



Lab Report 6

Admin Panel for Product, Brand Category and order management

Course: CSC381 - E-Commerce

Submitted by:

Parakram Kharel
Roll No: 24

Kathford International College of Engineering and Management
Affiliated to Tribhuvan University

August 17, 2025

1. Objective

Design and implement an admin panel to manage products, brands, categories, and orders, enabling administrators to add, update, soft-delete/restore products and to review and update order statuses in a secure, efficient workflow.

2. Tools and Technologies Used

Technology	Purpose
HTML5	Admin pages and form structure
CSS3	Layout and visual styling
JavaScript	Inline confirmations and UI behavior
PHP	Server-side logic, CRUD
MySQL	Persistent storage for catalog and orders
Caddy	Web server and local hosting
Bootstrap 5	Admin UI, tables, forms, responsive grid
VS Code	Development environment
Browser DevTools	Testing and debugging

3. Theory / Background

The admin panel serves as a secure CRUD interface over MySQL, where PHP executes parameterized queries to manage products, brands, categories, and orders. Server-side rendering outputs compact, responsive pages with Bootstrap tables and forms, while output escaping prevents XSS. Soft delete toggles a status flag instead of removing rows, preserving data integrity and references.

4. Page Layout Design

4.1 Product Listing Grid

- Admin list shows products with image, title, price, status, and actions.
- Inline actions: edit, soft delete (disable), and optional restore.
- Compact, responsive rows for quick scanning.

4.2 Filtering by Brand and Category

- Sidebar links filter server-side by brand/category id.
- Empty results display a short “No data found” notice.

4.3 Product Details Page

- Secure load by id (prepared statement).
- Up to three images, description, brand, category, and price.

4.4 Navigation and Search

- Top nav for sections (Products, Brands, Categories, Orders).
- Search form submits to server to filter listings.

4.5 Data and Security Practices

- Prepared statements for all write/read by id.
- Escaped outputs for titles, images, and prices.
- Soft delete via status field; restore supported from admin.

4.6 Order Management

- Admin list shows orders with ID, order number, total amount, payment mode, status, and placed date.
- Inline actions: update status (pending, paid, processing, shipped, delivered, cancelled).
- Responsive table for easy scanning.

5. Code Snippets

5.1 Admin Manage Order

```
1  <?php
2  $flash = '';
3  if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['action']) &&
4  $_POST['action'] === 'update_status') {
5      $order_id = isset($_POST['order_id']) ? (int)$_POST['order_id'] : 0;
6      $status = $_POST['status'] ?? '';
7      if ($order_id <= 0) {
8          $flash = 'Bad order id.';
9      } elseif (!in_array($status, $ALLOWED, true)) {
10         $flash = 'Invalid status.';
11     } else {
12         $u = $conn->prepare("UPDATE orders SET status = ? WHERE id = ?
13         LIMIT 1");
14         $u->bind_param('si', $status, $order_id);
15         if ($u->execute()) {
16             $flash = 'Order status updated.';
17         } else {
18             $flash = 'Update failed.';
19         }
20         $u->close();
21     }
22 }
```

5.2 PHP Code to Add Brand and Category

```
1  // INSERT BRAND
2  <?php
3  $insert_query = "INSERT INTO `brands`(`brand_title`) VALUES
4  ('$brand_title')";
5  $result = mysqli_query($conn, $insert_query);
6
7  if ($result) {
8      echo "<script>alert('Brand has been inserted successfully')</script>";
9  } else {
10     die("Insert failed: " . mysqli_error($conn));
11 }
```

// INSERT CATEGORY

```

12 $insert_query = "INSERT INTO `categories`(`category_title`) VALUES
13 ('$category_title')";
14 $result = mysqli_query($conn, $insert_query);
15 if ($result) {
16     echo "<script> alert('categories has been inserted
17     successfully')</script>";
18 } else {
19     die("Insert failed: " . mysqli_error($conn));
20 }
21 ?>

