
<td></td>
</tr>
<tr>
<td>Classic Hamburger</td>
<td>1234001</td>
<td>0.084</td>
<td>$</td>
<td></td>
</tr>
<tr>
<td>BLT Croissant</td>
<td>112000</td>
<td>0.672</td>
<td>$</td>
<td></td>
</tr>
<tr>
<td>Low Carb Burger</td>
<td>180000</td>
<td>1.041</td>
<td>$</td>
<td></td>
</tr>
<tr>
<td>Grilled Cheese</td>
<td>100000021</td>
<td>0.43</td>
<td>$</td>
<td></td>
</tr>
<tr>
<td>Mother Load Burger</td>
<td>125000</td>
<td>1.671</td>
<td>$</td>
<td></td>
</tr>
<tr>
<td>20 oz Beverage</td>
<td>1230005</td>
<td>0.18</td>
<td>$</td>
<td></td>
</tr>
<tr>
<td>Rookie Malt</td>
<td>118000</td>
<td>1.281</td>
<td>$</td>
<td></td>
</tr>
<tr>
<td>Boston Style Sandwich</td>
<td>1350000</td>
<td>1.83</td>
<td>$</td>
<td></td>
</tr>
<tr>
<td>Corn Beef Hash Low Carb</td>
<td>1180000</td>
<td>1.221</td>
<td>$</td>
<td></td>
</tr>
<tr>
<td>French Fries Side</td>
<td>111000</td>
<td>0.164</td>
<td>$</td>
<td></td>
</tr>
<tr>
<td>Onion Ring Side</td>
<td>1110002</td>
<td>0.33</td>
<td>$</td>
<td></td>
</tr>
<tr>
<td>American Hamburger</td>
<td>180002</td>
<td>1.903</td>
<td>$</td>
<td></td>
</tr>
<tr>
<td>Southwest Hamburger</td>
<td>180004</td>
<td>1.852</td>
<td>$</td>
<td></td>
</tr>
<tr>
<td>Super Soda 32 oz</td>
<td>180300</td>
<td>0.28</td>
<td>$</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Combo Number</th>
<th>Cost Price</th>
<th>Edit</th>
<th>Remove</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cheesy Combo</td>
<td>33001</td>
<td>0.52</td>
<td>$</td>
</tr>
<tr>
<td>Classic Combo</td>
<td>3344555</td>
<td>0.69</td>
<td>$</td>
</tr>
<tr>
<td>Grilled Cheese Combo</td>
<td>443334</td>
<td>3.037</td>
<td>$</td>
</tr>
</tbody>
</table>

Figure 29. List of Products/Combos

4.16 Specials Option on User’s Menu

This option will display all the existing items on the database that were bought on a special price. Also the products that use these items are going to be displayed as a special. So, in this way GRIPED can help the user to decide when it is the right time to set a special price of a product, and what can be included on it (Figure 30).
Welcome to GRIPED
Gerardo Staff

<table>
<thead>
<tr>
<th>Category</th>
<th>Name of Item</th>
<th>Item Number</th>
<th>Cost Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>Beverages</td>
<td>BIB Coca Cola 32 oz</td>
<td>1000001</td>
<td>.256</td>
</tr>
<tr>
<td>Dairy</td>
<td>Yellow Cheese 2 Slices</td>
<td>8000001</td>
<td>.25</td>
</tr>
<tr>
<td>Meats</td>
<td>Hamburger Patty 8 oz</td>
<td>3000001</td>
<td>.750</td>
</tr>
<tr>
<td>Meats</td>
<td>Chicken Fried Steak</td>
<td>3000002</td>
<td>.92</td>
</tr>
<tr>
<td>Vegetables</td>
<td>Tomato Sliced 2 oz</td>
<td>5000002</td>
<td>.1805</td>
</tr>
<tr>
<td>Vegetables</td>
<td>Green Leaf</td>
<td>5000003</td>
<td>.041</td>
</tr>
</tbody>
</table>

Figure 30. Items and Products on Special

4.17 Logout Option on User’s Menu

This option will terminate the user’s session and redirect them to the initial login page (Figure 4).
5.1 Conclusions

Pricing out products for any given restaurant menu has been a major problem and a time consuming task for medium and small size restaurants. Throughout the years of experience I have with restaurants and the food industry, I have seen several tools that intended to solve this problem. Unfortunately, all of these are very complex, complicated to operate and full of functions that are not necessary for medium and small size restaurants. The inconvenience and complexity of these systems obligate restaurants' owners to abandon on these kinds of tools, making them return to the old fashioned and time consuming pricing methods using paper and pencil.

Thinking of all the inefficient and complex software that there are in the market, and keeping in mind the small food industry business necessities, GRIPED was developed using ideas from real scenarios and based on their specific necessities and problems. GRIPED was designed to solve the problem of pricing the products out for restaurant owners, reducing considerably the margin of error on real cost prices so customers and owners would be
benefited by getting the right prices every day. So as its name stated, GRIPED (Get the Right Price Everyday) will be the beginning of a new way to price and control prices on food industry.

5.2 Proposed Future Development

1. Develop a third main menu on the system for servers. This third menu basically is going to have an electronic version of the paper menus. So everything that a client could order from the paper menu will be on the electronic one that the servers will have. Once the server takes the order from the client, this one is submitted through the system and a display monitor on the kitchen alerts to the chefs of the new incoming order almost in real time. In that way the chefs can start the dish as soon as the server submits the order and reduce the waiting time for the customer on the table.

2. Add a new option on the administrator menu called “Shopping List”. This option will start creating a list of the products that administrator needs to buy. This list will be according to the orders that servers submitted
during the day or week. So all the items or products they use on their orders are going to be discounted from the main inventory and when they reach a reorder level, the product will be displayed on the shopping list.
APPENDIX

PROGRAM
<div id="header"
<?php
session_start();
?>
<div style='float: left;
> Welcome to GRIPED</div>
<?php
echo "\n\n\tdate("I F Y ") . \t date("g:i a"). "\n</div>";
echo "\n\tdate: none;\"$_SESSION\[name\]\n</div>";
</div>

<?php
include "header.inc";
include "menu.php";
$temp=$_GET['start'];
if($temp=="
{
$start=0;
$back=0;
}
else{
$start=$temp;
$back = $start - $limit;
}
$limit=10;
$sql="SELECT * FROM provider";
$rs=mysql_query($sql);
echo mysql_error();
$val=mysql_num_rows($rs);

bgcolor="#f5f5f5"
</div id="center">";
echo \n<h3>VIEW PROVIDERS</h3>\n<table cellspacing="1"
"Provider"<td>;
echo \n"Sales Representative"<td>;
```php
$inc = 0;
while ($rows = mysql_fetch_array($result)) {
    $inc = $inc + 1;
    if($bgcolor='#f1f1f1')
        echo '<td bgcolor="#f1f1f1" class="displayer">$rows[provider]</td>
    else{
        echo '<td bgcolor="#ffffff" class="displayer">$rows[provider]</td>
    }
    $start = $start + $limit;
    if($back > 0){
        $back = $back - 10;
    }
    if($start == $val) {
        $start = $val - $limit;
    }
    if($start == 10 && $back==0) {
        if($val > 10) {
            echo '<a href=viewproviders.php?start=$start>next</a>";
        }
    }
    else{
        if($start > $val){
            echo '<a href=viewproviders.php?start=$back>previous</a>";
        } else{
            echo '<a href=viewproviders.php?start=$back>previous</a>". ". " <a href=viewproviders.php?start=$start>next</a>";
        }
    }
    echo '</div>";
}?
</body>
</html>
```
Adduser.php

```php
<?php
session_start();

//if($_SESSION["username"] == ""){
//header ("Location: login.php");
//}
?

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html40/loose.dtd">
<html>
<head>
<title>Create User</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<link rel="stylesheet" type="text/css" href="griped.css" title="format">
<script src="griped.js" type="text/javascript"></script>
</head>

<body>

<form action="insert_user.php" method="post" onSubmit="return validate()">
<table border="0" cellspacing="0" cellpadding="0" width=160 bgcolor="#EEEEEE">
<tr>
<td><span style="color:#FFOOOO;">Are required fields.</span></td>
</tr>
<tr>
<td><b>Username:</b></td>
<td><input type="text" name="username" size=25></td>
</tr>
<tr>
<td><b>First Name:</b></td>
<td><input type="text" name="Fname" size=25></td>
</tr>
<tr>
<td><b>Last Name:</b></td>
<td><input type="text" name="Lname" size=25></td>
</tr>
<tr>
<td><b>Password:</b></td>
<td><input type="password" name="password" size=25></td>
</tr>
<tr>
<td><b>Password Confirmation:</b></td>
<td><input type="password" name="password_confirmation" size=25></td>
</tr>
<tr>
<td><b>Email Address:</b></td>
<td><input type="text" name="email" size=25></td>
</tr>
<tr>
<td><b>Telephone Number:</b></td>
<td><input type="text" name="phone" size=25></td>
</tr>
<tr>
<td><b>Address:</b></td>
<td><input type="text" name="address" size=25></td>
</tr>
<tr>
<td><b>City:</b></td>
<td><input type="text" name="city" size=25></td>
</tr>
<tr>
<td><b>State:</b></td>
<td><input type="text" name="state" size=25></td>
</tr>
<tr>
<td><b>Zip Code:</b></td>
<td><input type="text" name="zip_code" size=25></td>
</tr>
<tr>
<td><b>Country:</b></td>
<td><input type="text" name="country" size=25></td>
</tr>
<tr>
<td><b>Username is unique. Please Check if user already exists:</b></td>
<td><input type="hidden" name="exist_user" value=""></td>
</tr>
<tr>
<td colspan="2"><input type="submit" value="Submit"></td>
</tr>
</table>
</form>
```

```html
66
```
<?php
Createacombo.php
</body>
</html>

