

# PLUGGED IN

**ITDBADM S12**

Lianne Balbastro  
Julianne Lammoglia  
Alyssa Mansueto  
Edriel Sta Maria



# MEET THE TEAM



maks



edie/eddy/eddiey

## BADM BADDIES

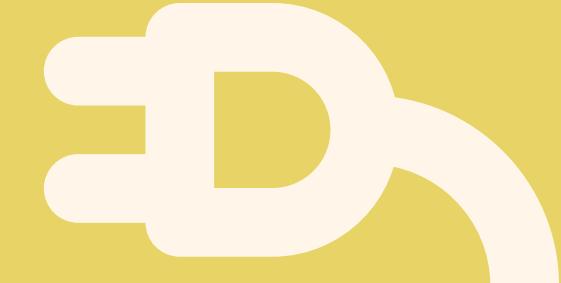


lyssa



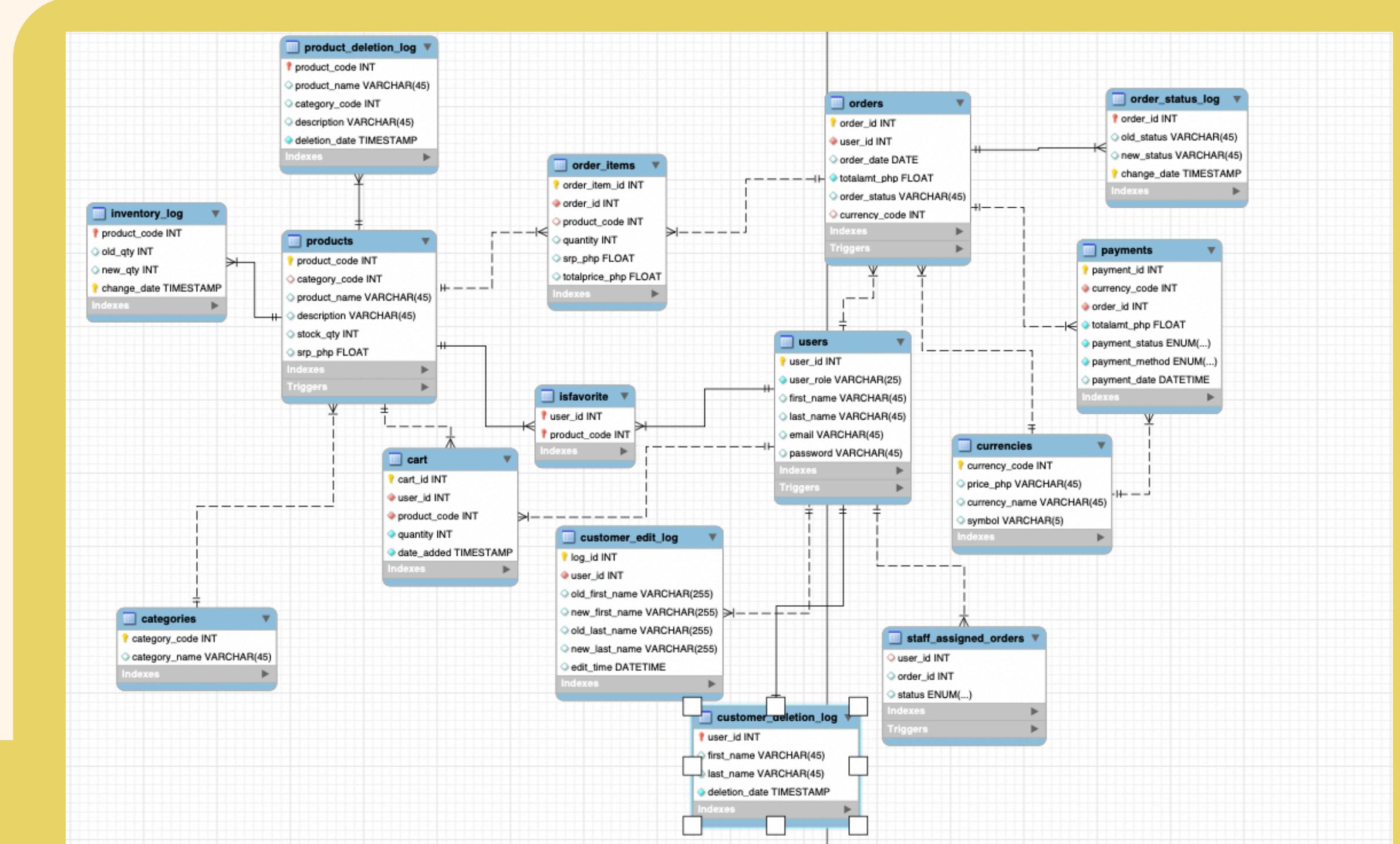
juls

# ABOUT PLUGGED-IN

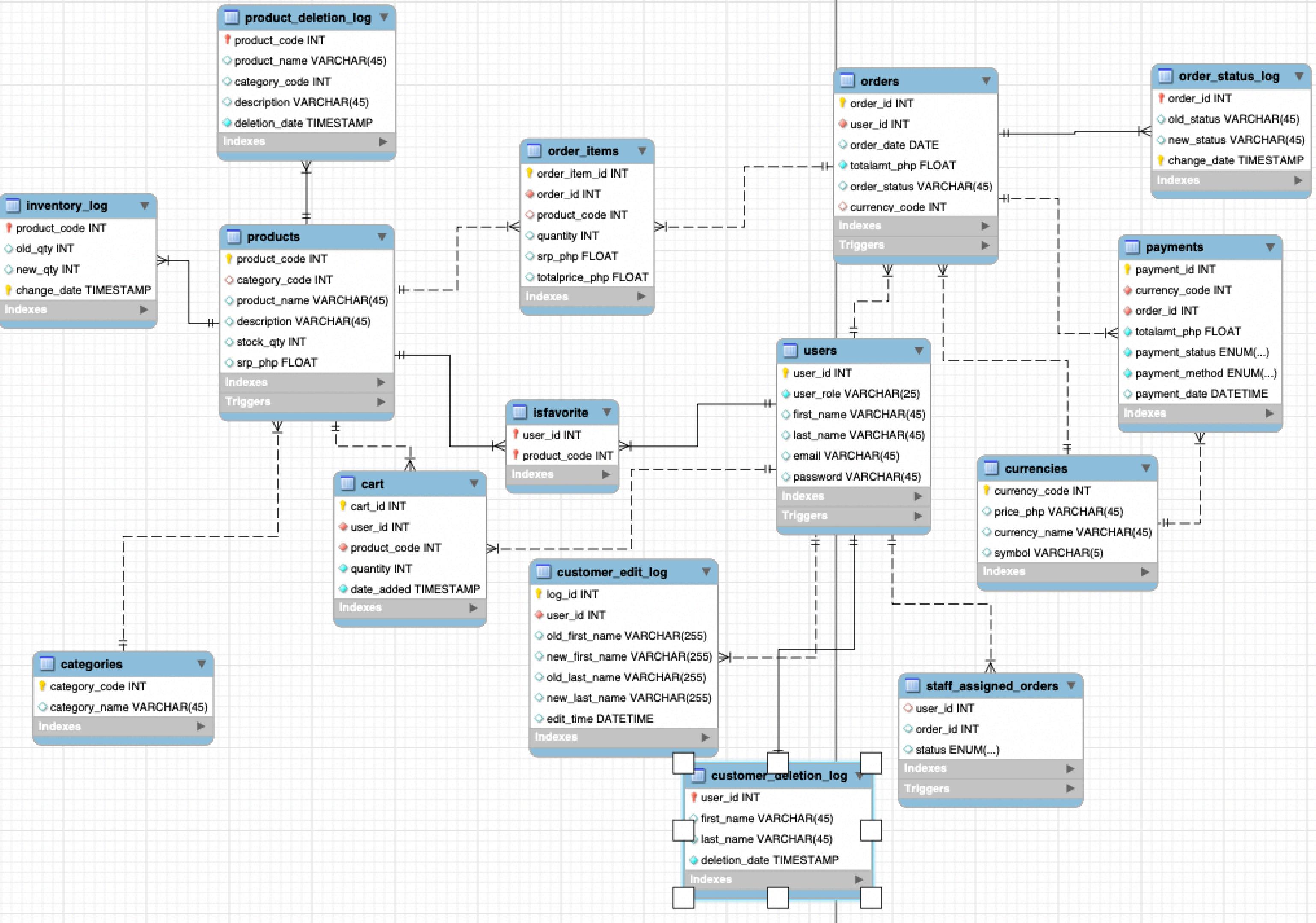


PluggedIn is an online electronics and accessories store specializing in high-quality, budget-friendly digital products. Our catalog includes Headphones, Earphones, Keyboards, Mics, Monitors, Speakers and Mice. All products are available for purchase in multiple currencies (PHP, USD, KRW), making it convenient for international users. The store will support a seamless user experience from browsing to checkout with real-time inventory tracking and transaction logging.

