Hong Kong College of Technology

Technology For Web Development (03CIT4053)  
Online shop Project

Student Name: Cheng Man Hei, Chow Sun Yee, Pak Hoi Yi

Teacher: Mr. Stephen NG

Content:

* Introduction
* Site Map
* Work Flow
* Features
* Technical Specifications
* Setup Manual
* Project Timeline
* User Acceptance
* Appendix
* Introduction

1. Providing a Convenient Shopping Experience

- Shop Anytime, Anywhere: One of the primary advantages of an online pet store is the ability for consumers to shop at their convenience. This flexibility allows pet owners to browse for products during their free time, whether it's late at night or during a lunch break. By eliminating the need to travel to a physical store, customers save time and can make informed decisions without feeling rushed.

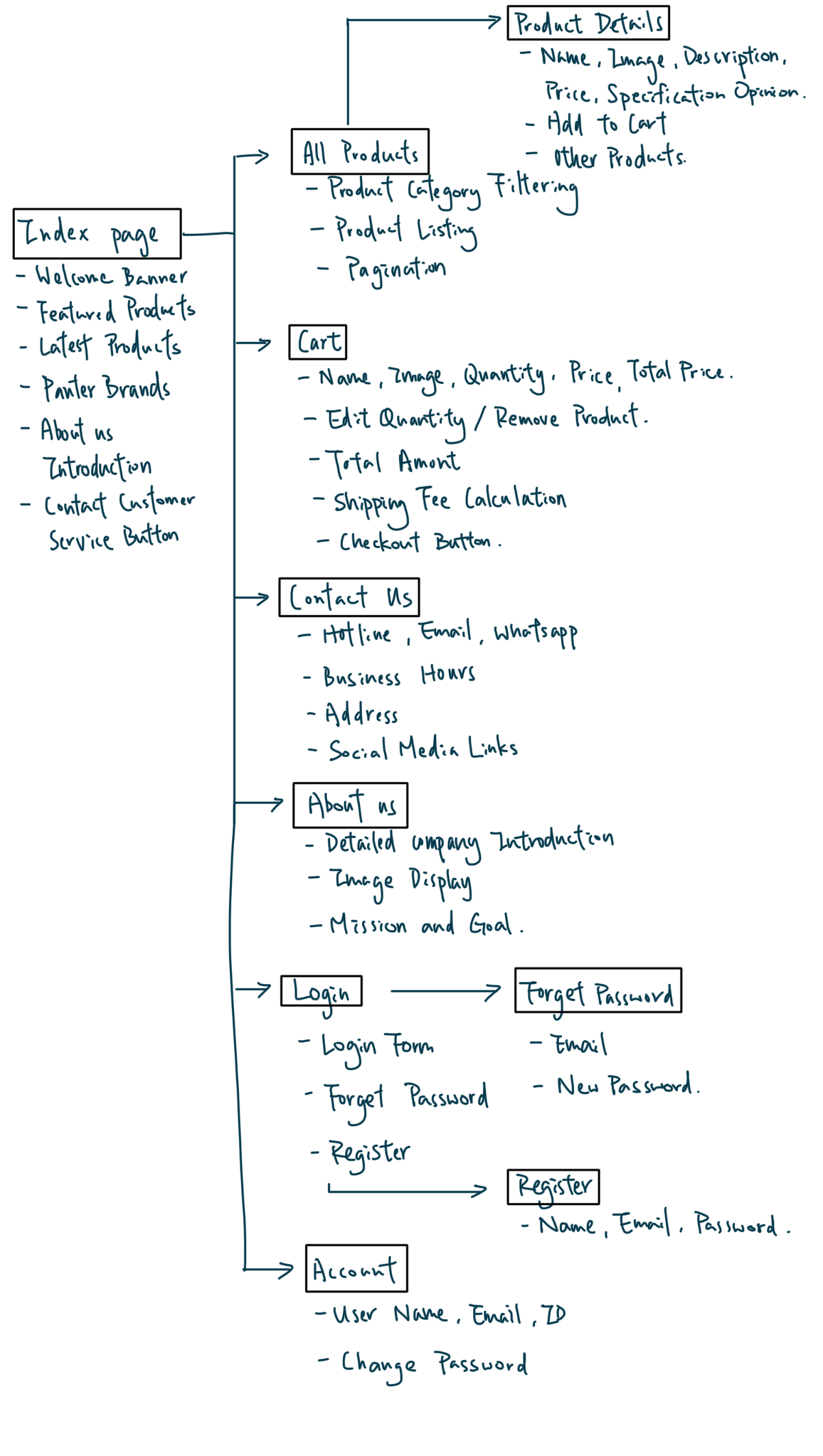
- Diverse Selection: Online pet stores typically offer a broader selection of products compared to brick-and-mortar shops. From specialty foods and health supplements to unique toys and accessories, pet owners can find items tailored to their pets’ specific needs. This variety ensures that customers can discover new products that enhance their pets' well-being and happiness.