Createacombo.php
session_start();
if($_SESSION['username'] == ''){
    header ('Location: login.php');
    exit;
}
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html40/loose.dtd">
<html>
<head>
<title>Create a New Combo</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<link rel='stylesheet' type='text/css' href='style/griped.css' title='format'>
<script src='griped.js' type='text/javascript'></script>
</head>
<body>
<?php
    $temp = $_GET['test'];
    $cname=$_POST['cname'];
    $num=$_POST['combo_id'];
    $products=$_POST['products'];
    $price=$_POST['price'];
    $val=1;
    include 'config.php';
    $temp=0;
?>

<h3>CREATE A COMBO</h3>
<form name="combo" action="createcombo.php" method="post" onsubmit="return cvalidate()">
<table>
    <tr>
        <td><b>Description</b></td>
        <td><b>Combo Number</b></td>
        <td><b>Price</b></td>
    </tr>
    <tr>
        <td><input type="text" name="cname" size="40"></td>
        <td><input type="text" name="combo_id" size="20"></td>
        <td><input type="text" name="price" size="11" readonly><input type="hidden" name="products"></td>
    </tr>
    <tr>
        <td colspan="3"><input type="submit" name="submit" value="Save"></td>
    </tr>
</table>
</form>

<?php
    include "header.inc";
    include "menu.php";
?>
</div id="center">
<h3>CREATE A COMBO</h3>
<form name="combo" action="createcombo.php" method="post" onsubmit="return cvalidate()">
<table>
    <tr>
        <td><b>Description</b></td>
        <td><b>Combo Number</b></td>
        <td><b>Price</b></td>
    </tr>
    <tr>
        <td><input type="text" name="cname" size="40"></td>
        <td><input type="text" name="combo_id" size="20"></td>
        <td><input type="text" name="price" size="11" readonly><input type="hidden" name="products"></td>
    </tr>
    <tr>
        <td colspan="3"><input type="submit" name="submit" value="Save"></td>
    </tr>
</table>
</form>

<?php
    $query = mysql_query("SELECT * from product");
    $val = mysql_num_rows($query);
    $maincount = 0;
    $columns = 5;
    $qry = "SELECT * FROM product";
    $sq = mysql_query($qry);
    $i=0;
    echo "<table>
    <tr><td colspan="{$columns}"></td><h3>Products</h3></tr>
    ";
    while($row = mysql_fetch_array($sq)){
        $p = $row[price];
        echo "<td>
        $p
        <br>
        ";
        echo "<input type="checkbox" name="combo" value="" . $row[product_number] . "" onClick="checkMade()">";
    };
</td>
</table>
</form>
</div id="center">
<h3>CREATE A COMBO</h3>
<form name="combo" action="createcombo.php" method="post" onsubmit="return cvalidate()">
<table>
    <tr>
        <td><b>Description</b></td>
        <td><b>Combo Number</b></td>
        <td><b>Price</b></td>
    </tr>
    <tr>
        <td><input type="text" name="cname" size="40"></td>
        <td><input type="text" name="combo_id" size="20"></td>
        <td><input type="text" name="price" size="11" readonly><input type="hidden" name="products"></td>
    </tr>
    <tr>
        <td colspan="3"><input type="submit" name="submit" value="Save"></td>
    </tr>
</table>
</form>

<?php
    include "header.inc";
    include "menu.php";
?>
</div id="center">
<?php
$qry = mysql_query("SELECT * FROM combo WHERE combo_number='$combo'");
if($rec=mysql_fetch_array($qry))
{
    echo "Item number already in use by another item.";
}
else{
    if($_POST['submit']){
        $qry="INSERT INTO combo(combo_name, combo_number, products, Price) VALUES ('$name', '$num', '$products', '$price');
        mysql_query($qry);
        echo "Combo Has Been added";
    }
}
?>
</div>
</body>
</html>

insert_user.php

<?php
session_start();

if($_SESSION['username'] == ""){
    header ("Location: login.php");
}

$user = $_POST['userID'];
$name = $_POST['name'];
$number = $_POST['number'];
$password = $_POST['password'];
$admin = $_POST['admin'];
$update = $_POST['update'];
$insert = $_POST['insert'];
$id = $_POST['id'];
$pid = $_POST['pid'];
<?php
require "config.php";
include "header.inc";
include "menu.php";
?>

<?php
if($update == "user"){
  echo "<h3>User Update</h3>";
  echo "User info has been Updated Successful <br>";
  if($adChk == "Y"){
    $utype = "Admin";
  } else {
    $utype = "Staff";
  }

  $qry = "UPDATE users SET userlD='$user', Fname='$Fname', Lname='$Lname', Password='$passwd', adminCheck='$adChk', user_type='$utype' WHERE id='$id';";
  mysql_query($qry);
  echo '<META HTTP-EQUIV="Refresh" Content="2; URL=viewusers.php">';
}
if($insert == "provider"){
  $provider=str_replace("&", "&amp;", $provider);
  mysql_query("INSERT INTO provider VALUES('','$provider','$salesrep','$phone','$fax','$mobile','$email','$website')");
  echo "Provider Added";
} else if($update == "prov"){
  /update provider
  echo "User Info has been Updated Successful <br>";
  $qry="UPDATE provider SET provider='$provider', sales_representative='$salesrep', sales_representative_phone='$phone', sales_representative_fax='$fax', sales_representative_mobile='$mobile', sales_representative_email='$email', provider_website='$website' WHERE id='$id';";
  mysql_query($qry);
  echo '<META HTTP-EQUIV="Refresh" Content="2; URL=viewproviders.php">';
}
else if($insert == "user"){
  if($chkuser=mysql_fetch_array(mysql_query("SELECT userID FROM users where userID='$user'"))){
    echo "Username ". " <b>$user</b> ". " Already exist please choose another username";
    echo '<META HTTP-EQUIV="Refresh" Content="2; URL=adduser.php?val=user">';
  } else{

  }
}?>
if($adChk == "Y"){
    $utype = "Admin";
} else {
    $utype = "Staff";
}

mysql_query("INSERT INTO users VALUES( ', '$user', '$Fname', '$Lname', '$password', '$adChk', '$utype');
    echo "<br><br>";
    echo "User Added";
}
</div>
</body>
</html>

Inv2.php

<?php
    require "config.php";
    session_start();

    if($_SESSION['username'] == ""){
        header ("Location: login.php");
    }

?>
</div>
</body>
</html>

<?php
    $iname=$_POST['pname'];
    $item=$_POST['inum'];
    $invoice=$_POST['invnum'];
    $price=$_POST['price'];
    $provider=$_POST['prov'];
    $pack=$_POST['pack'];
    $unit=$_POST['unit'];
    $category=$_POST['category'];
    $subcategory=$_POST['sub'];
    $salesrep=$_POST['sale'];
    $phone=$_POST['pphone'];
    $fax=$_POST['fax'];
    $mobile=$_POST['mobile'];
    $email=$_POST['email'];
    $website=$_POST['web'];
    $comment=$_POST['comment'];
    $display=$_POST['display'];
    $locate=$_POST['search'];
    $items = $_POST['items'];
    $additems = $_POST['adds'];
    $specials = $_POST['special_box'];
    $id=$_POST['id'];
?>
<body>
<?php
include "header.inc";
include "menu.php";
?>