# ERD



- **product\_deletion\_log**
- **inventory\_log**
- **products**
- **order\_items**
- **orders**
- **payments**
- **users**
- **isfavorite**
- **cart**
- **customer\_edit\_log**
- **categories**
- **currencies**
- **staff\_assigned\_orders**
- **customer\_deletion\_log**
- **order\_status\_log**



# STORED PROCEDURES



- `add_new_product`
- `delete_customer_account`
- `delete_product`
- `update_order_status`
- `update_product_stock`

# ADD\_NEW\_PRODUCT



```
DELIMITER $$  
CREATE DEFINER=`root`@`localhost`  
PROCEDURE `add_new_product`(IN `product_name` VARCHAR(45),  
IN `category_code` INT, IN `description` VARCHAR(45),  
IN `stock_qty` INT, IN `srp_php` FLOAT)  
BEGIN  
    INSERT INTO products (product_name, category_code,  
    description, stock_qty, srp_php)  
    VALUES (product_name, category_code, description,  
    stock_qty, srp_php);  
END$$  
DELIMITER ;
```

- Procedure for adding a new product
- Used in admin.php

# **DELETE CUSTOMER\_ ACCOUNT**



```
DELIMITER $$  
  
CREATE DEFINER=`root`@`localhost`  
PROCEDURE `delete_customer_account`  
(IN `customer_id` INT)  
  
BEGIN  
  
    DELETE FROM orders WHERE user_id = customer_id;  
    DELETE FROM cart WHERE user_id = customer_id;  
    DELETE FROM isfavorite WHERE user_id = customer_id;  
    DELETE FROM users WHERE user_id = customer_id;  
  
END$$  
DELIMITER ;
```

- **Procedure for deleting a customer**
- **Used in admin.php and in edit\_user.php**
- **admin can delete customers and customers can delete their own account**

# DELETE PRODUCT



```
DELIMITER $$  
CREATE DEFINER='root'@'localhost'  
    PROCEDURE `delete_product`  
        (IN `input_product_code` INT)  
BEGIN  
    DELETE FROM cart  
        WHERE product_code = input_product_code;  
    DELETE FROM isfavorite  
        WHERE product_code = input_product_code;  
    DELETE FROM order_items  
        WHERE product_code = input_product_code;  
    DELETE FROM products  
        WHERE product_code = input_product_code;  
END$$  
DELIMITER ;
```

- Procedure for adding a deleting a product
- Used in admin.php

# UPDATE\_ ORDER\_ STATUS



```
DELIMITER $$  
CREATE DEFINER=`root`@`localhost`  
PROCEDURE `update_order_status`  
(IN `input_order_id` INT, IN `new_status` VARCHAR(45))  
BEGIN  
    UPDATE orders SET order_status = new_status  
    WHERE order_id = input_order_id;  
END$$  
DELIMITER ;
```

- Procedure for updating the status of an order
- Used in `admin.php` and `assigned_orders.php`

# UPDATE PRODUCT\_ STOCK



```
DELIMITER $$  
CREATE DEFINER=`root`@`localhost`  
PROCEDURE `update_product_stock`  
(IN `input_product_code` INT, IN `new_stock` INT)  
BEGIN  
    UPDATE products SET stock_qty = new_stock  
    WHERE product_code = input_product_code;  
END$$  
DELIMITER ;
```

- Procedure for updating the available stock of a product
- Used in admin.php and stock\_management.php

# TRIGGERS



- `check_assignedorders`
- `customer_deletion_log_trigger`
- `inventory_adjustment_trigger`
- `log_customer_edits`
- `order_status_logging_trigger`
- `prevent_negative_inventory`
- `product_deletion_log_trigger`
- `validate_order_completion`

