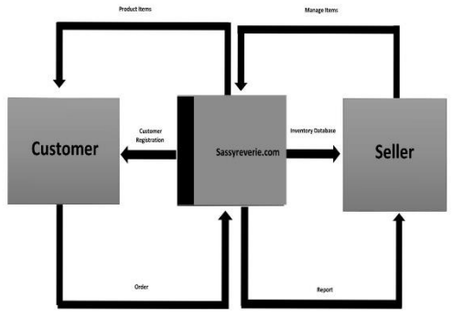
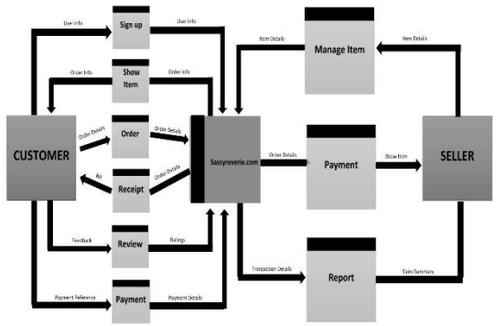
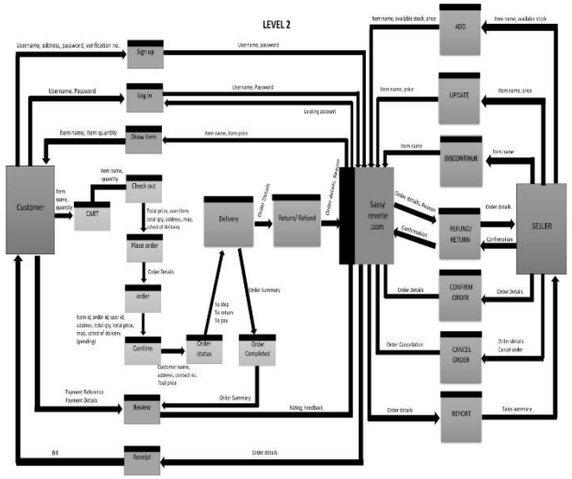
Group name: **REVERIE**
Members: **RICO SUSAINE, REODIQUE CURT JUSTIN N. VALEZA MHELARRY, SAÑADO FAITH ANN S., BORBE RECHELLE**
Course Yr. & Block: **BSIS-2A**

/*my role was on the last part*/

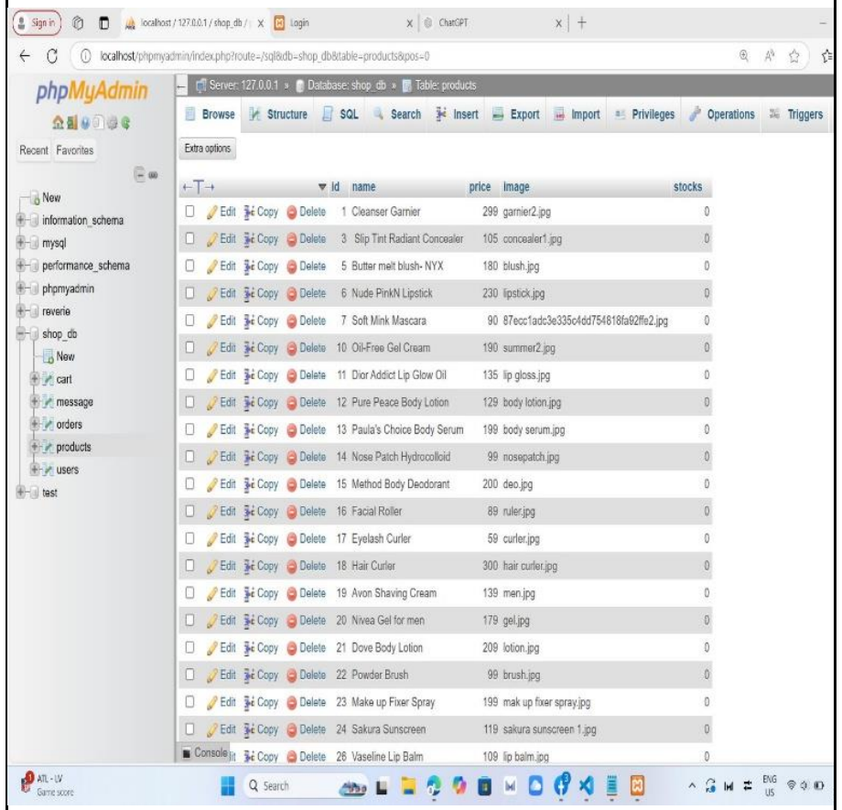
Problem Number 1: DFD
Problem Statement: Our Level 2 Data Flow Diagram contains inaccurate process icons. The database was incorrectly represented as a process instead of a data store.