<div id="center">

<?php
if($locate=="update"){
    echo "<h3>Item Update</h3>";
    echo "Your Item has been Updated Successful <br>";

    if($_POST['submit']){
        $qry="UPDATE item SET item_name="$iname", Pack="$pack", Unit="$unit", item_number="$item", invoice_number="$invoice", Category="$category", SubCategory="$subcategory", Provider="$provider", Comments="$comment", Price="$price", special="$specials" WHERE id="$id"; mysql_query($qry);
    }
}
else if($locate=="newprod"){
    $pname = $_POST['pname'];
    $pnum = $_POST['product_id'];
    $price = $_POST['price'];
    $specials = $_POST['special_box'];

    echo "<h3>Product Update</h3>";
    echo "Your Product has been Updated <br>";
    if($_POST['submit']){
        //Update the database with new checkboxes
        $qry="UPDATE product SET product_name="$pname", product_number="$pnum", items="$items", price="$price" where product_number="$pnum"; mysql_query($qry);
    }
}
else if($locate=="newcombo"){
    $cname = $_POST['cname'];
    $cnum = $_POST['combo_id'];
    $cprice = $_POST['price'];
    $products = $_POST['products'];

    echo "<h3>Combo Update</h3>";
    echo "Your Combo has been Updated <br>";
    if($_POST['submit']){
        //Update the database with new checkboxes
        $qry="UPDATE combo set combo_name="$cname", combo_number="$cnum", products="$products", Price="$cprice" where combo_number="$cnum"; mysql_query($qry);
    }
}
else {
    $qry = mysql_query("SELECT * FROM item WHERE item_number="$item";
    if($rec=mysql_fetch_array($qry)) {
        echo "Item number already in use by another item.;"
        echo "<META HTTP-EQUIV="Refresh" Content="2; URL=Itementry.php">;"
    } else {
        echo "Your New Entry Has Been Created;"
        echo "<br>";
        // add price, name, item_number check here
}
</div>

72
```php
mysql_query("INSERT INTO item VALUES("$iname", "$pack", "$unit", "$item", "$invoice", "$category", "$subcategory", "$provider", "$comment", "$price", "$specials")");
$query="SELECT * FROM item";
.rs=mysql_query($query);
$num = mysql_num_rows($rs);
echo mysql_error();

while($rows = mysql_fetch_array($rs)){
    echo "<tr>
    " . $rows[item_name] . "</td>
    " . $rows[Pack] . "</td>
    " . $rows[Unit] . "</td>
    " . $rows[item_number] . "</td>
    " . $rows[Invoice_number] . "</td>
    " . $rows[Category] . "</td>
    " . $rows[Subcategory] . "</td>
    " . $rows[Provider] . "</td>
    " . $rows[Comments] . "</td>
    " . $rows[Price] . "</td>
    " . $rows[Special] . "</td>
    </tr>
}

echo "</table>";
?>
</div>
</body>
</html>

Inv.php

<?php
    session_start();
    if($_SESSION["username"] == ""){
        header ("Location: login.php");
    }
?>

<html>
<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<script type="text/javascript">
    function go(usr){
        if (usr=="b1") {
            location = "inventoryedit.php"
        } else {
```
else
if(usr="b2")
{
  location = "inv.php?loc=view"
}
else if(usr="b3")
{
  location = "#"
}
</script>
</head>
<?php
include "config.php";
$name=$_POST["lname"];
$desc=$_POST["desc"];
$qty=$_POST["qty"];
$price=$_POST["price"];
$uom=$_POST["measure"];
$locate=$_GET["loc"];
?>
<body>
<?php
if($locate=="view"){
  $bgcolor="#ffffff"
  echo "<table>
  <tr>
  <td Item Name</td>
  <td Description</td>
  <td Quantiy</td>
  <td UOM</td>
  <td Price</td>
  </tr>
  THIS WORKED FINE <br>
  $query="Select * from products";
  $result=mysql_query($query);
  echo mysql_error();
  while($row=mysql_fetch_array($result))
  { if($bgcolor==="#fffff"){$bgcolor="#ffffff";}
    $bgcolor="#fffff"
    echo "<tr >
    <td align=left bgColor=$bgcolor id="title">$row[itemName]<td>
    <td align=left bgColor=$bgcolor id="row[Description]";
    <td align=left bgColor=$bgcolor id="row[Quantity]";
    <td align=left bgColor=$bgcolor id="row[UOM]";
    <td align=left bgColor=$bgcolor id="row[Price]";
    </tr>
  }
  echo "</table>
  }
  else {
  echo "THIS DIDN'T WORK";
  if($name == ""){
    echo "Name is required";
    }
  else
  $sql=mysql_query("INSERT INTO products VALUES(NULL, "$name", "$desc", "$qty", "$price", "$uom")");
}
$query = "SELECT * FROM products";
$rs = mysql_query($query);
echo mysql_error();
bgcolor="#f1f1f1";
echo "<table cellspacing='4' cellpadding='0'>";
echo "<tr>";
echo "<td bgcolor='dfdfdf'>Item Name</td>";
echo "<td bgcolor='dfdfdf'>Description</td>";
echo "<td bgcolor='dfdfdf'>Quantity</td>";
echo "<td bgcolor='dfdfdf'>UOM</td>";
echo "<td bgcolor='dfdfdf'>Price</td></tr>";
while($rows = mysql_fetch_array($rs)){
    if($bgcolor=='#f1f1f1'){$bgcolor='#ffffff';}
    else{$bgcolor="#f1f1f1";} 
    echo "<tr>";
    echo "<td align=left bgcolor=$bgcolor id='title'>$rows[ItemName]</td>";
    echo "<td align=left bgcolor=$bgcolor id='title'>$rows[Description]</td>";
    echo "<td align=center bgcolor=$bgcolor id='title'>$rows[Quantity]</td>";
    echo "<td align=left bgcolor=$bgcolor id='title'>$rows[UOM]</td>";
    echo "<td align=left bgcolor=$bgcolor id='title'>$rows[Price]</td></tr>";
} }
?>
<table><tr><td><input type='button' name='b1' value='Add Inventory' onClick='go("b1")' size='50'></td>
<td><input type='button' name='b2' value='View Inventory' onClick='go("b2")'></td>
<td><input type='button' name='b3' value='Modify Inventory' onClick='go("b3")'></td>
</tr>
</table>
</body>
</html>

Invdel.php

<?php
include "config.php";
session_start();
if($SESSION["username"] == ""){
    header ("Location: login.php");
}
else{
    $del = $_GET['del'];
    $val = $_GET['val'];
    include "header.inc";
    echo "<link rel='stylesheet' type='text/css' href='style/griped.css' title='format'>";
    if ($del == "item"){
        $qry = "select * from product";
        $sq = mysql_query($qry);
        echo "<div id='center'>";
        echo "Successfully Deleted Item";
        echo "</div>";
    }
$sq = mysql_query("DELETE FROM item where id='$val'");
include "menu.php";
}
else if($del == "product"){
    echo "<div id="center">">
    echo "Successfully Deleted Product";
    echo "</div>";
    $sq = mysql_query("DELETE FROM product where id='$val'");
    include "menu.php";
}
else if($del == "combo"){
    echo "<div id="center">">
    echo "Successfully Deleted Combo";
    echo "</div>";
    $sq = mysql_query("DELETE FROM combo where id='$val'");
    include "menu.php";
}
else if($del == "user"){
    echo "<div id="center">">
    echo "Successfully Deleted User";
    echo "</div>";
    $sq = mysql_query("DELETE FROM users where userlD='$val'");
    include "menu.php";
}
else if($del == "provider"){
    echo "<div id="center">">
    echo "Successfully Deleted Provider";
    echo "</div>";
    $sq = mysql_query("DELETE FROM provider where id='$val'");
    include "menu.php";
}
}
?>

Invedit.php

<?php
session_start();
include "config.php";
if($_SESSION["username"] == ""){
    header ("Location: login.php");
}?

<!DOCTYPE HTML PUBLIC "://W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html40/loose.dtd">
<html>
<head>
<title>Access Existing Items</title>
<link rel="stylesheet" type="text/css" href="style/griped.css" title="format">
</head>

<body>

<?php
include "header.inc";
include "menu.php";
?>
<div id="center">
<h3>EXISTING ENTRIES</h3>

<?php
if($_SESSION['adchk'] == "Y"){
    $query = "SELECT * from item order by Category";
    $sql=mysql_query($query);
    echo "<table class="displayer_head" cellspacing='0'">";
    echo "<tr>";
    echo "<td class="displayer_head">Category</td>";
    echo "<td class="displayer_head">Name of Item</td>";
    echo "<td class="displayer_head">Item Number</td>";
    echo "<td class="displayer_head">Cost Price</td>";
    echo "<td class="displayer_head">Edit</td>";
    echo "<td class="displayer_head">Remove</td>";
    echo "</tr>";

    while($rs=mysql_fetch_array($sql)){
        $iname = $rs[item_name];
        $itemnum = $rs[item_number];
        $iprice = $rs[Price];
        if($bgcolor=='#f1f1f1'){$bgcolor="#ffffff"}
        else{
            $bgcolor="#f1f1f1";
        }
        echo "<td bgcolor='$bgcolor'>";
        echo "<a href='update.php?edit=item&amp;val=$iname'><img src='\images/edit.gif' width='141' height='151' border='0' alt='Edit'></a></td>"
        echo "<td bgcolor='$bgcolor'>";
        echo "<a href='invdel.php?del=item&amp;val=$iname'><img src='\images/delete.gif' width='141' height='151' border='0' alt='Remove'></a></td>"
    }
}
}