2. Providing Professional Knowledge and Services

- Sharing Pet Care Knowledge: In addition to selling products, online pet stores can serve as valuable resources for pet owners. By providing informative content such as articles, how-to guides, and instructional videos, these platforms can educate customers on proper pet care, nutrition, and training techniques. This knowledge not only empowers pet owners but also fosters a sense of community and trust.

- Personalized Recommendations: Utilizing data analytics, online pet stores can offer personalized shopping experiences. By analyzing customers' previous purchases and preferences, the store can suggest relevant products, making it easier for pet owners to find what they need. This tailored approach enhances customer satisfaction and encourages repeat business, as shoppers feel understood and valued.

* Site Map



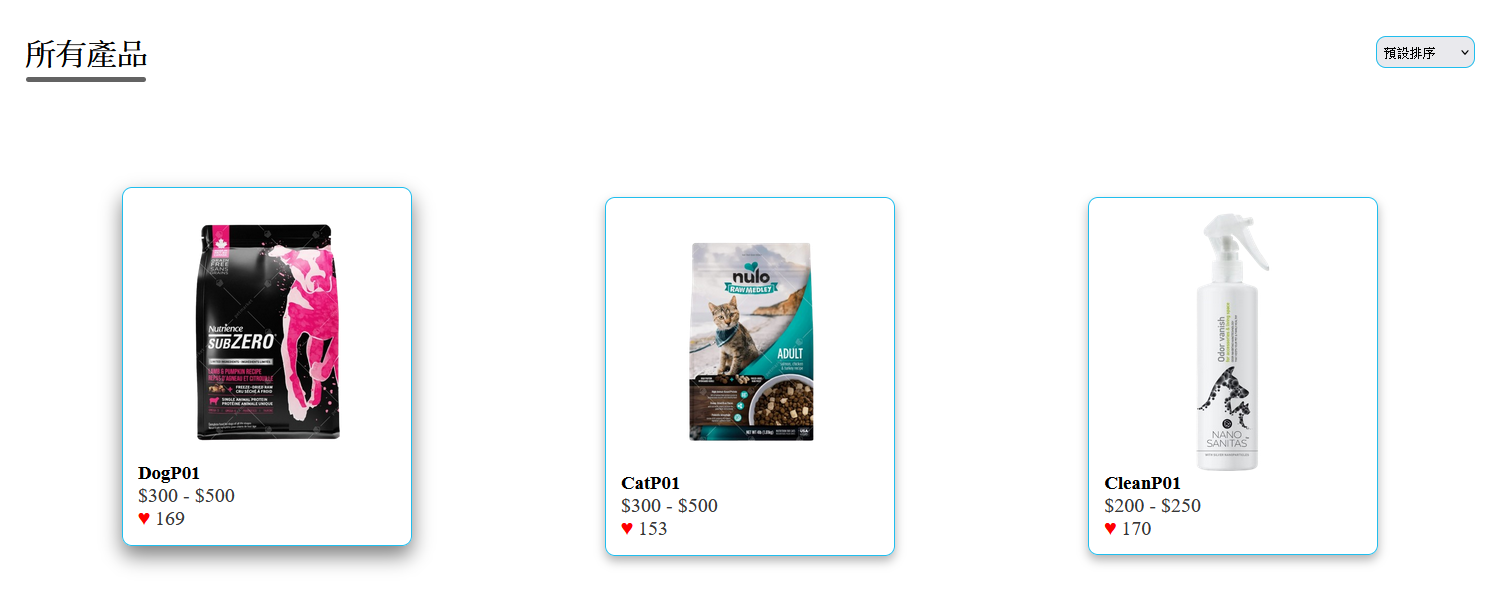
* Work Flow

Coding the web in HTML without database,  
Switch the HTML into PHP coding,  
Setting up the database, and pulling data from the database to the website.

