

MONTCLAIR STATE

UNIVERSITY

StarNaVs La: Hollywood La Based eCommerce Website

**Final Application Project
CSIT: 555 Database Systems**

Nirav Parmar : Parmarn2@montclair.edu

Vivek Kumar Mamidi: mamidiv1@montclair.edu

Shweta Sah: sahs2@montclair.edu

Amishaben Natavarbahi Chaudhari: chaudharia1@montclair.edu

System User Accounts

Business Manager: manager@starnavs.la

Username: manager

Password: 123456

Inventory View: inventory@starnavs.la

Username: inventory

Password: 123456

Fulfillment View : fulfillment@starnavs.la

Username: fulfillment

Password: 123456

Editor View : Username: editor

Password: 123456

User View : Customer : john@example.com

Username: john_doe

Password: 123456

User View : Customer : jane@example.com

Username: jane_smith

Password: 123456

Executive Summary

StarNaVs LA is an ecommerce site that focuses on Hollywood culture, where purchasing products as well as educating viewers through a media component are incorporated. The website is completely database-driven, constructed using PHP, MySQL, and authentication. The site also has a customer section where viewers are able to view products, make purchases, and even access their receipts. Additionally, it has a human resources component that accommodates various employee types.

Client Need Analysis:

Project response to a client in need of an online store uniquely themed as Hollywood-themed. This system should be able to perform tasks such as storing educational content, handling multiple user roles, maintaining inventory, showing products, ordering products, creating receipts, and logging employee activities. Encryption of passwords, protection against SQL-injection attacks, and a fully secured login landing page is also required.

Data Views & SQL Demonstrations

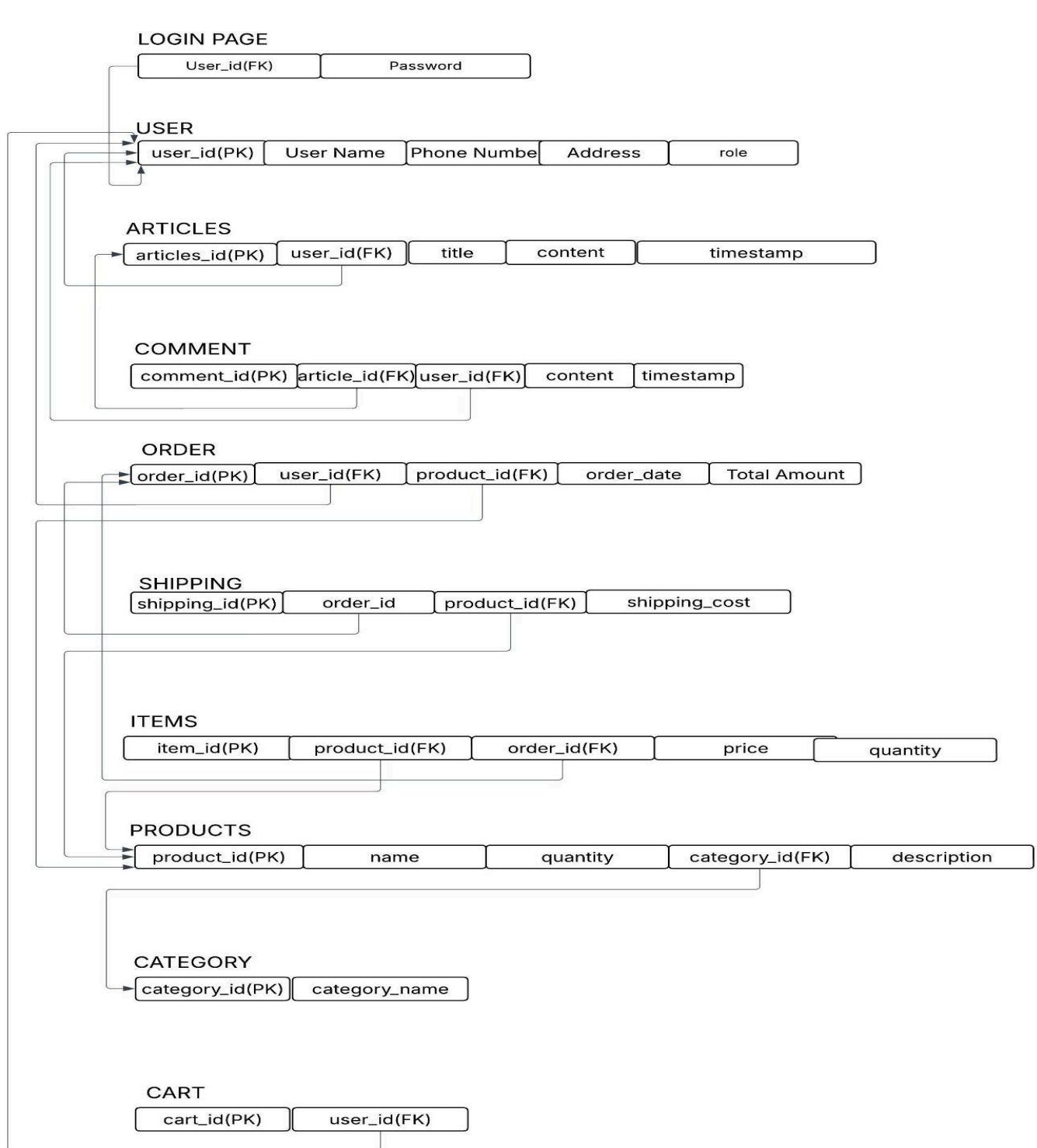
The five major views in this system are Inventory View, Business Manager View, Employee View, Editor View and Customer View. Each of these views performs data retrieval based on access permission in a secure way with prepared SQL statements.

Business Plan:

StarNaVs LA combines fashion, décor, accessories, limited-edition collectibles, and movie-based items inspired by the entertainment culture of Los Angeles. The education section teaches users about the history of Hollywood, its landmarks, and film evolution. On a business note, this plan is to bring a premium Hollywood experience to all customers while keeping order processing as efficient as possible, employee management, and a polished UI.