```

5.3 Add Products

```

1  <?php
2  $stmt = $conn->prepare("INSERT INTO `products` (product_title,
3  product_description,product_keywords,category_id,
4  brand_id,product_image1,product_image2,
5  product_image3,product_price,date,
6  status) VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, NOW(), ?)");
7
8  if (!$stmt) {
9      error_log('Prepare failed: ' . $conn->error);
10     echo "<script>alert('Database error. Please try again
11     later.');

```

```

32 <label for="description">Product Description <span>*</span></label>
33 <textarea id="description" name="description" rows="3"
34     placeholder="Enter product description" required></textarea>
35 <label for="product_keywords">Keywords <span>*</span></label>
36 <input type="text" id="product_keywords" name="product_keywords"
37     placeholder="Enter keywords separated by comma" required />
38
39 <label for="product_category">Category <span>*</span></label>
40 <select id="product_category" name="product_category" required>
41     <option value="" disabled selected>Select category</option>
42     <?php
43         $categories = mysqli_query($conn, "SELECT * FROM cateogries");
44         while ($cat = mysqli_fetch_assoc($categories)) {
45             echo "<option
46                 value='{$cat['category_id']}'>{$cat['category_title']}</option>";
47         }
48     ?>
49 </select>
50 <label for="product_brand">Brand <span>*</span></label>
51 <select id="product_brand" name="product_brand" required>
52     <option value="" disabled selected>Select brand</option>
53     <?php
54         $brands = mysqli_query($conn, "SELECT * FROM brands");
55         while ($brand = mysqli_fetch_assoc($brands)) {
56             echo "<option
57                 value='{$brand['brand_id']}'>{$brand['brand_title']}</option>";
58         }
59     ?>
60 </select>
61
62 <label for="product_price">Price (NPR) <span>*</span></label>
63 <input type="number" min="0" id="product_price" name="product_price"
64     required />
65
66 <label for="product_image1">Upload Images<span>*</span></label>
67 <input type="file" name="product_image1" accept="image/*"
68     id="product_image1" required />
69 <input type="file" name="product_image2" accept="image/*"
70     id="product_image2" required />
71 <input type="file" name="product_image3" accept="image/*"
72     id="product_image3" required />
73 <button type="submit" name="insert_product">Insert Product</button>
74 </form>

```

5.4 PHP: Edit Product

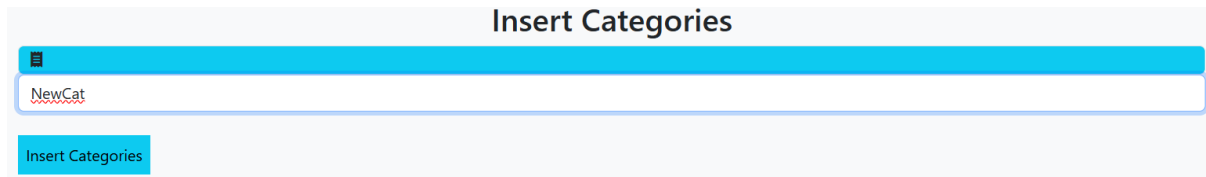
```

1  <?php
2  $stmt2 = $conn->prepare("
3      UPDATE products
4          SET product_title = ?, product_description = ?,
5          product_keywords = ?, category_id = ?,
6          brand_id = ?, product_image1 = ?,
7          product_image2 = ?, product_image3 = ?,
8          product_price = ?, status = ?
9      WHERE product_id = ?
10     LIMIT 1
11 ");
12
13 $typeString = 'sssiisssssi';
14 $stmt2->bind_param(
15     $typeString, $title, $desc, $keys, $catId, $brandId,
16     $images['product_image1'], $images['product_image2'],
17     $images['product_image3'], $price, $status, $product_id
18 );
19
20 if ($stmt2->execute()) {
21     echo "<script>alert('Product updated successfully.')</script>";
22 } else {
23     echo "<script>alert('Update failed. Please try again.')</script>";
24 }
25 $stmt2->close();
26 ?>
27 <form method="post" enctype="multipart/form-data" class="row g-3 bg-white
28 p-3 rounded shadow-sm">
29     <label class="form-label">Title</label>
30     <input type="text" name="product_title" value="<?=
31     e($current['product_title']) ?>" required>
32
33     <label class="form-label">Keywords</label>
34     <input type="text" name="product_keywords" value="<?=
35     e($current['product_keywords']) ?>" required>
36
37     <label class="form-label">Description</label>
38     <textarea name="product_description" rows="4" required><?=
39     e($current['product_description']) ?></textarea>
40
41     <label class="form-label">Brand</label>
42     <select name="brand_id" required>
43         <option value="">Select brand...</option>
44         <?php foreach ($brands as $b): ?>
45             <option value="<?= (int)$b['brand_id'] ?>" <?=
46                 $current['brand_id']==$b['brand_id']?'selected':'' ?>>