<?php

if($_SESSION['adchk'] != "Y"){
    $query = "SELECT * from product";
    $sql=mysql_query($query);
    echo "<table class="displayer_head" cellspacing='0'">";
    echo "<tr>";
    echo "<td class="displayer_head">Name of Product</td>";
    echo "<td class="displayer_head">Product Number</td>";
    echo "<td class="displayer_head">Cost Price</td>";
    echo "<td class="displayer_head">Edit</td>";
    echo "<td class="displayer_head">Remove</td>";
    echo "</tr>";

    while($rs=mysql_fetch_array($sql)){
        $pname = $rs[product_name];
        $pid = $rs[product_id];
        $price = $rs[Price];
        if($bgcolor=='#f1f1f1'){$bgcolor="#ffffff"}
        else{
            $bgcolor="#f1f1f1";
        }
        echo "<td bgcolor='$bgcolor'>";
        echo "<a href='update.php?edit=product&amp;val=$pid'><img src='\images/edit.gif' width='141' height='151' border='0' alt='Edit'></a></td>"
    }
}

?>
```html
<?php
$bgcolor = '#f1f1f1';
if($bgcolor == '#f1f1f1') {$bgcolor = '#ffffff';}
else {$bgcolor = '#f1f1f1';}

echo "<table border=0 width=100%>
<tr><td bgcolor="$bgcolor" class="displayer_head" style="padding-top:16px;">Name of Combo</td><td bgcolor="$bgcolor" class="displayer_head">Combo Number</td><td bgcolor="$bgcolor" class="displayer_head">Cost Price</td><td bgcolor="$bgcolor" class="displayer_head">Edit</td><td bgcolor="$bgcolor" class="displayer_head">Remove</td></tr>

while($rs = mysql_fetch_array($sql)) {
    $cname = $rs[combo_name];
    $cnum = $rs[id];
    $price = $rs[Price];
    $cid = $rs[combo_number];
    if($bgcolor == '#f1f1f1') {$bgcolor = '#ffffff';}
    else {$bgcolor = '#f1f1f1';}
    echo "<td bgcolor="$bgcolor" class="displayer">" . $cname . "</td>";
    echo "<td bgcolor="$bgcolor" class="displayer">" . $cnum . "</td>";
    echo "<td bgcolor="$bgcolor" class="displayer">" . $price . "</td>";
    echo "<td bgcolor="$bgcolor" class="displayer" href=update.php?edit=combo&val=$cnum><img src=images/edit.gif width="14" height="15" border="0" alt="Edit"></a></td>";
    echo "<td bgcolor="$bgcolor" class="displayer" href=invdel.php?del=combo&val=$cnum><img src=images/delete.gif width="14" height="15" border="0" alt="Remove"></a></td>";
}
</tr>
</table>

<?php
session_start();
require "config.php";
if($_SESSION["username"] == ""){
    header("Location: login.php");
}

<!DOCTYPE HTML PUBLIC ">-//W3C//DTD HTML 4.01 Transitional//EN" http://www.w3.org/TR/html40/loose.dtd">
<html>
<head>
<title>New Item</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>
<body>
</html>
```
CREATE A NEW ITEM

Item Name:
Pack:
Unit:
Item number:
Invoice Number:
Category:
Subcategory:
Provider:

<?php
$qry = "SELECT * from category order by category";
$sql = mysql_query($qry);
while($row = mysql_fetch_array($sql)){
    echo "<option value='$row[category]'>$row[category]</option>
";
}
?>

<?php
$qry = "SELECT * from provider order by provider";
$sql = mysql_query($qry);
while($row = mysql_fetch_array($sql)){
    echo "<option value='$row[provider]'>$row[provider]</option>
";
}
?>

<form name="myForm" action="inv2.php" method="post" onSubmit="return validate()"
<tr>
<td align="right"><span style="color:#FF0000;">*</span>Item Name:</td>
<td><input type="text" name="pname" size="20"></td>
</tr>
<tr>
<td align="right">Pack:</td>
<td><input type="text" name="pack" size="20"></td>
</tr>
<tr>
<td align="right">Unit:</td>
<td><input type="text" name="unit" size="20"></td>
</tr>
<tr>
<td align="right"><span style="color:#FF0000;">*</span>Item number:</td>
<td><input type="text" name="inum" size="20"></td>
</tr>
<tr>
<td align="right"><span style="color:#FF0000;">*</span>Invoice Number:</td>
<td><input type="text" name="invnum" size="15"></td>
</tr>
<tr>
<td align="right"><span style="color:#FF0000;">*</span>Category:</td>
<td><?php
$qry = "SELECT * from category order by category";
$sql = mysql_query($qry);
while($row = mysql_fetch_array($sql)){
    echo "<option value='$row[category]'>$row[category]</option>
";
}
?>
</td>
</tr>
<tr>
<td align="right">Subcategory:</td>
<td><input type="text" name="sub" size="20"></td>
</tr>
<tr>
<td align="right"><span style="color:#FF0000;">*</span>Provider:</td>
<td><?php
$qry = "SELECT * from provider order by provider";
$sql = mysql_query($qry);
while($row = mysql_fetch_array($sql)){
    echo "<option value='$row[provider]'>$row[provider]</option>
";
}
?>
</td>
</tr>
</form>
Login.php

<?php
    session_start;
    $log = $_GET['log'];

    if($log=="logout"){ 
        session_unset();
    }

<?IDCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html40/loose.dtd">
<html>
<head>
<title>Login</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<link rel="stylesheet" type="text/css" href="style/griped.css" title="format">
<script src="griped.js" type="text/javascript"></script>
</head>
<body>
    ?php
    include "header.inc";
    ?

    <div id="center">
        <form name="login" action="loginaction.php" method="post" onSubmit="return lvalidate()">
            <table class="login" border="0" cellpadding="4" cellspacing="0" bgcolor="#EEEEEE">
                <tr><td colspan="2"><h3>Welcome to GRIPED</h3>Please log in with your assigned username and password.</td></tr>
                <tr><td width="80">Username:</td><td><input type="text" name="username" value="" size="20"></td></tr>
                <tr><td width="80">Password:</td><td><input type="password" name="password" value="" size="20"></td></tr>
            </table>
        </form>
    </div>
</body>
</html>
<?php
require "config.php";
session_start();
// start session
?

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html40/loose.dtd">
<html>
<head>
<title>GRIPED Home page</title>
<link rel="stylesheet" type="text/css" href="style/griped.css">
</head>
<body>

<?php
if ($_SESSION['Loggedln'] != 'true') {
   $_SESSION['username'] = $_POST['username'];
   $_SESSION['password'] = $_POST['password'];
   $adChk = $_POST['admin'];
   $_SESSION['admin'] = $_POST['admin'];
   $loc = $_GET['chk'];
}
if($_SESSION['username'] != '') {
   if($rec=mysql_fetch_array(mysql_query("SELECT * FROM users WHERE userID="$_SESSION['username']") AND Password = "$_SESSION['password']"))) {
      $_SESSION['Loggedln'] = 'true';
   } else {
      //session_unset();
      echo "<span style="color:#FF0000;"">Wrong Login. Use your correct Userid and Password</span>";
      include "login.php";
      exit();
   }
} else if($_SESSION['username'] == ''){
   echo "<span style="color:#FF0000;">Invalid username and/or password</span>";
   //include "login.php";
   //exit();
}
$name = $_SESSION['username'];

$result = mysql_query("SELECT * from users WHERE userID="$name" ");
if (!$result) {
   echo("<P>Error performing query: " . mysql_error() . "</P>");

</td colspan="2" align="center"><input type="submit" value="Login" name="B1"><input type="reset" value="Clear" name="B2"></td>
</tr>
</table>
</form>
</div>
</body>
</html>

Loginaction.php

<?php
require "config.php";
session_start();
// start session
?>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html40/loose.dtd">
<html>
<head>
<title>GRIPED Home page</title>
<link rel="stylesheet" type="text/css" href="style/griped.css">
</head>
<body>

<?php
if ($_SESSION['Loggedln'] != 'true') {
   $_SESSION['username'] = $_POST['username'];
   $_SESSION['password'] = $_POST['password'];
   $adChk = $_POST['admin'];
   $_SESSION['admin'] = $_POST['admin'];
   $loc = $_GET['chk'];
}
if($_SESSION['username'] != '') {
   if($rec=mysql_fetch_array(mysql_query("SELECT * FROM users WHERE userID="$_SESSION['username']") AND Password = "$_SESSION['password']"))) {
      $_SESSION['Loggedln'] = 'true';
   } else {
      //session_unset();
      echo "<span style="color:#FF0000;"">Wrong Login. Use your correct Userid and Password</span>";
      include "login.php";
      exit();
   }
} else if($_SESSION['username'] == ''){
   echo "<span style="color:#FF0000;">Invalid username and/or password</span>";
   //include "login.php";
   //exit();
}
$name = $_SESSION['username'];