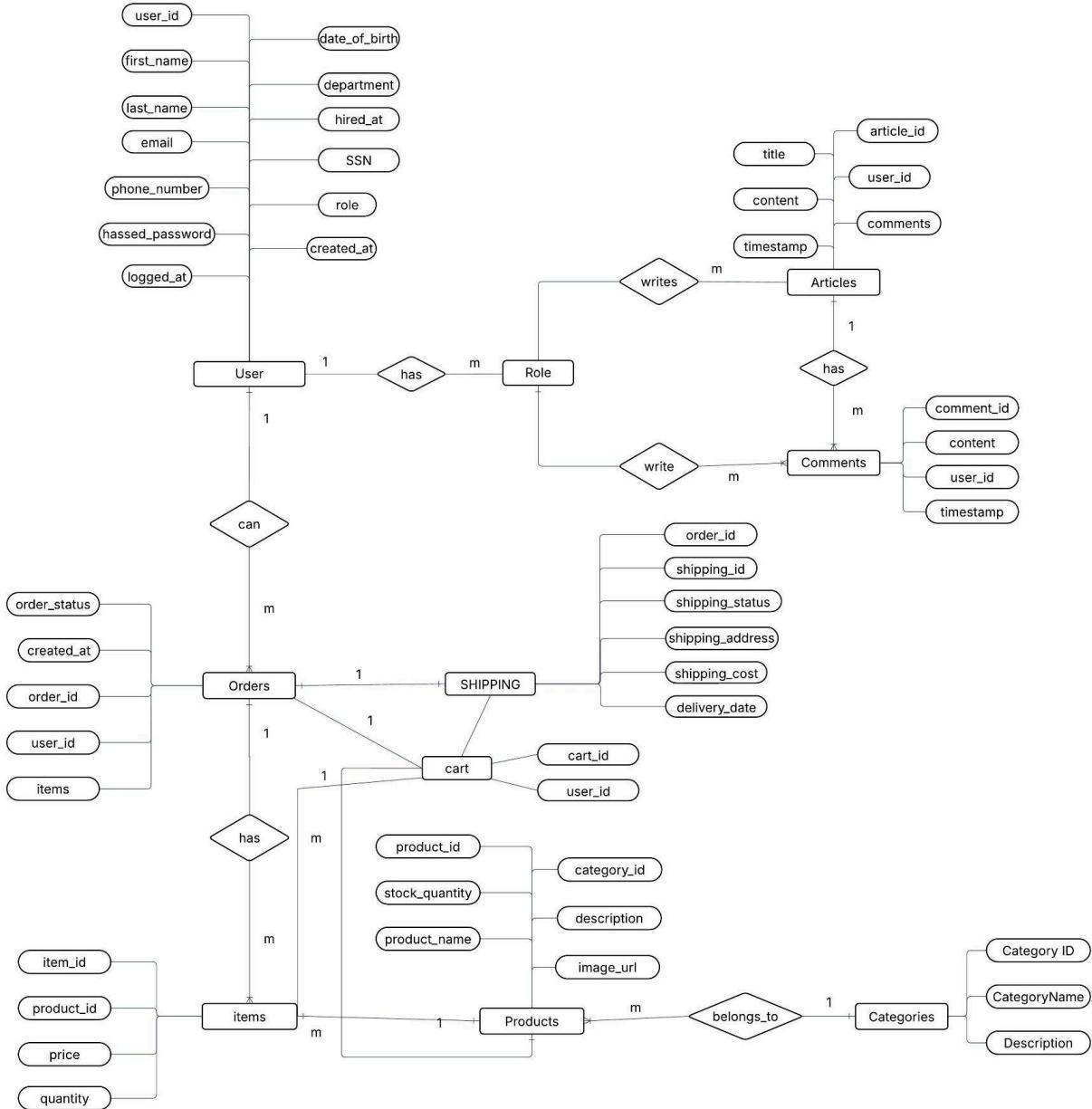
Conceptual Data Model:

The conceptual model was developed by the identification of key entities such as Users, Employees, Customers, Products, Categories, Orders, Order Items, Inventory, and Articles. Throughout the semester, the model was also evolved to support additional HR features, role based access, and receipt tracking.



Implementation Data Model:

The implementation of the ER model usually formalizes the relationships using the primary keys, foreign keys, and third normal form design. Order processing is also handled through the order and order_items tables with strict inventory enforcement.



Assumptions

- ❖ **Users**
 - Every user of the system should have a user account.
 - Each user will be having one role assigned to them such as Admin or the HR.
 - Every user will have a unique email. No two people can log in by using the same email.
- ❖ **Customers**
 - A customer is also a user of the system.
 - One customer can place many orders or never place an order.
 - The customer's phone number and addresses are stored in the Customers table.

- ❖ Employees
 - Every employee is also a user.
 - Additional employee information includes department, hire date, and SSN.
 - Writing on the website can only be done by the employees.
- ❖ Orders
 - Only customers can place the orders.
 - One customer places multiple orders.
 - One order can have many items.
 - Order status is stored in plain text.
- ❖ Order Items
 - Every order item corresponds to only one product.
 - It stores the price at the time of purchase, so old orders remain accurate even when prices are changed later.
- ❖ Products and Categories
 - Every product belongs to one category.
 - One category can have many products.
 - Each product has details such as the name, price, image, and stock quantity.
- ❖ Articles and Comments
 - Article-writers are employees only.
 - Every user can comment on the articles no matter whether they are a customer or an employee.
 - One article can have many comments and each of the comments belongs to one article.
- ❖ Keys and IDs
 - The system automatically generates all these IDs, including the UserID, OrderID, and the ProductID.
 - Foreign keys will ensure the relationships between the tables are valid.
- ❖ Inventory
 - Each product will have an inventory record.
 - Inventory will tell us the number of units currently on hand for a product.
 - Inventory is updated for every sale and restocking of a product.
 - Each record in inventory has an ID.
- ❖ Shipping
 - Shipping details are stored separately for every order.
 - Shipping includes the method, address, status, cost, and tracking number, if they are available.
 - The date of delivery is recorded when the order is delivered.
 - Every shipping record is related to one specific order.
- ❖ Cart
 - Each customer has a cart that holds items that will be purchased by them.
 - The cart shows the total price of everything added to it.
 - The cart also stores the date when it was last updated.
 - Each cart has an ID.
- ❖ General Assumptions
 - A person can be both a customer and also can be an employee at the same time.
 - All the dates are kept in the format of YYYY-MM-DD Users bearing important data like orders or comments cannot be deleted from the system.