# CHECK\_ASSIGNEDORDERS

```
DELIMITER $$  
CREATE TRIGGER `check_assignedorders`  
BEFORE INSERT ON `staff_assigned_orders`  
FOR EACH ROW  
BEGIN  
    IF EXISTS (  
        SELECT 1  
        FROM staff_assigned_orders  
        WHERE order_id = NEW.order_id  
    ) THEN  
        SIGNAL SQLSTATE '45000'  
        SET MESSAGE_TEXT = 'Order is already assigned to a staff member.';  
    END IF;  
END $$  
DELIMITER ;
```



- used to check assigned orders of a staff
- checks staff\_assigned\_orders

# CUSTOMER\_DELETION\_LOG\_TRIGGER

```
DELIMITER $$  
CREATE TRIGGER `customer_deletion_log_trigger`  
AFTER DELETE ON `users`  
FOR EACH ROW  
BEGIN  
    INSERT INTO customer_deletion_log (user_id, first_name,  
                                         last_name, deletion_date)  
    VALUES (OLD.user_id, OLD.first_name, OLD.last_name,  
             CURRENT_TIMESTAMP());  
END $$  
DELIMITER ;
```



- logs when a customer deletes their profile/account
- logs in customer\_deletion\_log

# INVENTORY\_ADJUSTMENT\_TRIGGER

```
DELIMITER $$  
CREATE TRIGGER `inventory_adjustment_trigger`  
AFTER UPDATE ON `products`  
FOR EACH ROW  
BEGIN  
    IF OLD.stock_qty != NEW.stock_qty THEN  
        INSERT INTO inventory_log (product_code, old_qty, new_qty, change_date)  
        VALUES (NEW.product_code, OLD.stock_qty, NEW.stock_qty, current_timestamp());  
    END IF;  
END $$  
DELIMITER ;
```



- logs when staff/admin changes inventory of a product
- logs in inventory\_log

# LOG\_CUSTOMER\_EDITS

- logs when a customer edits their name on their profile
- logs in customer\_edit\_log



```
DELIMITER $$  
CREATE TRIGGER `log_customer_edits`  
AFTER UPDATE ON `users`  
FOR EACH ROW  
BEGIN  
    IF LOWER(OLD.user_role) = 'customer' AND (  
        OLD.first_name <> NEW.first_name OR  
        OLD.last_name <> NEW.last_name  
    ) THEN  
        INSERT INTO customer_edit_log (  
            user_id,  
            old_first_name,  
            new_first_name,  
            old_last_name,  
            new_last_name  
        ) VALUES (  
            OLD.user_id,  
            OLD.first_name,  
            NEW.first_name,  
            OLD.last_name,  
            NEW.last_name  
        );  
    END IF;  
END $$  
DELIMITER ;
```

# ORDER\_STATUS\_LOGGING\_TRIGGER

```
DELIMITER $$  
CREATE TRIGGER `order_status_logging_trigger`  
AFTER UPDATE ON `orders`  
FOR EACH ROW  
BEGIN  
    IF OLD.order_status != NEW.order_status THEN  
        INSERT INTO order_status_log (order_id, old_status, new_status, change_date)  
        VALUES (NEW.order_id, OLD.order_status, NEW.order_status, NOW());  
    END IF;  
END $$  
DELIMITER ;
```



- logs when the order status of an order is changed
- logs in order\_status\_log

# PREVENT\_NEGATIVE\_INVENTORY\_TRIGGER

```
DELIMITER $$  
CREATE TRIGGER `prevent_negative_inventory`  
BEFORE UPDATE ON `products`  
FOR EACH ROW  
BEGIN  
    -- Ensure that the new stock quantity is not negative  
    IF NEW.stock_qty < 0 THEN  
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Stock quantity cannot be negative';  
    END IF;  
END $$  
DELIMITER ;
```



- prevents negative inventory and blocks data from being inserted in the products table

# PRODUCT\_DELETION\_LOG\_TRIGGER

```
DELIMITER $$  
CREATE TRIGGER `product_deletion_log_trigger`  
AFTER DELETE ON `products`  
FOR EACH ROW  
BEGIN  
    INSERT INTO product_deletion_log (  
        product_code, product_name, category_code, description, deletion_date)  
    VALUES (  
        OLD.product_code, OLD.product_name, OLD.category_code, OLD.description, CURRENT_TIMESTAMP());  
END $$  
DELIMITER ;
```



- logs when a product has been deleted by the admin
- logs in product\_deletion\_log