$result = mysql_query("SELECT * from users WHERE userID="$name" ");
if (!$result) {
   echo("<P>Error performing query: " . mysql_error() . "</P>");

</td colspan="2" align="center"><input type="submit" value="Login" name="B1"><input type="reset" value="Clear" name="B2"></td>
</tr>
</table>
</form>
</div>
</body>
</html>

Loginaction.php

<?php
require "config.php";
session_start();
// start session
?>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html40/loose.dtd">
<html>
<head>
<title>GRIPED Home page</title>
<link rel="stylesheet" type="text/css" href="style/griped.css">
</head>
<body>

<?php
if ($_SESSION['Loggedln'] != 'true') {
   $_SESSION['username'] = $_POST['username'];
   $_SESSION['password'] = $_POST['password'];
   $adChk = $_POST['admin'];
   $_SESSION['admin'] = $_POST['admin'];
   $loc = $_GET['chk'];
}
if($_SESSION['username'] != '') {
   if($rec=mysql_fetch_array(mysql_query("SELECT * FROM users WHERE userID="$_SESSION['username']") AND Password = "$_SESSION['password']"))) {
      $_SESSION['Loggedln'] = 'true';
   } else {
      //session_unset();
      echo "<span style="color:#FF0000;"">Wrong Login. Use your correct Userid and Password</span>";
      include "login.php";
      exit();
   }
} else if($_SESSION['username'] == ''){
   echo "<span style="color:#FF0000;">Invalid username and/or password</span>";
   //include "login.php";
   //exit();
}
$name = $_SESSION['username'];

$result = mysql_query("SELECT * from users WHERE userID="$name" ");
if (!$result) {
   echo("<P>Error performing query: " . mysql_error() . "</P>");

</td colspan="2" align="center"><input type="submit" value="Login" name="B1"><input type="reset" value="Clear" name="B2"></td>
</tr>
</table>
</form>
</div>
</body>
</html>

Loginaction.php

<?php
require "config.php";
session_start();
// start session
?>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html40/loose.dtd">
<html>
<head>
<title>GRIPED Home page</title>
<link rel="stylesheet" type="text/css" href="style/griped.css">
</head>
<body>

<?php
if ($_SESSION['Loggedln'] != 'true') {
   $_SESSION['username'] = $_POST['username'];
   $_SESSION['password'] = $_POST['password'];
   $adChk = $_POST['admin'];
   $_SESSION['admin'] = $_POST['admin'];
   $loc = $_GET['chk'];
}
if($_SESSION['username'] != '') {
   if($rec=mysql_fetch_array(mysql_query("SELECT * FROM users WHERE userID="$_SESSION['username']") AND Password = "$_SESSION['password']"))) {
      $_SESSION['Loggedln'] = 'true';
   } else {
      //session_unset();
      echo "<span style="color:#FF0000;"">Wrong Login. Use your correct Userid and Password</span>";
      include "login.php";
      exit();
   }
} else if($_SESSION['username'] == ''){
   echo "<span style="color:#FF0000;">Invalid username and/or password</span>";
   //include "login.php";
   //exit();
}
$name = $_SESSION['username'];

$result = mysql_query("SELECT * from users WHERE userID="$name" ");
if (!$result) {
   echo("<P>Error performing query: " . mysql_error() . "</P>");

</td colspan="2" align="center"><input type="submit" value="Login" name="B1"><input type="reset" value="Clear" name="B2"></td>
</tr>
</table>
</form>
</div>
</body>
</html>
exit();
}
while ( $row = mysql_fetch_array($result) ) {
$user = $row['Fname']. " " . $row['Lname'];
$check = $row['adminCheck'];
$pass = $row['Password'];
$_SESSION['adchk'] = $check;
$_SESSION['user'] = $user;
$_SESSION['name'] = $user;
}

include "header.inc";
include "menu.php";
?>
<div id="center">
<h3>GRIPED</h3>
Welcome to “Get the Right Price Everyday”. GRIPED was designed for the purpose of enabling a systematic approach to the setting of accurate product pricing in the food industry. With GRIPED food industry professionals are able to ensure that inventory is always properly valued as GRIPED ensures that their prices are based on their original purchase price. Specifically, GRIPED includes all of the following functionality:
<ul>
<li>Access to the most recent prices of each item, product and combo</li>
<li>Price comparisons between vendors</li>
<li>Helps staff to decide on when is the right time to set a special price of a product and which products should be included</li>
<li>Manage and control the inventory and prices</li>
<li>Create new menu products based on the purchased items registered in the database</li>
<li>Allows users to calculate and create prices of new items, and products</li>
<li>Allows users to create products and combinations from existing items</li>
</ul>
</div>
</div>
</body>
</html>

Menu.php

<?php
session_start();
?>
<div id="left">
<?php

echo "<table class="menu" >";

<tr><td class="menu" ><a href="/loginaction.php">Home</a></td></tr>
<tr><td class="menu" >
if($_SESSION["adchk"] == "Y")
{
    echo "<tr><td class="menu" ><a href="/itementry.php">New Item</a></td></tr>";
    echo "<tr><td class="menu" ><a href="/invedit.php">View Items</a></td></tr>";
    echo "<tr><td class="menu" ><a href="/adduser.php?val=provider">Add Provider</a></td></tr>";
    echo "<tr><td class="menu" ><a href="/adduser.php?val=user">New User</a></td></tr>";
    echo "<tr><td class="menu" ><a href="/viewproviders.php">View Providers</a></td></tr>";
    echo "<tr><td class="menu" ><a href="/viewusers.php">View Users</a></td></tr>";
    echo "<tr><td class="menu" ><a href="/search.php">Search</a></td></tr>";
}
else
{
    echo "<tr><td class="menu" ><a href="/product.php">New Product</a></td></tr>";
    echo "<tr><td class="menu" ><a href="/createcombo.php">New Combo</a></td></tr>";
    echo "<tr><td class="menu" ><a href="/itementry.php">Products/Combos</a></td></tr>";
}
echo "<tr class="menu" >
    <td class="menu" ><a href="/special.php">Specials</a></td></tr>
    echo "<tr class="menu" >
    <td class="menu" ><a href="/login.php?log=logout">Logout</a></td></tr>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title>GRIPED</title>
</head>
<body>

Welcome to “Get the Right Price Everyday”. GRIPED was designed for the purpose of enabling a systematic approach to the setting of accurate product pricing in the food industry. With GRIPED food industry professionals are able to ensure that inventory is always properly valued as GRIPED ensures that their prices are based on their original purchase price. Specifically, GRIPED includes all of the following functionality:
<ul>
<li>Access to the most recent prices of each item, product and combo</li>
<li>Price comparisons between vendors</li>
<li>Helps staff to decide on when is the right time to set a special price of a product and which products should be included</li>
<li>Manage and control the inventory and prices</li>
<li>Create new menu products based on the purchased items registered in the database</li>
<li>Allows users to calculate and create prices of new items, and products</li>
<li>Allows users to create products and combinations from existing items</li>
</ul>
</body>
</html>
Product.php

<?php
  session_start();
  include "config.php";

  if($_SESSION["username"] == ""){
    header ("Location: login.php");
  }
?>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title>Create a New Product</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<link rel="stylesheet" type="text/css" href="style/griped.css" title="format">
<script src="griped.js" type="text/javascript"></script>
</head>
<body>
<?php
  include "header.inc";
  include "menu.php";
?>

<?php
  $temp = $_GET['test'];
  $pname = $_POST['pname'];
  $num = $_POST['product_id'];
  $item = $_POST['items'];
  $price = $_POST['price'];
  $val = 1;
  $tmp = $_POST['adds'];

  ?
</div id="center">
<h3>CREATE A PRODUCT</h3>
<form name="product" action="product.php" method="post" onsubmit="return pvalidate()">
  <table>
    <tr>
      <td><b>Description</b></td>
      <td><b>Product Number</b></td>
      <td><b>Price</b></td>
    </tr>
    <tr>
      <td><input type="text" name="pname" size="40"></td>
      <td><input type="text" name="product_id" size="20"></td>
      <td><?php echo "<input type=text name=price size=11 readonly><input type=hidden name=items>"; ?></td>
    </tr>
    <tr>
      <td colspan="3"><input type="submit" name="submit" value="Save"></td>
    </tr>
  </table>