Database Structure:

Recent | Favorites

| Table | Action | Rows | Type | Collation | Size | Overhead | |
|------------------|--|------|--------|-------------------|------------------------------|-----------|-----|
| articles | Browse Structure Search Insert Empty Drop | 5 | InnoDB | latin1_swedish_ci | 96.0 KiB | - | |
| article_comments | Browse Structure Search Insert Empty Drop | 5 | InnoDB | latin1_swedish_ci | 48.0 KiB | - | |
| audit_log | Browse Structure Search Insert Empty Drop | 73 | InnoDB | latin1_swedish_ci | 64.0 KiB | - | |
| categories | Browse Structure Search Insert Empty Drop | 5 | InnoDB | latin1_swedish_ci | 48.0 KiB | - | |
| customers | Browse Structure Search Insert Empty Drop | 3 | InnoDB | latin1_swedish_ci | 48.0 KiB | - | |
| employees | Browse Structure Search Insert Empty Drop | 3 | InnoDB | latin1_swedish_ci | 64.0 KiB | - | |
| inventory | Browse Structure Search Insert Empty Drop | 30 | InnoDB | latin1_swedish_ci | 48.0 KiB | - | |
| orders | Browse Structure Search Insert Empty Drop | 6 | InnoDB | latin1_swedish_ci | 80.0 KiB | - | |
| order_items | Browse Structure Search Insert Empty Drop | 10 | InnoDB | latin1_swedish_ci | 48.0 KiB | - | |
| password_resets | Browse Structure Search Insert Empty Drop | 0 | InnoDB | latin1_swedish_ci | 80.0 KiB | - | |
| products | Browse Structure Search Insert Empty Drop | 30 | InnoDB | latin1_swedish_ci | 112.0 KiB | - | |
| roles | Browse Structure Search Insert Empty Drop | 6 | InnoDB | latin1_swedish_ci | 32.0 KiB | - | |
| shopping_cart | Browse Structure Search Insert Empty Drop | 1 | InnoDB | latin1_swedish_ci | 48.0 KiB | - | |
| users | Browse Structure Search Insert Empty Drop | 8 | InnoDB | latin1_swedish_ci | 96.0 KiB | - | |
| user_sessions | Browse Structure Search Insert Empty Drop | 0 | InnoDB | latin1_swedish_ci | 48.0 KiB | - | |
| 15 tables | Sum | | | | 185 MyISAM latin1_swedish_ci | 960.0 KiB | 0 B |

Check all With selected: ▾

Recent | Favorites

Browse | Structure | SQL | Search | Insert | Export | Import | Operations | Triggers

Table structure Relation view

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|----|----------------|--------------|-------------------|-----------------------------|------|-------------------|----------|-----------------------------|-------------------|
| 1 | article_id | int(11) | latin1_swedish_ci | | No | None | | AUTO_INCREMENT | Change Drop More |
| 2 | title | varchar(255) | latin1_swedish_ci | | No | None | | | Change Drop More |
| 3 | slug | varchar(255) | latin1_swedish_ci | | No | None | | | Change Drop More |
| 4 | content | text | latin1_swedish_ci | | No | None | | | Change Drop More |
| 5 | summary | text | latin1_swedish_ci | | Yes | NULL | | | Change Drop More |
| 6 | author_id | int(11) | | | No | None | | | Change Drop More |
| 7 | category | varchar(100) | latin1_swedish_ci | | Yes | NULL | | | Change Drop More |
| 8 | featured_image | varchar(255) | latin1_swedish_ci | | Yes | NULL | | | Change Drop More |
| 9 | view_count | int(11) | | | Yes | 0 | | | Change Drop More |
| 10 | is_published | tinyint(1) | | | Yes | 0 | | | Change Drop More |
| 11 | published_at | timestamp | | | Yes | NULL | | | Change Drop More |
| 12 | created_at | timestamp | | | No | CURRENT_TIMESTAMP | | | Change Drop More |
| 13 | updated_at | timestamp | | on update CURRENT_TIMESTAMP | No | CURRENT_TIMESTAMP | | ON UPDATE CURRENT_TIMESTAMP | Change Drop More |

Check all With selected: Browse Change Drop Primary Unique Index Spatial Fulltext

Recent | Favorites

Table structure | Relation view

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|--------------|------------|-------------------|------------|------|-------------------|----------|----------------|--------------------|
| 1 | comment_id | int(11) | | | No | None | | AUTO_INCREMENT | Change Drop More |
| 2 | article_id | int(11) | | | No | None | | | Change Drop More |
| 3 | customer_id | int(11) | | | No | None | | | Change Drop More |
| 4 | comment_text | text | latin1_swedish_ci | | No | None | | | Change Drop More |
| 5 | is_approved | tinyint(1) | | | Yes | 1 | | | Change Drop More |
| 6 | created_at | timestamp | | | No | CURRENT_TIMESTAMP | | | Change Drop More |

Check all With selected:

1 column(s) after created_at

Indexes

| Action | Keyname | Type | Unique | Packed | Column | Cardinality | Collation | Null | Comment |
|--------------------|--------------|-------|--------|--------|-------------|-------------|-----------|------|---------|
| Edit Rename Drop | PRIMARY | BTREE | Yes | No | comment_id | 5 | A | No | |
| Edit Rename Drop | idx_article | BTREE | No | No | article_id | 4 | A | No | |
| Edit Rename Drop | idx_customer | BTREE | No | No | customer_id | 3 | A | No | |

Recent | Favorites

Table structure | Relation view

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|----|------------|--------------|-------------------|------------|------|-------------------|----------|----------------|--------------------|
| 1 | log_id | int(11) | | | No | None | | AUTO_INCREMENT | Change Drop More |
| 2 | user_id | int(11) | | | Yes | NULL | | | Change Drop More |
| 3 | action | varchar(100) | latin1_swedish_ci | | No | None | | | Change Drop More |
| 4 | table_name | varchar(50) | latin1_swedish_ci | | Yes | NULL | | | Change Drop More |
| 5 | record_id | int(11) | | | Yes | NULL | | | Change Drop More |
| 6 | old_value | text | latin1_swedish_ci | | Yes | NULL | | | Change Drop More |
| 7 | new_value | text | latin1_swedish_ci | | Yes | NULL | | | Change Drop More |
| 8 | ip_address | varchar(45) | latin1_swedish_ci | | Yes | NULL | | | Change Drop More |
| 9 | user_agent | varchar(255) | latin1_swedish_ci | | Yes | NULL | | | Change Drop More |
| 10 | created_at | timestamp | | | No | CURRENT_TIMESTAMP | | | Change Drop More |

Check all With selected:

Recent | Favorites

Table structure | Relation view

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|---------------|--------------|-------------------|------------|------|-------------------|----------|----------------|--------------------|
| 1 | category_id | int(11) | | | No | None | | AUTO_INCREMENT | Change Drop More |
| 2 | category_name | varchar(100) | latin1_swedish_ci | | No | None | | | Change Drop More |
| 3 | description | text | latin1_swedish_ci | | Yes | NULL | | | Change Drop More |
| 4 | image_url | varchar(255) | latin1_swedish_ci | | Yes | NULL | | | Change Drop More |
| 5 | is_active | tinyint(1) | | | Yes | 1 | | | Change Drop More |
| 6 | created_at | timestamp | | | No | CURRENT_TIMESTAMP | | | Change Drop More |

Check all With selected:

1 column(s) after created_at

Table structure

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|----|---------------|--------------|-------------------|------------|------|-------------------|----------|----------------|--------|
| 1 | customer_id | int(11) | | | No | None | | AUTO_INCREMENT | |
| 2 | user_id | int(11) | | | No | None | | | |
| 3 | first_name | varchar(50) | latin1_swedish_ci | | No | None | | | |
| 4 | last_name | varchar(50) | latin1_swedish_ci | | No | None | | | |
| 5 | phone | varchar(20) | latin1_swedish_ci | | Yes | NULL | | | |
| 6 | address_line1 | varchar(255) | latin1_swedish_ci | | Yes | NULL | | | |
| 7 | address_line2 | varchar(255) | latin1_swedish_ci | | Yes | NULL | | | |
| 8 | city | varchar(100) | latin1_swedish_ci | | Yes | NULL | | | |
| 9 | state | varchar(50) | latin1_swedish_ci | | Yes | NULL | | | |
| 10 | zip_code | varchar(20) | latin1_swedish_ci | | Yes | NULL | | | |
| 11 | country | varchar(100) | latin1_swedish_ci | | Yes | USA | | | |
| 12 | created_at | timestamp | | | No | CURRENT_TIMESTAMP | | | |

Check all With selected:

Table structure

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|----|---------------|--|-------------------|------------|------|-------------------|----------|----------------|--------|
| 1 | employee_id | int(11) | | | No | None | | AUTO_INCREMENT | |
| 2 | user_id | int(11) | | | No | None | | | |
| 3 | first_name | varchar(50) | latin1_swedish_ci | | No | None | | | |
| 4 | last_name | varchar(50) | latin1_swedish_ci | | No | None | | | |
| 5 | employee_type | enum('Inventory Manager', 'Order Fulfillment Rep') | latin1_swedish_ci | | No | None | | | |
| 6 | phone | varchar(20) | latin1_swedish_ci | | Yes | NULL | | | |
| 7 | hire_date | date | | | No | None | | | |
| 8 | salary | decimal(10,2) | | | Yes | NULL | | | |
| 9 | is_active | tinyint(1) | | | Yes | 1 | | | |
| 10 | created_at | timestamp | | | No | CURRENT_TIMESTAMP | | | |

Check all With selected:

Table structure

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|--------------------|-----------|-----------------------------|------------|------|-------------------|----------|-----------------------------|--------|
| 1 | inventory_id | int(11) | | | No | None | | AUTO_INCREMENT | |
| 2 | product_id | int(11) | | | No | None | | | |
| 3 | quantity_available | int(11) | | | No | 0 | | | |
| 4 | quantity_reserved | int(11) | | | No | 0 | | | |
| 5 | reorder_level | int(11) | | | Yes | 10 | | | |
| 6 | last_updated_by | int(11) | | | Yes | NULL | | | |
| 7 | updated_at | timestamp | on update CURRENT_TIMESTAMP | | No | CURRENT_TIMESTAMP | | ON UPDATE CURRENT_TIMESTAMP | |

Check all With selected:

Add 1 column(s) after updated_at

Recent Favorites

Table structure Relation view

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|----|------------------------|---|-------------------|------------|------|-------------------|----------|----------------|--------|
| 1 | order_id | int(11) | | | No | None | | AUTO_INCREMENT | |
| 2 | customer_id | int(11) | | | No | None | | | |
| 3 | order_date | timestamp | | | No | CURRENT_TIMESTAMP | | | |
| 4 | total_amount | decimal(10,2) | | | No | None | | | |
| 5 | status | enum('Pending','Processing','Shipped','Deliver... | latin1_swedish_ci | | Yes | Pending | | | |
| 6 | shipping_address_line1 | varchar(255) | latin1_swedish_ci | | No | None | | | |
| 7 | shipping_address_line2 | varchar(255) | latin1_swedish_ci | | Yes | NULL | | | |
| 8 | shipping_city | varchar(100) | latin1_swedish_ci | | No | None | | | |
| 9 | shipping_state | varchar(50) | latin1_swedish_ci | | No | None | | | |
| 10 | shipping_zip | varchar(20) | latin1_swedish_ci | | No | None | | | |
| 11 | shipping_country | varchar(100) | latin1_swedish_ci | | Yes | USA | | | |
| 12 | notes | text | latin1_swedish_ci | | Yes | NULL | | | |
| 13 | fulfilled_by | int(11) | | | Yes | NULL | | | |
| 14 | fulfilled_at | timestamp | | | Yes | NULL | | | |

Recent Favorites

Table structure Relation view

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|---------------|---------------|-----------|------------|------|---------|----------|----------------|--------|
| 1 | order_item_id | int(11) | | | No | None | | AUTO_INCREMENT | |
| 2 | order_id | int(11) | | | No | None | | | |
| 3 | product_id | int(11) | | | No | None | | | |
| 4 | quantity | int(11) | | | No | None | | | |
| 5 | unit_price | decimal(10,2) | | | No | None | | | |
| 6 | subtotal | decimal(10,2) | | | No | None | | | |

Check all With selected: Primary Unique Index Spatial Fulltext

Recent Favorites

Browse Structure SQL Search Insert Export Import Operations Triggers

Table structure Relation view

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|------------|--------------|-------------------|-----------------------------|------|-------------------|----------|-----------------------------|--------|
| 1 | reset_id | int(11) | | | No | None | | AUTO_INCREMENT | |
| 2 | user_id | int(11) | | | No | None | | | |
| 3 | token | varchar(255) | latin1_swedish_ci | | No | None | | | |
| 4 | expires_at | timestamp | | on update CURRENT_TIMESTAMP | No | CURRENT_TIMESTAMP | | ON UPDATE CURRENT_TIMESTAMP | |
| 5 | used | tinyint(1) | | | Yes | 0 | | | |
| 6 | created_at | timestamp | | | No | CURRENT_TIMESTAMP | | | |

Check all With selected: Primary Unique Index Spatial Fulltext

Print Propose table structure

Recent Favorites

Table structure Relation view

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|----|--------------|---------------|-------------------|-----------------------------|------|-------------------|----------|-----------------------------|--------|
| 1 | product_id | int(11) | | | No | None | | AUTO_INCREMENT | |
| 2 | category_id | int(11) | | | No | None | | | |
| 3 | product_name | varchar(200) | latin1_swedish_ci | | No | None | | | |
| 4 | description | text | latin1_swedish_ci | | Yes | NULL | | | |
| 5 | price | decimal(10,2) | | | No | None | | | |
| 6 | sku | varchar(50) | latin1_swedish_ci | | No | None | | | |
| 7 | image_url | varchar(255) | latin1_swedish_ci | | Yes | NULL | | | |
| 8 | is_active | tinyint(1) | | | Yes | 1 | | | |
| 9 | created_by | int(11) | | | Yes | NULL | | | |
| 10 | created_at | timestamp | | | No | CURRENT_TIMESTAMP | | | |
| 11 | updated_at | timestamp | | on update CURRENT_TIMESTAMP | No | CURRENT_TIMESTAMP | | ON UPDATE CURRENT_TIMESTAMP | |

phpMyAdmin

Recent Favorites

information_schema
sahs2_Company
sahs2_dbassgn1
sahs2_Inventory
sahs2_starnavs_la
New
articles
article_comments
audit_log
categories

Browse Structure SQL Search Insert Export Import Operations Triggers

Table structure Relation view

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|-------------|-------------|-------------------|------------|------|-------------------|----------|----------------|--------|
| 1 | role_id | int(11) | | | No | None | | AUTO_INCREMENT | |
| 2 | role_name | varchar(50) | latin1_swedish_ci | | No | None | | | |
| 3 | description | text | latin1_swedish_ci | | Yes | NULL | | | |
| 4 | created_at | timestamp | | | No | CURRENT_TIMESTAMP | | | |

Check all With selected: Browse Primary Unique Index Spatial Fulltext Print Propose table structure

phpMyAdmin

Recent Favorites

information_schema
sahs2_Company
sahs2_dbassgn1
sahs2_Inventory
sahs2_starnavs_la
New
articles
article_comments
audit_log
categories

Browse Structure SQL Search Insert Export Import Operations Triggers

Table structure Relation view

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|-------------|-----------|-----------|------------|------|-------------------|----------|----------------|--------|
| 1 | cart_id | int(11) | | | No | None | | AUTO_INCREMENT | |
| 2 | customer_id | int(11) | | | No | None | | | |
| 3 | product_id | int(11) | | | No | None | | | |
| 4 | quantity | int(11) | | | No | 1 | | | |
| 5 | added_at | timestamp | | | No | CURRENT_TIMESTAMP | | | |

Check all With selected: Browse Primary Unique Index Spatial Fulltext

phpMyAdmin

Recent Favorites

information_schema
sahs2_Company
sahs2_dbassgn1
sahs2_Inventory
sahs2_starnavs_la
New
articles
article_comments
audit_log
categories
customers
employees
inventory
orders