# PRODUCT\_DELETION\_LOG\_TRIGGER

```
DELIMITER $$  
CREATE TRIGGER `validate_order_completion`  
BEFORE UPDATE ON `staff_assigned_orders`  
FOR EACH ROW  
BEGIN  
    IF NEW.status = 'COMPLETED' THEN  
        IF NOT EXISTS (  
            SELECT 1  
            FROM orders  
            WHERE order_id = NEW.order_id  
            AND order_status = 'Delivered'  
        ) THEN  
            SIGNAL SQLSTATE '45000'  
            SET MESSAGE_TEXT = 'Cannot mark order as COMPLETED. Order must be in Delivered status first.';  
        END IF;  
    END IF;  
END $$  
DELIMITER ;
```



- ensures that an order cannot be marked as **COMPLETED** if not marked off as delivered first



GUI

GUI

GUI

# SIGN IN & REGISTER

## Welcome to PluggedIn

Sign in to your account or create a new one

[→ Sign In](#) [👤 Register](#)

Email

Password  
 [👁](#)

[Forgot password?](#)

[→ Sign In](#)

## Welcome to PluggedIn

Sign in to your account or create a new one

[→ Sign In](#) [👤 Register](#)

First Name  Last Name

Email

Password (minimum 6 characters)  
 [👁](#)

Confirm Password  
 [👁](#)

ⓘ Your account will be created as a Customer account

I agree to the [Terms of Service](#) and [Privacy Policy](#)

[👤 Create Account](#)

# INDEX PAGE + ADD TO CART

**PLUGGED**

Search products... 

P PHP ▾   

**PLUGGED**

Search products... 

P PHP ▾   



Discover the latest in audio, input devices, and display technology. Quality gear for professionals and enthusiasts.

12+ Premium Products    5 Categories    3 Currencies

**Browse Products**

All Products    Headphones    Keyboards    Mice    Monitors    Speakers

**PLUGGED**

Search products... 

P PHP ▾   



Product Image

Headphones  
**Airpods Max**  
Wireless Headphones  
₱35,000.00  
102 in stock



Product Image

Speakers  
**Bose SoundLink Revolve+**  
Portable Bluetooth Speaker  
₱9,000.00  
6 in stock



Product Image

Keyboards  
**Corsair K95 RGB Platinum**  
Mechanical Gaming Keyboard  
₱8,000.00  
1201 in stock



Product Image

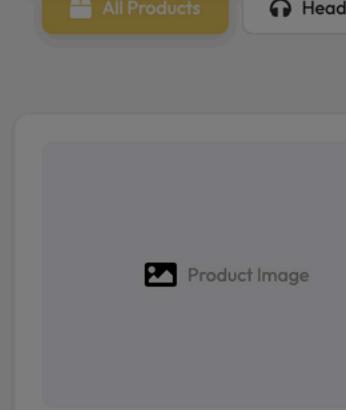
Speakers  
**JBL Flip 5**  
Portable Bluetooth Speaker  
₱7,000.00  
99 in stock

**Add to Cart** **Add to Cart** **Add to Cart** **Add to Cart**

**PLUGGED**

Search products... 

P PHP ▾   



Product Image

Headphones  
**Airpods Max**  
Wireless Headphones  
₱35,000.00  
102 in stock

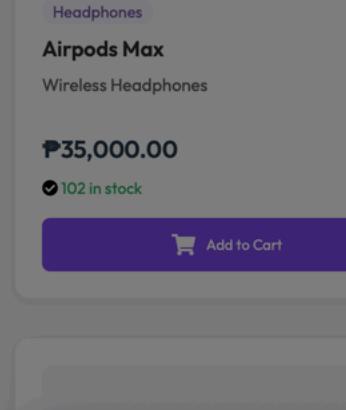
**Add to Cart**

**Product Details**

**Corsair K95 RGB Platinum**  
Mechanical Gaming Keyboard  
₱8,000  
Stock: 1201 available

Quantity: - 1 +

**Add to Cart** **Heart**



Product Image

Speakers  
**JBL Flip 5**  
Portable Bluetooth Speaker  
₱7,000.00  
99 in stock

**Add to Cart**

**PLUGGED**

Search products... 

P PHP ▾   



Discover the latest in audio, input devices, and display technology. Quality gear for professionals and enthusiasts.