* Features

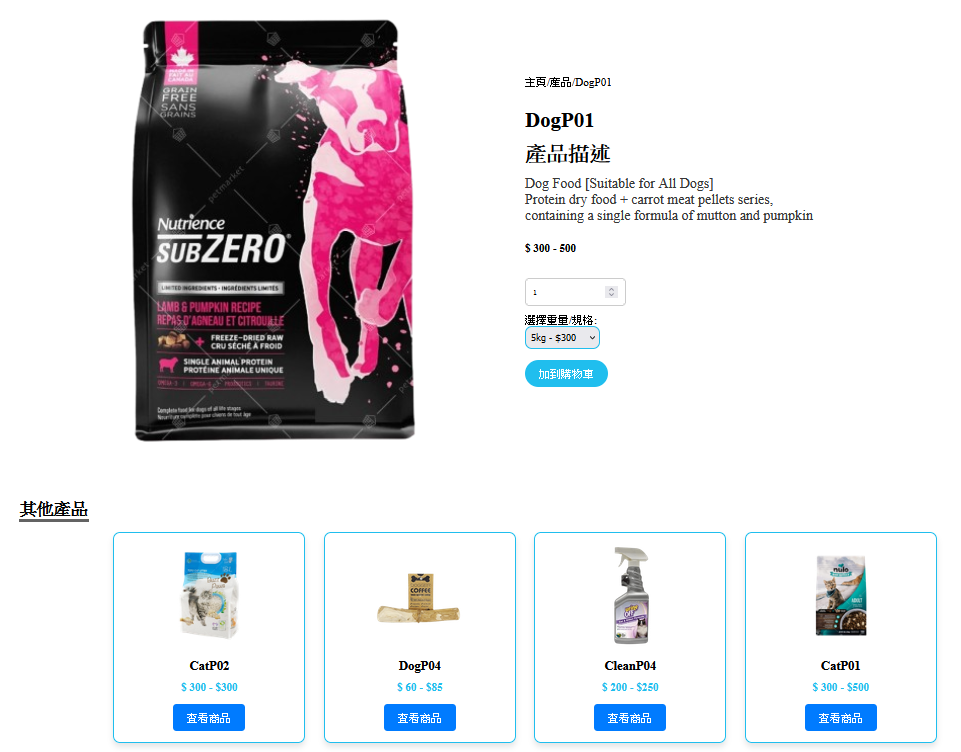
Feature Products: can be update in database

Slide Show: Having 4 pictures to show out

Product cards: Click on it and jump to that product’s PID(Product ID)

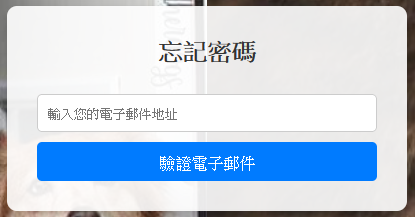
Product Details: can select needed quantity/ specification options, can add products to Cart

Other Products: can click on the button, and jump on that product



Cart: can change quantity and update cart/ delete product

Login: Login Form, Forget Password/ Register



Contact Us: Word of “Follow us” can be click and color can be change



About Us: Mainly the pictures on the right side, will change automatically after 3s

Account: can check user’s data and change password

* Technical Specifications

Navbar: can be detect rather login or not by UID(User ID)

session\_start();

$is\_logged\_in = isset($\_SESSION['user\_id']); // 檢查是否已登入

<!-- 橫向菜單 -->

<nav>

<ul id="menuItems">

<li><a href="Index.php">主頁</a></li>

<li><a href="Products.php">產品</a></li>

<li><a href="Contact.php">聯絡我們</a></li>

<li><a href="About.php">關於我們</a></li>

<?php

if (isset($\_SESSION['uid'])) {

// 如果已登入，顯示賬戶連結

echo '<li><a href="Account.php">賬戶</a></li>';