```

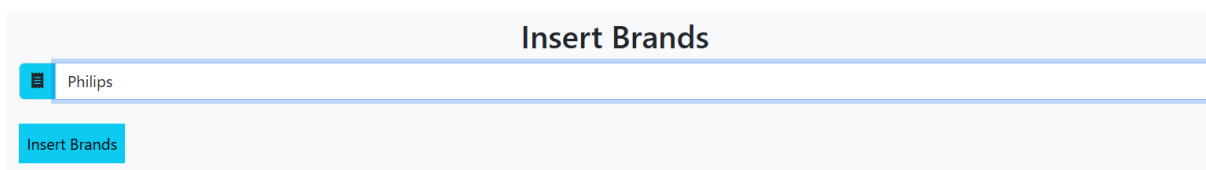
```
42         <?= e($b['brand_title']) ?>
43     </option>
44     <?php endforeach; ?>
45 </select>
46
47 <label class="form-label">Category</label>
48 <select name="category_id" required>
49     <option value="">Select category...</option>
50     <?php foreach ($cats as $c): ?>
51         <option value="<?= (int)$c['category_id'] ?>" <?=
52             $current['category_id']==$c['category_id']?'selected':'' ?>
53             <?= e($c['category_title']) ?>
54         </option>
55     <?php endforeach; ?>
56 </select>
57
58 <label class="form-label">Price</label>
59 <input type="text" name="product_price" value="<?=
60 e($current['product_price']) ?>" required>
61
62 <label class="form-label">Status</label>
63 <select name="status" required>
64     <option value="true" <?= $current['status']==='true'?'selected':''
65     ?>>Active</option>
66     <option value="false" <?= $current['status']!=='true'?'selected':''
67     ?>>Inactive</option>
68 </select>
69
70 <label>Replace Image 1</label>
71 <input type="file" name="product_image1" accept="image/*">
72
73 <label>Replace Image 2</label>
74 <input type="file" name="product_image2" accept="image/*">
75
76 <label>Replace Image 3</label>
77 <input type="file" name="product_image3" accept="image/*">
78
79 <button type="submit">Save</button>
80 <a href="index.php?list_products">Back</a>
81 </form>
```