12+ Premium Products    5 Categories    3 Currencies

**Browse Products**

All Products    Headphones    Keyboards    Mice    Monitors    Speakers

**Shopping Cart**

**Corsair K95 RGB Platinum** TechPeripherals ₱8,000.00  
- 2 +

Subtotal ₱16000.00  
Total ₱16000.00

**PROCEED TO CHECKOUT**

# FAVORITES

PLUGGED  
IN

Search products...



P PHP ▾



PLUGGED  
IN

Search products...



P PHP ▾



## My Favorites



Product Image

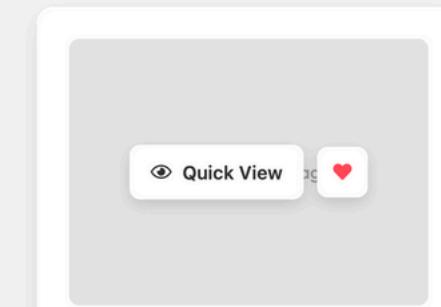
Headphones

Airpods Max

Wireless Headphones

₱35,000.00

102 in stock



Quick View



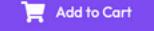
Speakers

Bose SoundLink Revolve+

Portable Bluetooth Speaker

₱9,000.00

6 in stock



Product Image

Speakers

JBL Flip 5

Portable Bluetooth Speaker

₱7,000.00

99 in stock



Product Image

Keyboards

Logitech MX Keys

Wireless Keyboard

₱5,000.00

39 in stock



## My Favorites



### No favorites yet

Start browsing our products and click the heart icon to add items to your favorites list.



© 2025 PluggedIn. All rights reserved.

Search products...



## Product Details

Product Image



**Bose SoundLink Revolve+**

Portable Bluetooth Speaker

**₱9,000.00**

Stock: 799 available

Quantity:



Product Image

Speakers

Bose SoundLink Revolve+

Portable Bluetooth Speaker

₱9,000.00

799 in stock



PLUGGED  
IN

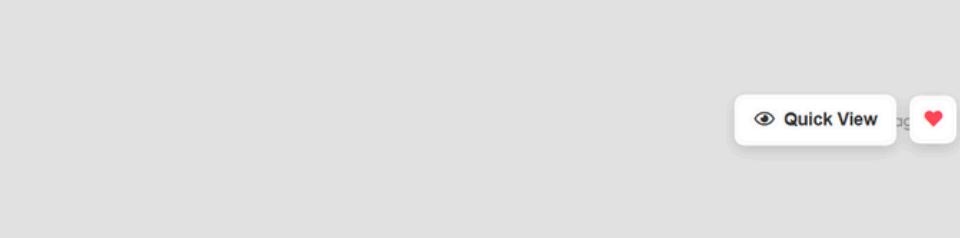
Search products...



P PHP ▾



## My Favorites



Quick View



Speakers

**Bose SoundLink Revolve+**

Portable Bluetooth Speaker

**₱9,000.00**

799 in stock



© 2025 PluggedIn. All rights reserved.

# PAYMENT

← Back to Home

## Checkout

### Payment Method

Card

E-Wallet

Cash

Card Number

1234 5678 9012 3456

Cardholder Name

John Doe

Expiry Date

MM/YY

CVV

123

Billing Address

123 Main St, City, State 12345

### Shipping Information

First Name

John

Last Name

Doe

Address

Street address

City

Las Piñas

Postal Code

1740

Phone Number

+83 912 345 6789

### Payment Method

Card

E-Wallet

Cash

E-Wallet Provider

Select your e-wallet

Phone Number / Account

+83 912 345 6789

You will be redirected to your e-wallet app to complete the payment.

## Checkout

### Payment Method

Card

E-Wallet

Cash

Pickup Location

Select pickup location

Preferred Pickup Date

mm / dd / yyyy

Preferred Pickup Time

Select time slot

Please bring exact change when picking up your order. A confirmation will be sent to your email.

### Order Summary

Subtotal

₱9,000.00

Total

₱9,000.00

Complete Payment

Secure Checkout

How to Buy a House Newsletter

← Back to Home

## Checkout

### Payment Method

Card

E-Wallet

Cash

Pickup Location

Main Branch - BGC

Preferred Pickup Date

07 / 28 / 2025

Preferred Pickup Time

### localhost says

Order confirmed! Order ID: 2

Total: ₱9000

You will receive a confirmation email shortly.