} else {

// 如果未登入，顯示登入連結

echo '<li><a href="Login.php">登入</a></li>';

}

?>

</ul>

</nav>

<a href="Cart.php">

<img src="image/cart.png" width="40px" height="40px">

<span id="cart-count"><?php echo count($\_SESSION['cart']); ?></span>

</a>

<img src="image/menu.png" class="menu-icon" onclick="menutoggle()">

</div>

</div>

Slideshow: through “n” to decide which picture to show out and click “<”(&#10094)/ ”>”(&#10095) to switch picture

<?php foreach ($newProducts as $product): ?>

<div class="slidesImg">

<img

src="image/<?php echo htmlspecialchars($product['PName']); ?>.png"

alt="<?php echo htmlspecialchars($product['PName']); ?>"

width="100%">

</div>

<?php endforeach; ?>

<a class="prev" onclick="plusSlides(-1)">&#10094;</a>

<a class="next" onclick="plusSlides(1)">&#10095;</a>

</div>

Slideshow javascript:

<script>

let slideIndex = 1;

showslides(slideIndex);

function plusSlides(n) {

showslides(slideIndex += n);

}

function currentSlide(n) {

showslides(slideIndex = n);

}

function showslides(n) {

let i;

let slides = document.getElementsByClassName("slidesImg");

if (n > slides.length) {slideIndex = 1}

if (n < 1) {slideIndex = slides.length}

for (i = 0; i < slides.length; i++) {

slides[i].style.display = "none";

}

slides[slideIndex - 1].style.display = "block";

}

</script>

Other products: products may repeat showed, or showed all products, limited to show only 4 cards, and group by GID(Group ID); click on the button, and it will jump to that product’s PID(Product ID)

// 獲取隨機的 4 個其他產品，基於 GID 分組，避免重複

$sql\_random\_products = "

SELECT p.GID, p.PID, p.PName, MIN(p.Price) AS minPrice, MAX(p.Price) AS maxPrice

FROM productdata p

WHERE p.GID != ?

GROUP BY p.GID

ORDER BY RAND()

LIMIT 4";

$stmt\_random = $conn->prepare($sql\_random\_products);

$stmt\_random->bind\_param("i", $gid); // 使用 GID 避免同產品變體出現

$stmt\_random->execute();

$result\_random\_products = $stmt\_random->get\_result();

$random\_products = [];

while ($row = $result\_random\_products->fetch\_assoc()) {

$random\_products[] = $row;

}

Website php:

<?php foreach ($random\_products as $random\_product): ?> <!-- 調試輸出 -->

<div class="col-pd-4">

<img src="image/<?php echo htmlspecialchars($random\_product['PName']); ?>.png" alt="<?php echo htmlspecialchars($random\_product['PName']); ?>">

<h4><?php echo htmlspecialchars($random\_product['PName']); ?></h4>

<p>

$

<?php

$minPrice = isset($random\_product['minPrice']) ? htmlspecialchars($random\_product['minPrice']) : 'N/A';

$maxPrice = isset($random\_product['maxPrice']) ? htmlspecialchars($random\_product['maxPrice']) : 'N/A';

echo $minPrice . " - $" . $maxPrice;

?>

</p>

<a href="Product-Details.php?pid=<?php echo htmlspecialchars($random\_product['PID']); ?>">

<button class="view-btn">查看商品</button>

</a>

</div>

<?php endforeach; ?>

Cart: product’s quantity can be change, and update the cart to refresh the total price, and products can be deleted if user want

// 處理刪除商品

if (isset($\_GET['action']) && $\_GET['action'] === 'delete' && isset($\_GET['pid']) && isset($\_GET['variant'])) {

$pid = intval($\_GET['pid']);

$variant = htmlspecialchars($\_GET['variant']);

foreach ($\_SESSION['cart'] as $key => $item) {

if ($item['pid'] === $pid && $item['variant'] === $variant) {

unset($\_SESSION['cart'][$key]);

break;