6. Output / Screenshots



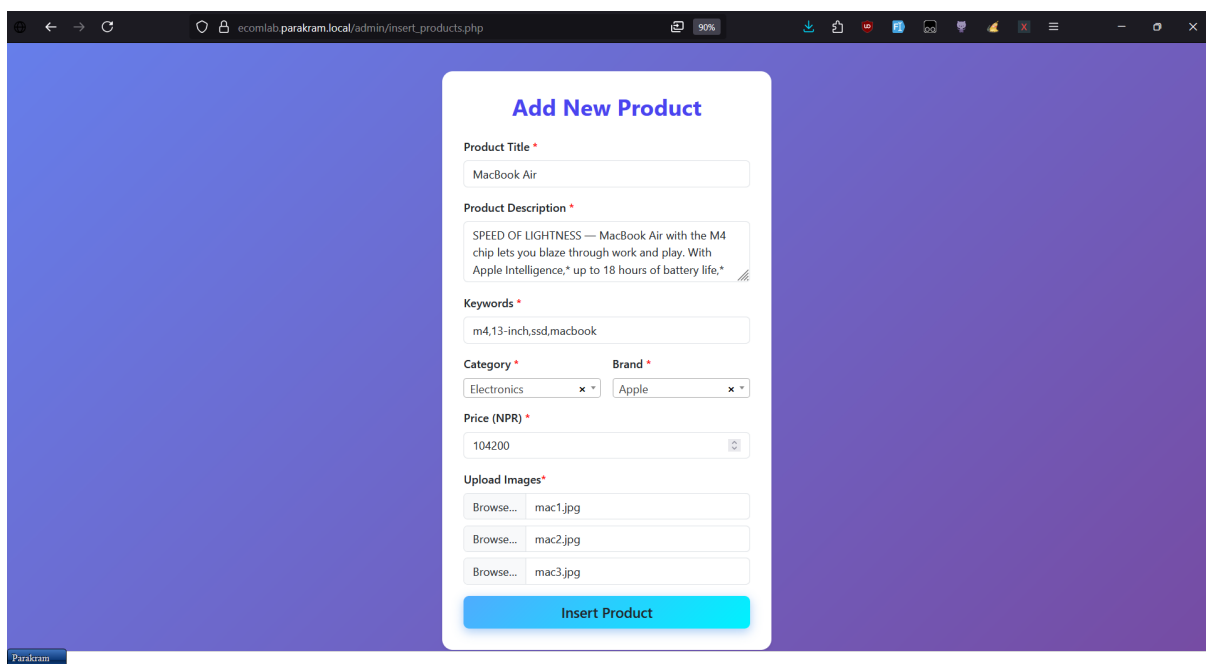
The screenshot shows a web form titled "Insert Categories". At the top, there is a blue header bar with the title. Below it is a text input field containing the text "NewCat". At the bottom of the form is a blue button labeled "Insert Categories".

Figure 1: Insert Category.



The screenshot shows a web form titled "Insert Brands". At the top, there is a blue header bar with the title. Below it is a text input field containing the text "Philips". At the bottom of the form is a blue button labeled "Insert Brands".

Figure 2: Insert Brand.

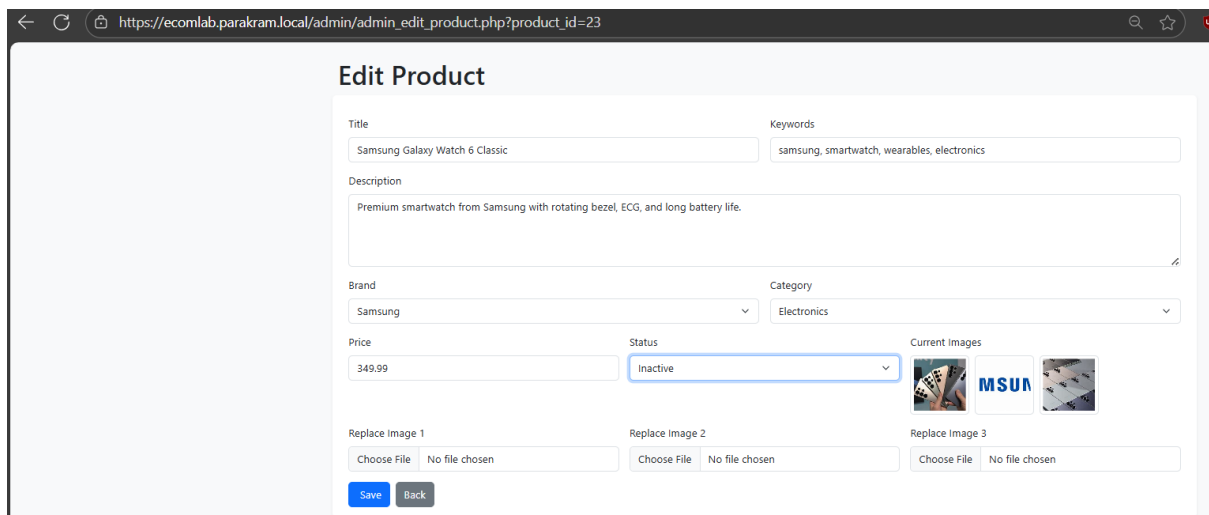


The screenshot shows a web browser window displaying the "Add New Product" form. The browser's address bar shows the URL "ecomlab.parakram.local/admin/insert_products.php". The form is titled "Add New Product" and contains the following fields:

- Product Title ***: Text input field with "MacBook Air" entered.
- Product Description ***: Text area with "SPEED OF LIGHTNESS — MacBook Air with the M4 chip lets you blaze through work and play. With Apple Intelligence,* up to 18 hours of battery life,*" entered.
- Keywords ***: Text input field with "m4,13-inch,ssd,macbook" entered.
- Category ***: Dropdown menu with "Electronics" selected.
- Brand ***: Dropdown menu with "Apple" selected.
- Price (NPR) ***: Text input field with "104200" entered.
- Upload Images ***: Three "Browse..." buttons, each followed by a filename: "mac1.jpg", "mac2.jpg", and "mac3.jpg".

At the bottom of the form is a blue button labeled "Insert Product".

Figure 3: Insert Product.



Edit Product

Title: Samsung Galaxy Watch 6 Classic

Keywords: samsung, smartwatch, wearables, electronics




Description: Premium smartwatch from Samsung with rotating bezel, ECG, and long battery life.

Brand: Samsung

Category: Electronics

Price: 349.99

Status: Inactive

Current Images:   

Replace Image 1: Choose File No file chosen

Replace Image 2: Choose File No file chosen

Replace Image 3: Choose File No file chosen

[Save](#) [Back](#)

Figure 4: Edit Product.

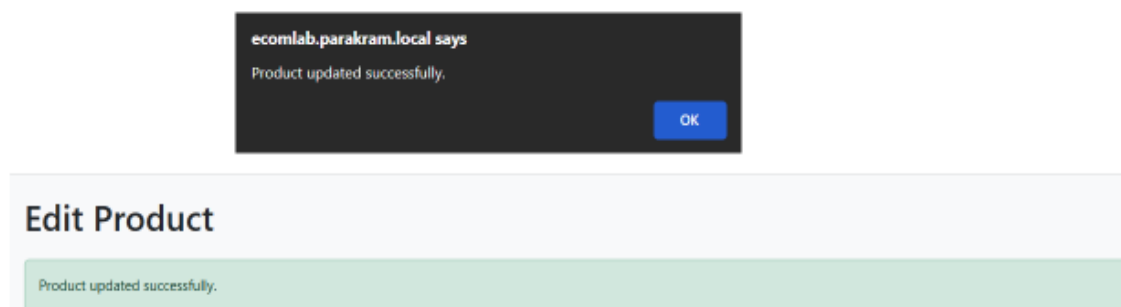


Figure 5: Edit Product Success.