CODE CHANGES	**INTERFACE CHANGES**	**Describe the functionality added or Changed**

None	DATA FLOW DIAGRAM LEVEL 0  LEVEL 1  	We corrected it to accurately reflect the system's data storage.



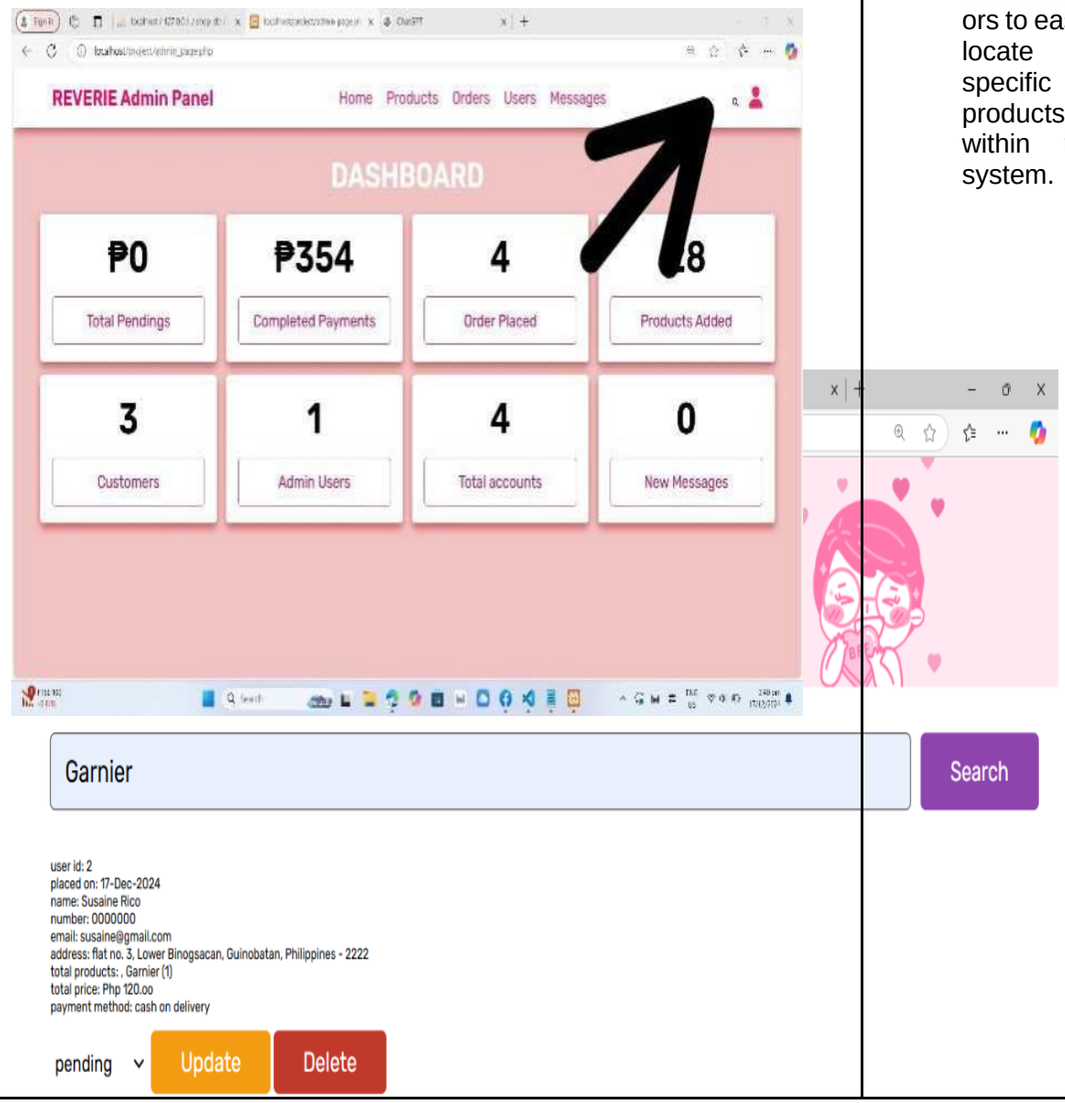
Problem Number 2: Add stock history
Problem Statement: Stocks problem statement: Our customers currently lack visibility into product availability. The absence of stock history on our website prevents them from seeing whether items are in stock before ordering, or how many stocks are still there in the products they are viewing.

CODE CHANGES	INTERFACE CHANGES	Describe the functionality added or Changed
		

Problem Number 3: ADMIN SEARCH BAR
Problem statement: Admin display will have a problem in the future once there are orders more than the display. Find a way to make it tabular and put a search bar that can find the specific order.