}

}

$\_SESSION['cart'] = array\_values($\_SESSION['cart']); // 重新索引購物車

}

// 處理更改數量

if ($\_SERVER['REQUEST\_METHOD'] === 'POST' && isset($\_POST['update\_cart'])) {

$updated\_cart = []; // 初始化一個新的購物車數據陣列

// 獲取數量和產品識別數據

if (isset($\_POST['quantities']) && is\_array($\_POST['quantities'])) {

foreach ($\_POST['quantities'] as $key => $quantity) {

// 分離 PID 和 Variant 作為鍵

list($pid, $variant) = explode('\_', $key);

// 確保數量為有效整數

$quantity = max(1, intval($quantity));

// 查找原始購物車數據並保留其他資訊

foreach ($\_SESSION['cart'] as $item) {

if ($item['pid'] == $pid && $item['variant'] == $variant) {

$item['quantity'] = $quantity; // 更新數量

$updated\_cart[] = $item; // 加回到新的購物車

break;

}

}

}

}

$\_SESSION['cart'] = $updated\_cart; // 更新購物車

}

Website show:

<?php

$total = 0; // 初始化總金額

foreach ($\_SESSION['cart'] as $item):

$subtotal = floatval($item['price']) \* intval($item['quantity']);

$total += $subtotal;

?>

<tr>

<td>

<div class="cart-info">

<img src="image/<?php echo htmlspecialchars($item['name']); ?>.png" alt="<?php echo htmlspecialchars($item['name']); ?>">

<div>

<p><?php echo htmlspecialchars($item['name']); ?></p>

<middle>價格: $<?php echo number\_format(floatval($item['price']), 2); ?></middle><br>

</div>

</div>

</td>

<td><?php echo htmlspecialchars($item['variant']); ?></td>

<td>

<!-- 使用 PID 和 Variant 作為鍵 -->

<input type="number" name="quantities[<?php echo $item['pid'] . '\_' . urlencode($item['variant']); ?>]" value="<?php echo intval($item['quantity']); ?>" min="1">

</td>

<td>$<?php echo number\_format(floatval($item['price']), 2); ?></td>

<td>$<?php echo number\_format($subtotal, 2); ?></td>

<td>

<a href="Cart.php?action=delete&pid=<?php echo $item['pid']; ?>&variant=<?php echo urlencode($item['variant']); ?>" class="btn delete-btn">刪除</a>