<?php
  $query = mysql_query("SELECT * from item");
  $val = mysql_num_rows($query);
  $column = 3;
  $qry = "SELECT * FROM item ORDER BY category";
  $sq = mysql_query($qry);
  $i=0;
  $last = 1;
  $maincount = 0;
  echo "<table><tr><td colspan="3"></td></tr><tr><h3>Items</h3><td><h3></td><tr">
  while($row = mysql_fetch_array($sq)){
if($row[Category] != $last) {
    echo "</tr>" . "<tr><td><b>" . $row[Category] . "</b></td></tr>
    $i=0;
    $maincount=$maincount+1;
    if($i == $columns) {
        echo "</tr>;"
        $i=0;
    }
    if($maincount == $val) {
        echo "</tr>";
        $last = $row[Category];
    }
    echo "</table>";
} else {
    echo "<td>" . $row[Name] . "</td>
    $i=$i+1;
    $maincount=$maincount + 1;
} if($row[Item_number] == "" . $row[Category] . "" . $row[Price] . "" . $row[item_number] . "" . $row[item_name] . "" . $i+1; $maincount=$maincount + 1;}
    echo "<td>" . $row[Name] . "</td>
    $i=$i+1;
    $maincount=$maincount + 1;
} if($maincount == $val) {
    echo "</tr>";
    $last = $row[Category];
} echo "</table>";
<?php
$qry = mysql_query("SELECT * FROM product WHERE product_number="$num"");
if($rec=mysql_fetch_array($qry)) {
    echo "Product number already in use by another product.";
} else {
    //save new product info
    if($POST[submit]) {
        echo "Product Created Successfully";
        $sql = mysql_query("INSERT INTO product(product_name, product_number, items, price) VALUES('$name', '$num', '$item', '$price')")
    }
} echo "</table>";
</div>
</body>
</html>

Search.php

<?php
session_start();
require "config.php";
if($_SESSION[username] == "") {
    header ("Location: login.php");
} else {
    <DOCTYPE HTML PUBLIC "/-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html40/loose.dtd">
    <html>
    <head>
    <title>Search</title>
    <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">" LINK rel="stylesheet" type="text/css" href="style/griped.css" title="format">
    <script src="griped.js" type="text/javascript"></script>
    </head>
    <?php
    $usrID=$_POST['usr'];
<?php
include "header.inc";
include "menu.php";
?>
<?php
echo "<div id="center">
if($_POST['submit']){
if(empty($_POST['pname'])){ $val = False; } else{ $val = True; $item = $_POST['pname']; } if(empty($_POST['inum'])){ $val = False; } else{ $val = True; $itemnum = $_POST['inum']; } if(empty($_POST['invnum'])){ $val = False; } else{ $val = True; $invoice = $_POST['invnum']; } if(empty($_POST['prov'])){ $val = False; } else{ $val = True; $provider = $_POST['prov']; }
if(($val & !$val) & ($val & !$val)) {
    echo "<p style='color:red'>Please enter a value into one of the search fields</p>";
} else{
    if($item != ""){
        $query="SELECT * FROM item WHERE item_name LIKE '$item%' ORDER BY item_name ASC";
    } if($itemnum != ""){
        $query="SELECT * FROM item WHERE item_number = '$itemnum' ORDER BY item_number ASC";
    } if($invoice != ""){
        $query="SELECT * FROM item WHERE Invoice_number = '$invoice' ORDER BY Invoice_number ASC";
    } if($provider != ""){
        $query="SELECT * FROM item WHERE Provider = '$provider'";
    }
    $rs=mysql_query($query);
    $set = mysql_num_rows($rs);
    if($set == 0){
        echo "Record could not be found in search";
    }else{
}
```php
while($rw=mysql_fetch_array($rs)){
    $row_counter++; // Increment the row counter
    if($row_counter % 2 == 0) {
        echo '<tr bgcolor="#87CEFA">' . $rw['item_name'] . '</td>' . $rw['Pack'] . '</td>' . $rw['Unit'] . '</td>' . $rw['item_number'] . '</td>' . $rw['lnvoice_number'] . '</td>' . $rw['Category'] . '</td>' . $rw['Subcategory'] . '</td>' . $rw['Provider'] . '</td>' . $rw['Comments'] . '</td>' . $rw['Price'] . '</td>' . $row_counter % 2 == 0 . '</tr>'; // Close the row
    } else {
        echo '<tr bgcolor="#D0D0D0">' . $rw['item_name'] . '</td>' . $rw['Pack'] . '</td>' . $rw['Unit'] . '</td>' . $rw['item_number'] . '</td>' . $rw['lnvoice_number'] . '</td>' . $rw['Category'] . '</td>' . $rw['Subcategory'] . '</td>' . $rw['Provider'] . '</td>' . $rw['Comments'] . '</td>' . $rw['Price'] . '</td>' . $row_counter % 2 == 0 . '</tr>'; // Close the row
    }
}
```

The above code snippet is designed to output a table in HTML format. It uses `$rs` as the result of a database query, typically fetching records from a database table. The code iterates through each record, printing it in a formatted table structure with alternating row colors for better readability. Each record is printed in a `<tr>` element with specific data attributes (e.g., `item_name`, `Pack`, `Unit`, etc.). The code snippet also includes comments for clarity and to indicate where additional code sections (beginning with `<?php` and ending with `?>`) handle database queries and form handling.
<?php
    session_start();
    include "config.php";

    if($_SESSION['username'] == ""){
        header ("Location: login.php");
    }
?>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html40/loose.dtd">
<html>
<head>
<title>Specials</title>
<LINK rel='stylesheet' type='text/css' href='style/griped.css' title='format'>
</head>
<body>
<?php
    include "header.inc";
    include "menu.php"
?>

<div id="center">
<h3>ITEMS ON SPECIAL</h3>

<?php
    $temp = array();
    $tid = array();
    $query = "SELECT * from item where Special = 1 order by Category";
    $sql=mysql_query($query);
    $size = mysql_num_rows($sql);
    echo "<table class="displayer_head" cellspacing=\"0\">
    echo "<tr>
    echo "<td class=\"displayer_headY\'>Category</td>
    echo "<td class=\"displayer_headY\'>Name of Item</td>
    echo "<td class=\"displayer_headY\'>Item Number</td>
    echo "<td class=\"displayer_headY\'>Cost Price</td>
    echo "</tr>" . "n";
    while($rs=mysql_fetch_array($sql)){
        $iname = $rs[item_name];
        $inum = $rs[id];
        $itnum = $rs[item_number];
        $ispecial = $rs[Special];
        $iprice = $rs[Price];

        $temp[]=array($iname,$inum,$itnum,$iprice);
        if($bgcolor="#f1f1f1"){$bgcolor="#ffffff";}
        else{$bgcolor="#f1f1f1";}
        echo "<tr>
        echo "<td bgcolor="#$bgcolor" class="displayer1">" . $iname . "</td>"
        echo "<td bgcolor="#$bgcolor" class="displayer1">" . $inum . "</td>"
        echo "<td bgcolor="#$bgcolor" class="displayer1">" . $itnum . "</td>"
        echo "<td bgcolor="#$bgcolor" class="displayer1">" . $iprice . "</td>"
    }
    echo "</table>";
</body>
</html>
```php
$qry = 'SELECT * FROM product ORDER BY product_name';
$sql = mysql_query($qry);

while($rw=mysql_fetch_array($sql)){
    $test=explode(' ', trim($rw['items']));
    foreach($temp as $tmp)
    {
        foreach($test as $t)
        {
            if($t == $tmp)
            {
                //echo "<br>" . $rw['id'] . "<br>";
                $id[] = $rw['id'];
            }
        }  //else
        //query = "SELECT * FROM product WHERE items LIKE '%$t%';
    }  ///
}

echo "<br><br><h3>PRODUCTS ON SPECIAL</h3><table cellspacing="0" cellspacing="0" class="displayer_head">
```
```
createcombo.php

<?php
  session_start();
  if($_SESSION['username'] == ''){
    header ('Location: login.php');
  }
?>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html40loomose.dtd">
<html>
<head>
<title>Create a New Combo</title>
<meta http-equiv='Content-Type' content='text/html; charset=iso-8859-1'>
<link rel='stylesheet' type='text/css' href='style/griped.css' title='format'>
<script src='griped.js' type='text/javascript'></script>
</head>

<?php
  $temp = $_GET['test'];
  $cname = $_POST['cname'];
  $num = $_POST['combo_id'];
  $products = $_POST['products'];
  $price = $_POST['price'];
  $val = 1;
  include "config.php";
  $temp = 0;
?>

<body>
  <?php
    include "header.inc";
    include "menu.php";
  ?>

  <div id='center'>
    <h3>CLOSE A COMBO</h3>
    <form name='combo' action='createcombo.php' method='post' onSubmit='return cvalidate()'>
      <table>
        <tr>
          <td>Description</td>
          <td>Combo Number</td>
          <td>Price</td>
        </tr>
        <tr>
          <td><input type='text' name='cname' size='40'></td>
          <td><input type='text' name='combo_id' size='20'></td>
          <td><input type='text' name='price' size='11' readonly><input type='hidden' name='products'></td>
        </tr>
      </table>
      <tr>
        <td colspan='3'><input type='submit' name='submit' value='Save'></td>
      </tr>
    </form>
  </div>
</body>

<?php
  $query = mysql_query("SELECT * from product");
  $val = mysql_num_rows($query);
  $maincount = 0;
  $scolumns = 5;

89