Browse Structure SQL Search Insert Export Import Operations Triggers

Table structure Relation view

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|---------------|--------------|-------------------|-----------------------------|------|-------------------|----------|-----------------------------|--------|
| 1 | user_id | int(11) | | | No | None | | AUTO_INCREMENT | |
| 2 | username | varchar(50) | latin1_swedish_ci | | No | None | | | |
| 3 | email | varchar(100) | latin1_swedish_ci | | No | None | | | |
| 4 | password_hash | varchar(255) | latin1_swedish_ci | | No | None | | | |
| 5 | role_id | int(11) | | | No | None | | | |
| 6 | is_active | tinyint(1) | | | Yes | 1 | | | |
| 7 | last_login | timestamp | | | Yes | NULL | | | |
| 8 | created_at | timestamp | | | No | CURRENT_TIMESTAMP | | | |
| 9 | updated_at | timestamp | | on update CURRENT_TIMESTAMP | No | CURRENT_TIMESTAMP | | ON UPDATE CURRENT_TIMESTAMP | |

Check all With selected: Browse Primary Unique Index Spatial Fulltext

phpMyAdmin

Server: localhost:3306 > Database: sahs2_starnavs_la > Table: user_sessions

Recent Favorites

information_schema
sahs2_Company
sahs2_dbassgn1
sahs2_Inventory
sahs2_starnavs_la
New
articles
article_comments
audit_log
categories
customers

Browse Structure SQL Search Insert Export Import Operations Triggers

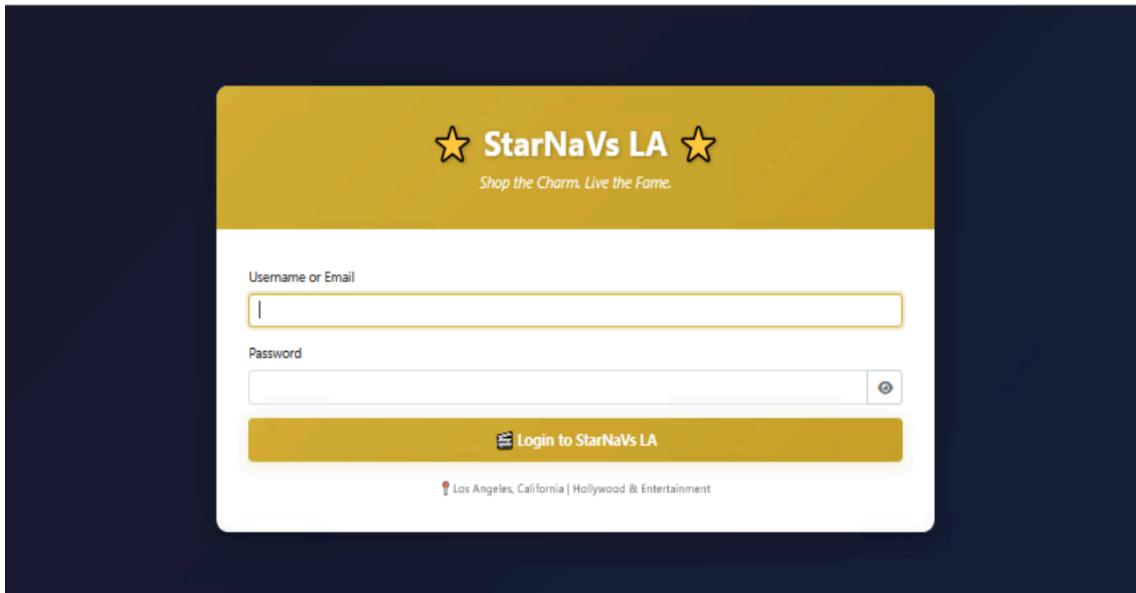
Table structure Relation view

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|---------------|--------------|-------------------|-----------------------------|------|-------------------|----------|-----------------------------|--------|
| 1 | session_id | varchar(128) | latin1_swedish_ci | | No | None | | | |
| 2 | user_id | int(11) | | | No | None | | | |
| 3 | ip_address | varchar(45) | latin1_swedish_ci | | Yes | NULL | | | |
| 4 | user_agent | varchar(255) | latin1_swedish_ci | | Yes | NULL | | | |
| 5 | last_activity | timestamp | | on update CURRENT_TIMESTAMP | No | CURRENT_TIMESTAMP | | ON UPDATE CURRENT_TIMESTAMP | |
| 6 | created_at | timestamp | | | No | CURRENT_TIMESTAMP | | | |

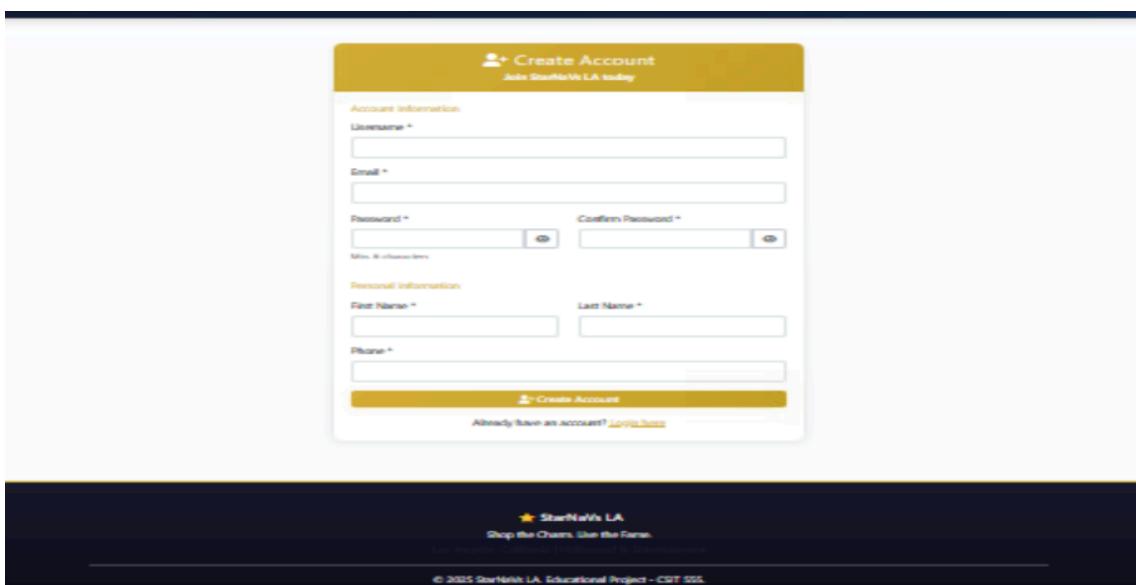
Check all With selected: Browse Primary Unique Index Spatial Fulltext

Website Implementation:

The website follows an MVC architecture with separate controllers for product management, HR tools, authentication, and cart operations. Prepared statements ensure secure database calls. Front-end components were built using Bootstrap for responsive layout. Employee dashboards provide tools for inventory updates, shipment status changes, and educational article publishing.