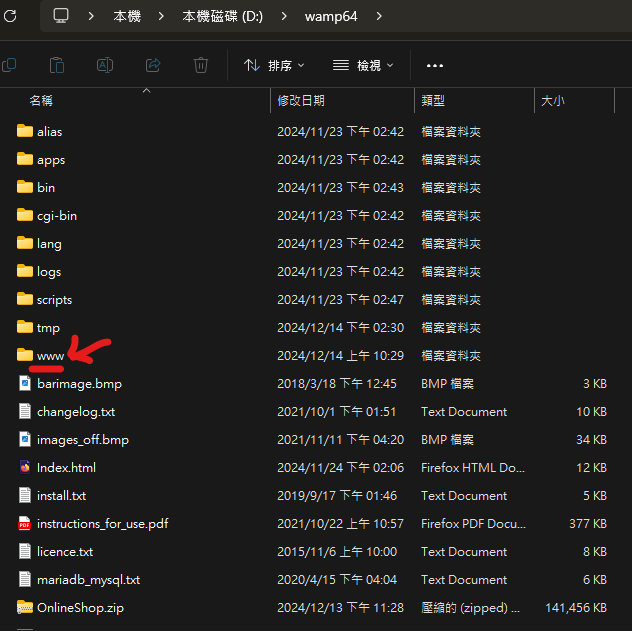
</td>

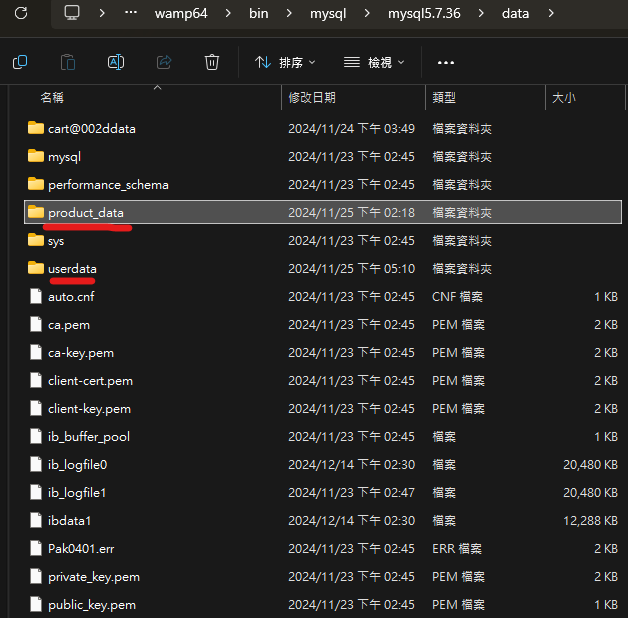
</tr>

<?php endforeach; ?>

* Setup Manual

First, Unzipping the project folder. Download “wamp64”(support windows only) / “xampp”(support windows, macOS, linux..) for holding a server to run the website and the database.

After downloading the server, open the file where you install the application, once you open it, you can see the file named “www”, put the project inside the “www” file.

In the project, you will see a file named “sql”, pull all the files to “wamp63>bin>mysql>mysql(version)>data” basically you can already run the website.

Then, you can start up the wamp server, and open the browser type “localhost/OnlineShop/Index.php” or “localhost/phpmyadmin/”, you can go to the website or phpMyadmin to edit your product data/ user data. if you want to edit the database of the website, while login, Username: root and without a password, server choices: MySQL. And you can login to edit the database.

* Project Timeline

First, Coding the web with HTML, web included Index, Products, Products Details, Login, Register, Cart, About Us pages.

Scound, Changing HTML into PHP coding, adding Contact Us, Forget Password, Account pages; also set up a database for the website to pull out data.

* User Accpetance

Login:

* if only fill in “user email” field, “password” field will appear ‘please fill in field’, can’t login
* if only fill in “password” field, can’t login, message “please fill in field”
* If the user fills in all the data, the user can login, message “Login successfully” and jump to “Index.php”
* if user fills in wrong password, “Password” field will jump out “Password error”

Forget Password:

* fill in user email, if the email is registered, it can jump to next page and setup new password
* fill in user email, if the email is not yet registered, it will pop out ‘this email is not yet registered.’, it will stay at the same page.
* if user fill in the same password as new password, it jump up message “new password can’t be the same as the password currently in use”
* if user fills in the new password, and the confirm password is not same as the new password, message “New Password is not the same as Confirm password”
* if the user fills in a new password and confirm password correctly, it will jump back to the login page.

Register:

* if fill in all blank, it won’t register successfully
* if any field is empty, the account can’t be register successfully
* If filled in all data correctly, the user can register successfully, and jump out a message”Register successfully, jump back to the Login page in 3 seconds..”

Reset Password:

* If a user fills in the same password as the new password, users can still update their password, message “Password update successfully”
* if the user fills in the wrong password that is currently in use and typed a new password, message “password currently in use is not correct”.

* Appendix

Used resources:

* <https://www.youtube.com/watch?v=yQimoqo0-7g&list=PLjwm_8O3suyM_2Lo9aAIw3HqjOPor8j9g>
* <https://www.youtube.com/watch?v=ZbnvP_hmxfE&list=PLjwm_8O3suyM_2Lo9aAIw3HqjOPor8j9g&index=2>
* <https://www.youtube.com/watch?v=ENyk_W-Eleo&list=PLjwm_8O3suyM_2Lo9aAIw3HqjOPor8j9g&index=3>
* <https://www.youtube.com/watch?v=oXrlgOEiy6o&list=PLjwm_8O3suyM_2Lo9aAIw3HqjOPor8j9g&index=4>
* <https://www.youtube.com/watch?v=vOXGuNVRGpA&list=PLjwm_8O3suyM_2Lo9aAIw3HqjOPor8j9g&index=5>