```php
$qry = "SELECT * FROM product";
$sql = mysql_query($qry);
$i=0;
while($row = mysql_fetch_array($sql)){
    $prc = $row[price];
    echo "<input type='hidden' name='prc' value='$prc'>";
    $i=$i+1;
    $maincount=$maincount +1;
    if($i == $columns){
        echo "</tr><tr>";
        $i=0;
    }
    if($maincount == $val){
        echo "</tr>";
    }
    echo "<td>.<input type='Y' name='Y' value='$row[product_number]' onClick='checkMade()'>";
    $i=$i+1;
}
if($maincount == $val){
    echo "</tr>";
}
echo "</table>";
?>
</td>
</table>
</form>

```
<?php
require "config.php";
$edit=$_GET['edit'];
$srch=$_GET['var'];
$tmp=S_POST['search'][j];
$tm=$GET['sta'];
$tm2=$GET['mod'][j];
if($tm == "")
{
    $start=0;
    $back=0;
}
else{
    $start=$tmp;
    $back = $start - $limit;
}
$limit=10;
?>
<body>
<?php
include "header.inc";
include "menu.php";
?>
<div id="center">
<?php
if($edit == "item"){
    echo "<h3>ITEM EDIT</h3>";
    echo "<span style="color:#FF0000;">* </span>Indicates a required field";
    $sql="SELECT * from item where id = '$srch'";
    $rs=mysql_query($sql);
    echo mysql_error();
    echo "<form name="myForm" action="inv2.php" method="post" onSubmit="return validate()" >";
    echo "<table cellspacing="5" cellpadding="" >";
    echo "<tr>";
    echo "<td align="right">Category</td>";
    echo "<td><input type="text" name="categ" value=""size="20"'></td>";
    echo "</tr>";
    echo "<tr>";
    echo "<td align="right">Provider</td>";
    echo "<td><input type="text" name="prov" value=""size="20"'></td>";
    echo "</tr>";
    echo "<tr>";
    echo "<td align="right">Pack</td>";
    echo "<td><input type="text" name="pack" value=""size="20"'></td>";
    echo "</tr>";
    echo "<tr>";
    echo "<td align="right">Unit</td>";
    echo "<td><input type="text" name="unit" value=""size="20"'></td>";
    echo "</tr>";
    echo "<tr>";
    echo "<td align="right">Item Number</td>";
    echo "<td><input type="text" name="item_num" value=""size="20"'></td>";
    echo "</tr>";
    echo "<tr>";
    echo "<td align="right">Invoice Number</td>";
    echo "<td><input type="text" name="inv_num" value=""size="15"'></td>";
    echo "</tr>";
    echo "</form>";
}?>
</div>
</body>
```php
$chk = 0;
if($rows[Special] == 1){
    $chk = "checked";
    
} else{
    $chk = "";
}

if($rows[Special] == 1){
    $chk = "checked";
    
} else{
    $chk = "";
}
```

```html
<form action="" method="POST"> 
<table border="1"> 
<tr><td><input type="hidden" name="id" value="$row[id]"/></td></tr>
<tr><td><input type="hidden" name="search" value=""/></td></tr>
<tr><td><input type="hidden" name="idt" value="$rows[id]"/></td></tr>
<tr><td><input type="submit" name="submit" value="Save"/></td><td><input type="reset" name="Reset" value=""/></td></tr>
</table>
</form>
```
} // End of Item Edit
//Start of the Product Section
else if($edit=='product')
{
$sql="SELECT * from product where id = "$srch";
$rs=mysql_query($sql);
echo mysql_error();
echo "<h3>Update Product</h3>
<form name="product" action="inv2.php1" method="post" onSubmit="return pvalidate()">
<table>
while($row=mysql_fetch_array($rs)){
$items = $row[items];
$stuff = explode("", $items);
$len = count($stuff); //count number of items associate with products

echo "<tr><td><table><tr><td
echo "$p
$query
</td><td>
$rw[product_name] $rows[product_number] readonly
size="20"">
</td>
</tr><td><input type="hidden" name="search" value="newprod"></td>
</tr><td><input type="hidden" name="items" value="newprod"></td>
</tr></table></td>
</tr><td>
</td></tr></table>
</form>

$qry = "SELECT * FROM item GROUP BY category ASC";
$query = mysql_num_row($query);
$columns = 3;
$value = $val / $columns;
$qry = "SELECT * FROM item ORDER BY category";
$sql = mysql_query($qry);
$start=0;
$maincount = 0;
$last = 1;
$i = 0;
$p = 0;
echo "<table><thead><tr valign="center">Category</tr><tr valign="center">Product Number</tr><tr valign="center">Description</tr></thead>
</table><tr class="table">
while($row=mysql_fetch_array($sql)){

if($row[Category] !=$last)
{
    echo "<tr><td valign="center"
    .n" ."<tr><td>
    $row[Category] ."</td></tr><tr">
    "$i=0;
}

$price = $row[Price];
if(trim($stuff[$p]) == $row[item_number]){
    $chk = "checked";
    $val = $val + $row[item_number] ."";
    $p = $p + 1;
}
else{
    $chk ="";
}
echo "<td><input type="checkbox" name="adds" $chk value="" . $row[item_number] . ""
onClick="(check)()"" . $row[item_name];
echo "<input type="hidden" name="prct" value="" . $price . ""/>
" . "$i;" . "$i = $i + 1;
if($i == $columns){
    echo "</tr><tr valign="center"
    .n" . "<tr>";
$i=0;

// Combo Section Start
else if ($edit == "combo")
{
    $sql="SELECT * from combo where id = "$srch";
    $rs=mysql_query($sql);
    echo mysql_error();
    echo "<h3>Update Combo</h3>
    <form name=combo action=inv2.php method=post onSubmit=return cvalidate()Y return cValidate()Y>
    <table>
        while($rows=mysql_fetch_array($rs)){
            Sproducts = $rows[products];
            $stuff = explode(",", $products);
            $len = count($stuff);
            // count number of items associate with products
            echo "<tr>
            " . $i . ">
            echo "<td>
            " . $i . ">
            echo "<input type=text name=cname value="$rows[combo_name]" size=40"></td>
            echo "<td"></td>
            echo "<td><input type=text name=combo_id readonly value="$rows[combo_number]" size=20"></td>
            echo "<td><input type=hidden name=search value=newcombo"></td>
            echo "<td><input type=text name=price readonly value="$rows[Price]" size=11"></td>
            echo "<input type=hidden value="$pval" name=products"></td>
            </tr>
        }" . $i . ">
        echo "<tr>
        " . $i . ">
        echo "$query = mysql_query("SELECT * from product");
        $val = mysql_num_rows($query);
        $columns = 5;
        $value = $val / $columns;
        $qry = "SELECT * FROM product";
        $sql = mysql_query($qry);
        $start=0;
        $i = 0;

49
<?php

$table = '<table>
<tr><td colspan=5>Products</td></tr>';

while($row=mysql_fetch_array($sql))
{
    $price = $row['price'];

    if(trim($stuff[$i]) == $row['product_number']){
        $chk = "checked";
        $val = $val . $row['product_number'] . "," . $i + 1;
    } else{
        $chk = "";
    }

    echo "<td>
        <input type="hidden" name="prc" value="Stprice">
        <input type="checkbox" name="combo" value="" onClick="checkMade()">
        <br>
        $row['product_name']
    </td>

    $i = $i + 1;
}

if($i == $columns){
    echo "</tr>
    <br>
    <tr>
        $start=0;
    }
}

echo "</table>
</td>
</tr>
</table>
</form>";

$total = $total + $val;
if($total == $columns){
    echo "</tr>
    <br>
    <tr>
        $start=0;
    }
}

while($row=mysql_fetch_array($sql)){
    echo "<td width='160'></td>
    <td><b>User Name:</b></td>
    <td><input type="text" name="Username"></td>
    <td><b>First Name:</b></td>
    <td><input type="text" name="Fname"></td>
    <td><b>Last Name:</b></td>
    <td><input type="text" name="Lname"></td>
    ";
}
?>
Provider:</td>
</tr>

else if($edit="provider"){
    echo "<h3>UPDATE A Provider</h3>
    $val=$GET['val'];
    $qry = "Select * from provider where id = $val";
    $sql = mysql_query($qry);
    while($row=mysql_fetch_array($sql)){
        echo "<table bgcolor="#EEEEE">");
            echo "<tr>");
                echo "<td width="160">$row[id]</td>");
                echo "<td><input type="checkbox" value="$row[id]" name="admin">Check if user will be an Administrator; 
        </td>");
        echo "<tr>");
    echo "<td width="160">$row[Provider]");
        echo "<td><input type="text" name="prov" value="$row[provider]" size="25"></td>");
        echo "<tr>");
    echo "<td><input type="text" name="sale" value="$row[sales_representative]" size="30"></td>");
    echo "<tr>");
    echo "<td><input type="text" name="pphone" value="$row[provider_phone]" size="30"></td>");
    echo "</form>");
}
}

</table>
</form>
</body>
</html>
```
$lo$.php

