

University of Macau

Faculty of Science and Technology

Department of Computer and Information Science



澳門大學

UNIVERSIDADE DE MACAU

UNIVERSITY OF MACAU

CISC3003 -- Web Programming

Project Walkthrough Example Brief

Shopping Cart System

Submitted by: D-B9-2130-6 FONG CHON HOU

Table of contents

INTRODUCTION	3
FILE STRUCTURE	3
HOMEPAGE (INDEX.PHP):	3
CART PAGE (CART.PHP):	3
CHECKOUT PAGE (CHECKOUT.PHP):	4
BACKEND OPERATIONS (ACTION.PHP):	4
SETTING TO CONNECT MYSQL (CONFIG.PHP):	5
SECURITY CONSIDERATIONS	5
SUMMARY	5

Introduction

The shopping cart system is an e-commerce application that allows users to browse products, add them to the shopping cart, and then checkout the following orders. The system uses MySQL and XAMPP stack to fully implement the selected project in the Eclipse IDE for PHP developers. This project also including HTML, CSS, JS, jQuery, PHP coding in this PHP files.

File Structure

- index.php: The homepage of the application, displaying the available products.
- config.php: Configuration file containing database connection details.
- checkout.php: The checkout page where users can enter their details and complete the purchase.
- cart.php: The cart page displaying the products added to the cart.
- action.php: Backend script handling various cart operations.

Homepage (index.php):

- The homepage displays a list of available products.
- Each product is shown with its name, price, image, and an "Add to Cart" button.
- When the "Add to Cart" button is clicked, the product details are sent to the server using AJAX and processed by the action.php script.
- If the product is not already in the cart, it is added to the cart table in the database, and a success message is displayed.
- If the product is already in the cart, an error message is displayed.

Cart Page (cart.php):

- The cart page displays the products added to the cart.
- Each product is shown with its name, price, quantity, and total price.
- Users can update the quantity of each product using input fields.
- The total price is automatically updated based on the quantity changes.
- Users can remove a product from the cart by clicking the "Remove" button.
- Users can also clear the entire cart by clicking the "Clear Cart" button.

Checkout Page (checkout.php):

- The checkout page displays a form where users can enter their personal and payment details to complete the purchase.
- Users enter their name, email, phone, address, and select a payment method.
- When the "Place Order" button is clicked, the form details are sent to the server using AJAX and processed by the action.php script.
- The order details are saved in the orders table in the database, and the cart is cleared.
- A thank you message is displayed, showing the purchased items, customer information, and payment details.

Backend Operations (action.php):

- The action.php script handles various cart operations.
- It establishes a connection to the database using the configuration details from config.php.
- It contains multiple conditional blocks for different actions, such as adding a product to the cart, updating quantities, removing items, and placing an order.
- The script uses prepared statements to prevent SQL injection and executes the necessary database queries.
- It also uses sessions to store and display alert messages for successful or failed operations.

Setting to connect MySQL (config.php):

- This config.php file contains the settings for database connections. Use the mysqli function to establish a connection to the MySQL database.
- It accepts four parameters: database host name (localhost), database username (root), database password (Your password), and database name (cart_system). In practical applications, we need to modify these values based on our own database settings.
- If the connection fails, display the connection error message through the die() function.

Security Considerations

- User inputs are validated and sanitized before processing.
- The system encrypts sensitive customer information, such as passwords or payment details, before storing them in the database.
- Proper error handling and exception handling techniques are implemented to ensure robustness and security.

Summary

The shopping cart system provides a user-friendly interface for users to browse products, add products to the shopping cart, and complete purchases. It can ensure data security, handle various shopping cart operations, and provide a smooth user experience.