OK

### Summary

Subtotal

₱9,000.00

Total

₱9,000.00

Processing...

Secure Checkout

Codeacademy

Gmail

# USER PAGE + EDIT USER

[← Back to Home](#)

**Max Balbastro**  
Member since 7/27/2025

[Profile](#) [Order History](#)

**Profile Information**

Full Name: Max Balbastro [Edit](#) [Delete](#)

Email: maxbalbastro@gmail.com [Edit](#) [Delete](#)

[Logout](#) [Edit Profile](#)

**Max Balbastro**  
Member since 7/27/2025

[Profile](#) [Order History](#)

**Edit Profile**

First Name: Maxene [Edit](#)

Last Name: Balbastro [Edit](#)

[Save Changes](#) [Delete Account](#)

Full Name: Max Balbastro [Edit](#) [Delete](#)

Email: maxbalbastro@gmail.com [Edit](#) [Delete](#)

[Logout](#) [Edit Profile](#)

[← Back to Home](#)

**Max Balbastro**  
Member since 7/27/2025

[Profile](#) [Order History](#)

**Order History**

**Order #1**  
Jul 27, 2025  
**₱25,000.00**

Ordered Jul 27 Processing Shipped Delivered DELIVERED PROCESSING

[← Back to Home](#)

**Maxene Balbastro**  
Member since 7/27/2025

[Profile](#) [Order History](#)

**Profile Information**

Full Name: Maxene Balbastro [Edit](#) [Delete](#)

Email: maxbalbastro@gmail.com [Edit](#) [Delete](#)

[Logout](#) [Edit Profile](#)

# STAFF PAGE

## Staff Dashboard

Welcome to Staff Dashboard

Select a tab above to manage your tasks:

- Stock Management**  
Update inventory levels and monitor stock status
- Assigned Orders**  
View and manage orders assigned to you
- Available Orders**  
Browse and pick up available orders

Go to Customer View | Logout

## Staff Dashboard

Stock Management Assigned Orders Available Orders

### My Assigned Orders

ORDER ID	CUSTOMER	TOTAL	STATUS	ITEMS	ORDER DATE	ACTIONS
1	Maxene Balbastro	₱25,000.00	Processing	LG UltraGear 27GN950 (x1)	2025-07-27	

Go to Customer View | Logout

## Staff Dashboard

Stock Management Assigned Orders Available Orders

### Set Stock Level

Product: Select product | New Stock Level: 0 | Set Stock Level

Go to Customer View | Logout

## Staff Dashboard

Stock Management Assigned Orders Available Orders

### Available Orders to Pick Up

ORDER ID	CUSTOMER	TOTAL	ORDER DATE	ACTIONS
2	Carla Reyes	₱9,000.00	2025-07-27 1 item(s)	

Go to Customer View | Logout

## Current Stock Levels

PRODUCT CODE	PRODUCT NAME	CATEGORY	CURRENT STOCK	PRICE (PHP)
6	Airpods Max	Headphones	102	₱35,000.00
10	Bose SoundLink Revolve+	Speakers	798	₱9,000.00
8	Corsair K95 RGB Platinum	Keyboards	1201	₱8,000.00

# ADMIN PAGE

## Admin Dashboard

[Logout](#)

## Admin Dashboard

[Logout](#)[Products](#)[Stock](#)[Staff & Users](#)[Orders](#)

### + Add New Product (Using Stored Procedure)

[Add Product](#)

### Product List

CODE	NAME	CATEGORY	STOCK	PRICE (PHP)	DESCRIPTION	ACTIONS
10	Bose SoundLink Revolve+	Speakers	798	₱9,000.00	Portable Bluetooth Speaker	
9	Logitech G502 HERO	Mice	1000	₱4,000.00	High-Performance Gaming Mouse	

## Admin Dashboard

[Logout](#)[Products](#)[Stock](#)[Staff & Users](#)[Orders](#)

### Update Stock (Using Stored Procedure)

[Update Stock](#)

### Low Stock Alert

[PRODUCT](#)[CURRENT STOCK](#)[CATEGORY](#)[ACTIONS](#)

JBL Flip 5

1

Speakers

100

[+ Restock](#)

## Admin Dashboard

[Products](#)[Stock](#)[Staff & Users](#)[Orders](#)