<?php **include 'config.php';** **session_start();** **$user_id = $_SESSION['user_id'];** **if (!isset($user_id)) {** **header('location:login.php');** **exit;** **}** **?>** **<!DOCTYPE html>** **<html lang="en">** **<head>** **<meta charset="UTF-8">** **<meta http-equiv="X-UA-Compatible" content="IE=edge">** **<meta name="viewport" content="width=device-width, initial-scale=1.0">** **<title>Search Orders</title>**	**INTERFACE CHANGES**	**Describe the functionality added or Changed** - - Enhanced the admin page with a user-friendly search functionality. A new “Search Products” button/icon was added to provide quick access to a search bar, aiming to allow

```
  <!-- font awesome cdn link -->
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.0.0/css/all.min.css">

  <!-- custom css file link -->
  <link rel="stylesheet" href="css/style.css">

</head>
<body>

<div class="heading">
  <h3>Search Orders</h3>
  <p><a href="admin_page.php">Dashboard</a> / search orders</p>
</div>

<section class="search-form">
  <form action="" method="post">
    <input type="text" name="search" placeholder="Search orders by product
name or order ID..." class="box">
    <input type="submit" name="submit" value="Search" class="btn">
  </form>
</section>

<section class="Orders" style="padding-top: 0;">

  <div class="box-container">
    <?php
    if (isset($_POST['submit'])) {
      $search_item = $_POST['search'];

      // Query to search orders for the logged-in user
      $select_orders = mysqli_query($conn, "SELECT * FROM `products`
WHERE name LIKE '%$search_item%'") or die('Query failed');

      if (mysqli_num_rows($select_orders) > 0) {
        while ($fetch_order = mysqli_fetch_assoc($select_orders)) {
```



administrators to easily locate specific products within the system.

```
            ?>
            <div class="box-container">
               <?php
               // Querying for all orders based on search result
               $select_orders_inner = mysqli_query($conn, "SELECT * FROM
`orders`") or die('Query failed');
               if (mysqli_num_rows($select_orders_inner) > 0) {
                  while ($fetch_orders = mysqli_fetch_assoc($select_orders_inner))
{
                     ?>
                     <div class="box">
                        <p>user id: <span><?php echo $fetch_orders['user_id'];
?></span></p>
                        <p>placed on: <span><?php echo $fetch_orders['placed_on'];
?></span></p>
                        <p>name: <span><?php echo $fetch_orders['name'];
?></span></p>
                        <p>number: <span><?php echo $fetch_orders['number'];
?></span></p>
                        <p>email: <span><?php echo $fetch_orders['email'];
?></span></p>
                        <p>address: <span><?php echo $fetch_orders['address'];
?></span></p>
                        <p>total products: <span><?php echo
$fetch_orders['total_products']; ?></span></p>
                        <p>total price: <span>Php <?php echo
$fetch_orders['total_price']; ?>.oo</span></p>
                        <p>payment method: <span><?php echo
$fetch_orders['method']; ?></span></p>
                        <form action="" method="post">
                           <input type="hidden" name="order_id" value="<?php echo
$fetch_orders['id']; ?>">
                           <select name="update_payment">
                              <option value="" selected disabled><?php echo
$fetch_orders['payment_status']; ?></option>
                              <option value="pending">pending</option>
```

```
                        <option value="completed">completed</option>
                    </select>
                    <input type="submit" value="update" name="update_order"
class="option-btn">
                    <a href="admin_orders.php?delete=<?php echo
$fetch_orders['id']; ?>" onclick="return confirm('delete this order?');"
class="delete-btn">delete</a>
                  </form>
                </div>
                <?php
              }
            } else {
              echo '<p class="empty">No orders placed yet!</p>';
            }
            ?>
          </div>
          <?php
        }
      } else {
        echo '<p class="empty">No matching orders found!</p>';
      }
    } else {
      echo '<p class="empty">Search for an order!</p>';
    }
    ?>
  </div>
</section>

<!-- custom js file link -->
<script src="js/script.js"></script>

</body>
</html>
```

NAME: SUSAINE RIICO
INDIVIDUAL ROLE: TESTER

As a tester, I was tasked with ensuring the functionality, usability, and responsiveness of the website. This involved testing elements like navigation, forms, and interactive features for any bugs or issues. Using tools like browser developer tools and automated testing scripts connected to the HTML and CSS files, I identified and documented errors effectively. Debugging code can sometimes be challenging, especially when errors are deeply embedded or hard to replicate. To improve my testing skills, I often watch tutorials on YouTube, like those on the SDPT Solutions channel, which explain concepts in a simple and easy-to-understand way. However, when I encounter complex issues or have specific questions, I rely on AI tools for guidance and troubleshooting.