<?php

$lo$.php

<html>
<head>
<title>Modify Inventory</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<script type="text/javascript">
function goURL()
{
x=document.getElementById("MySelect")
if(x.options[x.selectedIndex].text="Search")
{
}
}
</script>
</head>
<body>
Modify Inventory
<form>
<table>
<tr><td><select id="MySelect">
<option value="Search">Search<option>
<option value="Update">Update<option>
<option value="Delete">Delete<option>
</select></td>
</tr>
</table>
</form>
</body>
</html>

viewusers.php

<?php

```
session_start();
include "config.php";
if($_SESSION['username'] == ''){
    header ('Location: login.php');
}

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html><head><title>View Users</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<LINK rel="stylesheet" type="text/css" href="style/griped.css" title="format"></head><body>
<?php
include "header.inc";
include "menu.php";
$Stem=$_GET['start'];
if($temp == ''){
    $start=0;
    $back=0;
}else{
    $start=$temp;
    $back = $start - $limit;
}
$limit=10;
$sql="SELECT * FROM users";
$rs=mysql_query($sql);
echo mysql_error();
$val=mysql_num_rows($rs);
$bgcolor="#f1f1f1";
echo "<div id="center">"
echo "<h3>VIEW USERS</h3>
echo "<table>
tr;
"<td bgcolor="#f1f1f1" class="displayer_head">User Name</td>
"<td bgcolor="#f1f1f1" class="displayer_head">First Name</td>
"<td bgcolor="#f1f1f1" class="displayer_head">Last Name</td>
"<td bgcolor="#f1f1f1" class="displayer_head">User Type</td>
"<td bgcolor="#f1f1f1" class="displayer_head">Edit</td>
"<td bgcolor="#f1f1f1" class="displayer_head">Remove</td>
"<tr>"
for ($inc = 0; $inc < $val; $inc++) {
    $result = mysql_query("SELECT * FROM users LIMIT $start, $limit")
    or die(mysql_error());
    $row = mysql_fetch_array($result);
    $uname=$row['user_name'];
    $inc = $inc + 1;
    if($bgcolor == '#f1f1f1'){$bgcolor = '#ffffff';}
    else{$bgcolor = '#f1f1f1';}
    echo "<td bgcolor="#$bgcolor" class="displayer">$row['user_name']</td>
    echo "<td bgcolor="#$bgcolor" class="displayer">$row['first_name']</td>
    echo "<td bgcolor="#$bgcolor" class="displayer">$row['last_name']</td>
    echo "<td bgcolor="#$bgcolor" class="displayer">$row['user_type']</td>
    echo "<td><a href="update.php?edit=user.FALSE;val=$uname"><img src="images/edit.gif" width="141" height="15" border="0"></a></td>
    echo "<td><a href="delete.php?name=$uname"</td>
"<tr>";
login.php

```php
<?php
session_start();
$log = $_GET['log'];

if($log=="logout"){
    session_unset();
}
?>

<!DOCTYPE HTML PUBLIC "+//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title>Login</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<LINK rel="stylesheet" type="text/css" href="style/griped.css" title="format">
<script src="griped.js" type="text/javascript"></script>
</head>

<body>

<?php
include "header.inc";
?>

<form name="login" action="loginaction.php" method="post" onSubmit="return lvalidate()">
<table class="login" border="0" cellspacing="0" cellpadding="4" bgcolor="#EEEEEE">
<tr>
<td><a href="adduser.php"><img src=Y'images/delete.gif' width="14" height="15" border="0"></a></td>
</tr>
</table>
</form>
```

99
<td colspan="2"><h3>Welcome to GRIPED</h3>Please log in with your assigned username and password.<br/></td>
<tr><td width="80">Username:</td><td><input type="text" name="username" value="" size="20"></td></tr>
<tr><td width="80">Password:</td><td><input type="password" name="password" value="" size="20"></td></tr>
<tr><td colspan="2" align="center"><input type="submit" value="Login" name="B1"><input type="reset" value="Clear" name="B2"/></td></tr>
</form>
</div>

</body>
</html>
return false;
} if(fname =="")
{ alert("You must enter a First Name");
 return false;
} if(lname =="")
{ alert("You must enter a Last Name");
 return false;
} return true;
}

//login validation
function lvalidate()
{
 x=document.login;
 uname=x.username.value;
 pass=x.password.value;
 okSubmit="True";
 if(uname =="")
 { alert("You must enter a username");
 return false;
 } if(pass =="")
 { alert("You must enter a password");
 return false;
 } return true;
}

//checkbox check
function check()
{
 adds=document.forms[0].adds
 pname=document.forms[0].pname
 items=document.forms[0].items
 price=document.forms[0].price
 prc=document.forms[0].prc
 txt=""
 tmp=""
 for (i=0;i<adds.length;++ i)
 { if (adds[i].checked)
 { txt=txt + adds[i].value + "," 
 tmp = parseFloat(tmp + parseFloat(prc[i].value));
 } }
 items.value= txt
 price.value=Math.round(tmp*1000) / 1000
}
function checkMade()
{
    combo=document.forms[0].combo
    products=document.forms[0].products
    price=document.forms[0].price
    prc=document.forms[0].prc
    txt=""
    tmp=""
    for (i=0;i<combo.length;++ i)
    {
        if (combo[i].checked)
        {
            txt=txt + combo[i].value
            tmp = parseFloat(tmp + parseFloat(prc[i].value));
        }
    }
    products.value= txt
    price.value=Math.round(tmp * 1000) / 1000
}

function checkMade2()
{
    combo=document.forms[0].combo
    products=document.forms[0].products
    price=document.forms[0].price
    prc=document.forms[0].prc
    txt=""
    tmp=""
    for (i=0;i<combo.length;++ i)
    {
        if (combo[i].checked)
        {
            txt=txt + combo[i].value
            tmp = parseFloat(tmp + parseFloat(prc[i].value));
        }
    }
    products.value= txt
    price.value=Math.round(tmp * 1000) / 1000
}
//item validation form
function validate()
{
    x=document.myForm;
    y=document.getElementById("category")
    z=document.getElementById("provider")
    pname=x.pname.value;
    inum=x.inum.value;
    invnum=x.invnum.value;
    price=x.price.value;
    category=y.value;
    provider=z.value;
    if(pname=="")
    {
        alert("must enter a name");
    }
return false;
}
if(num=="")
{
  alert("must enter a item number");
  return false;
}
if(invnum=="")
{
  alert("must enter an invoice");
  return false;
}
if(category=="0")
{
  alert("must select a category value");
  return false;
}
if(provider=="0")
{
  alert("must select a provider value");
  return false;
}
if(price=="")
{
  alert("must enter a price");
  return false;
}
if(price==0 || price < 0)
{
  alert("price must be greater than 0");
  return false;
}
if(price==parseFloat(price))
{
  alert("Price must be a number");
  return false;
}
return true;

//product form validation
function pvalidate() 
{
  x=document.product;
  pname=x.pname.value;
  productid=x.product_id.value;
  price=x.price.value;
  //okSubmit="true";

  if(pname =="")
  {
    alert("You must enter a product name");
    return false;
  }
  if(productid =="")
  {
    alert("You must enter a product number");
  }
function cvalidate()
{
x=document.combo;
cname=x.cname.value;
comboid=x.combo_id.value;

if(cname =="")
{
    alert("You must enter a combo name");
    return false;
}
if(comboid =="")
{
    alert("You must enter a combo number");
    return false;
}
checkMade();
return true;
}

function provalidate()
{
x=document.provide;
prov=x.prov.value;
pphone=x.pphone.value;
sale=x.sale.value;

if(prov =="")
{
    alert("You must enter a Provider");
    return false;
}
if(pphone =="")
{
    alert("You must enter a Provider's Phone ");
    return false;
}
if(sale =="")
{
    alert("You must enter a name of a sales representative");
    return false;
}
return true;
}

Index.php
<?php
    session_start();
if($_SESSION['username'] == true && $_SESSION['password'] == true)
{
    header("Location: loginaction.php");
}
else {
    header("Location: login.php");
}
REFERENCES

[1] MySQL General information

[2] PHP Programming
http://www.php.net

http://www.apache.org

[4] SYSCO
http://www.sysco.com/