### + Add New Staff

[Add Staff Member](#)

### Staff List

ID	NAME	EMAIL	ROLE	ACTIONS
4	Carla Reyes	carla_reyes@dlsu.edu.ph	Staff	

### ✖ Delete Customer Account

[Delete Customer Account](#)

## Admin Dashboard

[Products](#)[Stock](#)[Staff & Users](#)[Orders](#)

### Order Management

ORDER ID	CUSTOMER	TOTAL (PHP)	ORDER STATUS	PAYMENT STATUS	PAYMENT METHOD	DATE	ACTIONS
2	Carla Reyes	₱9,000.00	PROCESSING	UNPAID	CASH	2025-07-27	<input type="button" value="Processing"/>
1	Maxene Balbastro	₱25,000.00	PROCESSING	UNPAID	CASH	2025-07-27	<input type="button" value="Processing"/>

# SQL SNIPPETS

## ATOMICITY

If a user were to terminate in the middle of the payment process, the payments will not push through, and the items in the order will remain in the cart of the user.

```
// Create payment record
$payment_status = ($payment_method === 'cash') ? 'unpaid' : 'paid';
$payment_date = ($payment_method === 'cash') ? NULL : date('Y-m-d H:i:s');

$payment_query = "INSERT INTO payments (currency_code, order_id, totalamt_php,
                                         | | | | | | | | | |
                                         | | | | | | | | | |
                                         VALUES (?, ?, ?, ?, ?, ?, ?)";

$stmt = $conn->prepare($payment_query);
$stmt->bind_param("iidsss", $currency_code, $order_id, $total_amount, $payment_
$stmt->execute();

// Clear cart
$clear_cart = "DELETE FROM cart WHERE user_id = ?";
$stmt = $conn->prepare($clear_cart);
$stmt->bind_param("i", $user_id);
$stmt->execute();

$conn->commit();
```

# SQL SNIPPETS

## CONSISTENCY

Negative inventory cannot be set by staff or Admin. This is ensured through the trigger prevent\_negative\_inventory.

```
DELIMITER $$  
CREATE TRIGGER `prevent_negative_inventory`  
BEFORE UPDATE ON `products`  
FOR EACH ROW  
BEGIN  
    -- Ensure that the new stock quantity is not negative  
    IF NEW.stock_qty < 0 THEN  
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Stock quantity cannot be negative';  
    END IF;  
END $$  
DELIMITER ;
```

# SQL SNIPPETS

## ISOLATION

Isolation was added to the website by implementing a formatPrice function. This function is called to ensure that users see the converted price before proceeding.

```
if (!function_exists(function: 'formatPrice')) {
    8 references
    function formatPrice($php_price, $currency_data): string {
        // price_php in the database should be the conversion rate FROM PHP
        // For PHP: rate = 1.0, for USD: rate = 0.018 (1 PHP = 0.018 USD), for KRW: rate = 24.5 (1 PHP = 24.5 KRW)
        $converted_price = convertPrice(PHP: $php_price, conversion_rate: $currency_data['price_php']);
        return $currency_data['symbol'] . number_format(num: $converted_price, decimals: 2);
    }
}
?>
```

```
while($row = $result->fetch_assoc()) {
    $items[] = [
        'cart_id' => $row['cart_id'],
        'product_code' => $row['product_code'],
        'name' => $row['product_name'],
        'quantity' => $row['quantity'],
        'price' => formatPrice(PHP: $row['srp_php'], currency_data: $current_currency),
        'total_price' => number_format(num: $row['total_price'], decimals: 2),
        'stock_qty' => $row['stock_qty']
    ];
    $total += $row['total_price'];
}
```

# SQL SNIPPETS

## DURABILITY

Durability is enforced through the majority of our triggers. For example, our `order_status_logging_trigger` ensures status transitions are recorded.

```
DELIMITER $$  
CREATE TRIGGER `order_status_logging_trigger`  
AFTER UPDATE ON `orders`  
FOR EACH ROW  
BEGIN  
    IF OLD.order_status != NEW.order_status THEN  
        INSERT INTO order_status_log (order_id, old_status, new_status, change_date)  
        VALUES (NEW.order_id, OLD.order_status, NEW.order_status, NOW());  
    END IF;  
END $$  
DELIMITER ;
```

THANK YOU

THANK YOU

THANK YOU

**PLUGGED  
IN**