This is the “Sign In” Page



This is the Sign Up. After successful login, you can see the “OUR WEBSITE PAGE”

The screenshot shows a web browser window with the URL <https://cyan.csam.montclair.edu/~sahs2/starly/public/employee/inventory.php>. The page title is "Inventory Management". Key statistics are displayed in four boxes: "TOTAL PRODUCTS" (30), "TOTAL ITEMS IN STOCK" (1,444), "LOW STOCK ALERTS" (0), and "OUT OF STOCK" (0). Below this is a table titled "All Inventory" listing products like "Signed Hollywood Art Print" and "Collector Edition Film Strip Wall Art" with columns for Product, Category, SKU, Price, In Stock, Reserved, Reorder Level, Status, and Action.

| Product | Category | SKU | Price | In Stock | Reserved | Reorder Level | Status | Action |
|---------------------------------------|------------------|--------------|----------|----------|----------|---------------|--------|--------|
| Signed Hollywood Art Print | Limited Editions | LE-SHAP-001 | \$299.99 | 10 | 0 | 2 | Good | |
| Collector Edition Film Strip Wall Art | Limited Editions | LE-FSMAC-005 | \$179.99 | 12 | 0 | 3 | Good | |

This is “Inventory Management” section .

The screenshot shows a large table titled "All Inventory" with many rows of product data. The columns include Product, Category, SKU, Price, In Stock, Reserved, Reorder Level, Status, and Action. The table is scrollable, indicating it contains many more items than can fit on the screen at once.

| Product | Category | SKU | Price | In Stock | Reserved | Reorder Level | Status | Action |
|---|-------------------|--------------|----------|----------|----------|---------------|--------|--------|
| Signed Hollywood Art Print | Limited Editions | LE-SHAP-001 | \$299.99 | 10 | 0 | 2 | Good | |
| Collector Edition Film Strip Wall Art | Limited Editions | LE-FSMAC-005 | \$179.99 | 12 | 0 | 3 | Good | |
| City of Stars Hollywood Canvas Art | Home & Decor | HD-CDSB-005 | \$129.99 | 15 | 0 | 5 | Good | |
| Walk of fame Collector Box | Limited Editions | LE-AOTC-002 | \$199.99 | 15 | 0 | 3 | Good | |
| Red Carpet Premiere Jacket | Hollywood Fashion | HF-RCPT-001 | \$189.99 | 17 | 0 | 5 | Good | |
| Director Chair - Hollywood Studio Edition | Home & Decor | HD-DSCE-001 | \$149.99 | 19 | 0 | 5 | Good | |
| Hollywood Skyline Poster | Limited Editions | LE-HSP-006 | \$124.99 | 20 | 0 | 5 | Good | |
| Mini Film Camera Sculpture | Movie Collections | MC-AMC-001 | \$99.99 | 20 | 0 | 5 | Good | |
| Hollywood Mystery Merch Set | Limited Editions | LE-HMPS-001 | \$149.99 | 25 | 0 | 5 | Good | |
| Hollywood Award Miniatures | Movie Collections | MC-HAM-001 | \$64.99 | 30 | 0 | 8 | Good | |
| Walk of fame Fedora Hat | Hollywood Fashion | HF-WFH-001 | \$89.99 | 30 | 0 | 8 | Good | |
| Classic Film Poster | Movie Collections | MC-CFP-001 | \$89.99 | 35 | 0 | 8 | Good | |
| Studio Spotlight Table Lamp | Home & Decor | HD-SSLT-001 | \$89.99 | 35 | 0 | 8 | Good | |
| Iconic Hollywood Sunglasses | Hollywood Fashion | HF-CHS-001 | \$79.99 | 39 | 0 | 10 | Good | |
| <hr/> | | | | | | | | |
| Director Chair - Hollywood Studio Edition | Home & Decor | HD-DSCE-001 | \$149.99 | 10 | 0 | 5 | Good | |
| Hollywood Skyline Poster | Limited Editions | LE-HSP-006 | \$124.99 | 20 | 0 | 5 | Good | |
| Mini Film Camera Sculpture | Movie Collections | MC-AMC-001 | \$99.99 | 20 | 0 | 5 | Good | |
| Hollywood Mystery Merch Set | Limited Editions | LE-HMPS-001 | \$149.99 | 25 | 0 | 5 | Good | |
| Hollywood Award Miniatures | Movie Collections | MC-HAM-001 | \$64.99 | 30 | 0 | 8 | Good | |
| Walk of fame Fedora Hat | Hollywood Fashion | HF-WFH-001 | \$89.99 | 30 | 0 | 8 | Good | |
| Classic Film Poster | Movie Collections | MC-CFP-001 | \$89.99 | 35 | 0 | 8 | Good | |
| Studio Spotlight Table Lamp | Home & Decor | HD-SSLT-001 | \$89.99 | 35 | 0 | 8 | Good | |
| Iconic Hollywood Sunglasses | Hollywood Fashion | HF-CHS-001 | \$79.99 | 39 | 0 | 10 | Good | |
| Hollywood Star Charm Bracelet | Accessories | AC-HSCB-001 | \$79.99 | 40 | 0 | 10 | Good | |
| Vintage Hollywood Vinyl Record | Movie Collections | MC-VHR-001 | \$44.99 | 40 | 0 | 10 | Good | |
| Film Reel Design Wall Clock | Home & Decor | HD-FRWC-001 | \$54.99 | 45 | 0 | 10 | Good | |
| Studio Script Booklet | Movie Collections | MC-SSB-001 | \$34.99 | 45 | 0 | 10 | Good | |
| Iconic Actor Figurine with Studio Base | Movie Collections | MC-IASB-001 | \$74.99 | 50 | 0 | 5 | Good | |
| Mini Oscar Replica | Limited Editions | LE-MOR-001 | \$29.99 | 50 | 0 | 10 | Good | |
| <hr/> | | | | | | | | |