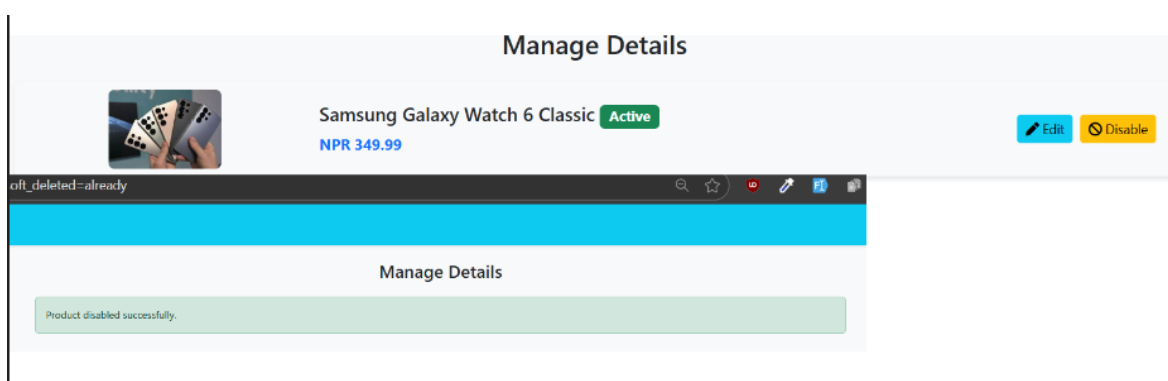
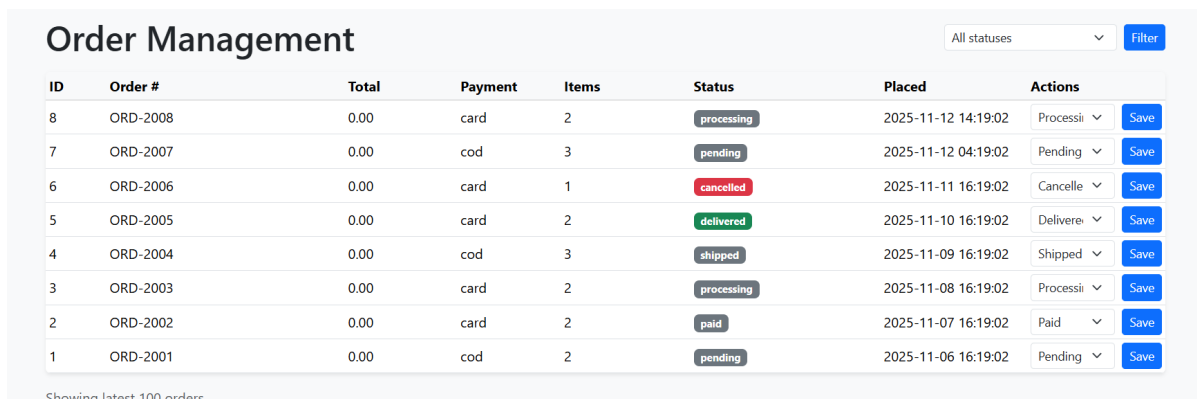


Figure 6: Delete Product.



ID	Order #	Total	Payment	Items	Status	Placed	Actions
8	ORD-2008	0.00	card	2	processing	2025-11-12 14:19:02	Processi <input type="button" value="Save"/>
7	ORD-2007	0.00	cod	3	pending	2025-11-12 04:19:02	Pending <input type="button" value="Save"/>
6	ORD-2006	0.00	card	1	cancelled	2025-11-11 16:19:02	Cancelle <input type="button" value="Save"/>
5	ORD-2005	0.00	card	2	delivered	2025-11-10 16:19:02	Delivere <input type="button" value="Save"/>
4	ORD-2004	0.00	cod	3	shipped	2025-11-09 16:19:02	Shipped <input type="button" value="Save"/>
3	ORD-2003	0.00	card	2	processing	2025-11-08 16:19:02	Processi <input type="button" value="Save"/>
2	ORD-2002	0.00	card	2	paid	2025-11-07 16:19:02	Paid <input type="button" value="Save"/>
1	ORD-2001	0.00	cod	2	pending	2025-11-06 16:19:02	Pending <input type="button" value="Save"/>

Showing latest 100 orders.

Figure 7: Order Management.

7. Result

The admin panel enables secure CRUD over the catalog and orders: products can be added, edited, soft-deleted/restored, and listed with images, price, and status. Orders are listed with totals and item counts, and their statuses can be updated inline (pending, paid, processing, shipped, delivered, cancelled). All interactions use parameterized queries, escaped outputs, and responsive Bootstrap tables and forms for reliable, fast administration.

8. Conclusion

The system delivers a maintainable back office for catalog and order management with clear workflows and safe data handling. Server-side rendering, prepared statements, and output escaping ensure integrity, while soft delete preserves history. The panel is ready for extensions like pagination, role-based access, audit logs, and export/reporting.

9. References

- PHP mysqli Prepared Statements
- PHP htmlspecialchars
- MySQL UPDATE Reference
- MySQL SELECT Reference
- Bootstrap 5 Documentation
- MDN HTTP Reference
- Caddy Web Server Documentation