| | | | | | | | | |
|--|-------------------|-------------|---------|-----|---|----|--|--|
| Iconic Hollywood Sunglasses | Hollywood Fashion | HF-CDS-001 | \$79.99 | 39 | 0 | 10 | | |
| Hollywood Star Charm Bracelet | Accessories | AC-HSCB-001 | \$79.99 | 40 | 0 | 10 | | |
| Vintage Hollywood Vinyl Record | Movie Collections | MC-HVNR-001 | \$44.99 | 40 | 0 | 10 | | |
| Film Reel Design Wall Clock | Home & Decor | HD-FRDC-001 | \$34.99 | 41 | 0 | 10 | | |
| Studio Script Booklet | Movie Collections | MC-SBC-001 | \$14.99 | 45 | 0 | 10 | | |
| Iconic Actor Figurine with Studio Base | Movie Collections | MC-AFB-001 | \$14.99 | 50 | 0 | 5 | | |
| Mini Oscar Replica | Limited Editions | LE-HOSR-001 | \$89.99 | 50 | 0 | 10 | | |
| Studio Crew Headie | Hollywood Fashion | HF-SDCH-001 | \$39.99 | 50 | 0 | 10 | | |
| Hollywood Enamel Pin Set | Accessories | AC-HEPS-001 | \$14.99 | 60 | 0 | 12 | | |
| Studio Cap | Hollywood Fashion | HF-SFC-001 | \$29.99 | 60 | 0 | 12 | | |
| Awards Night Canvas Tote Bag | Accessories | AC-ANTB-001 | \$14.99 | 70 | 0 | 15 | | |
| Star Print Designer Tee | Hollywood Fashion | HF-SDPT-001 | \$34.99 | 74 | 0 | 15 | | |
| Hollywood Skyline Scented Candle | Home & Decor | HD-HSC-001 | \$39.99 | 78 | 0 | 15 | | |
| Premiere Edition Phone Case | Accessories | AC-PDPC-001 | \$29.99 | 90 | 0 | 20 | | |
| Clockwise Coaster Set (4 Pack) | Home & Decor | HD-CDS-001 | \$14.99 | 100 | 0 | 20 | | |
| Movie Red Carpet | Accessories | AC-MRC-001 | \$16.99 | 120 | 0 | 25 | | |
| Hollywood Star-Inspired Keychain | Accessories | AC-HSKC-001 | \$19.99 | 150 | 0 | 25 | | |

This is “Manager” section .

| Order # | Customer | Date | Total | Ship To | Status | Actions |
|---------|----------|----------------------|----------|-------------------------|---------|---------|
| #6 | John Doe | Dec 11, 2025 9:31 AM | \$149.99 | Los Angeles, California | Pending | |

This is “order fulfillment” section .

| Title | Category | Author | Published Date | Status | Comments | Actions |
|--|-----------|------------|----------------|-----------|----------|---------|
| The Birth of Hollywood: How Los Angeles Became the Entertainment Capital | History | Shweta Sah | Dec 10, 2025 | Published | 2 | |
| Hollywood Walk of Fame: Stars on Sidewalks | Landmarks | Shweta Sah | Dec 10, 2025 | Published | 1 | |

This is "editor view" section .

https://cyan.csam.montclair.edu/~sahs2/starly/public/customer/shop.php

80% Cart john_doe

Shop Hollywood Merchandise

Discover exclusive Hollywood-inspired fashion, decor, and collectibles

Search for Hollywood products...

All Products (30)

Accessories (6)

Hollywood Fashion (6)

Home & Decor (6)

Shop Hollywood Merc...

As a user what we can see and shop.

StarNavs LA Home Shop Hollywood 101 My Orders

Shop / Hollywood Fashion / Red Carpet Premiere Jacket

Red Carpet Premiere Jacket

\$189.99

SKU: HF-RCP-001

Availability: In Stock (17 available)

Description: Luxurious velvet jacket inspired by Hollywood premieres, perfect for those special nights out.

Quantity: 1 Add to Cart Continue Shopping

Product Features:

- Authentic Hollywood-inspired design
- High-quality velvet fabric
- Perfect for Hollywood enthusiasts and collectors
- Made from Los Angeles, California
- Makes a great gift for film lovers

Red Carpet Premiere J...

After clicking on the product what we can see is.



Red Carpet Premiere Jacket

\$189.99

SKU: HF-RCPI-001

Availability: ✓ In Stock (17 available)

Description

Luxurious velvet jacket inspired by Hollywood premieres, perfect for those special nights out.

Quantity

- +

Add to Cart

[← Continue Shopping](#)

tures

We can add the quantity here



Luxurious velvet jacket inspired by Hollywood premieres, perfect for those special nights out.

Quantity

- +

Add to Cart

[← Continue Shopping](#)

Product Features

- Authentic Hollywood-inspired design
- High-quality materials and craftsmanship
- Perfect for Hollywood enthusiasts and collectors
- Ships from Los Angeles, California
- Makes a great gift for film lovers

These are the features of the product.

The screenshot shows a shopping cart interface. On the left, there's a section titled "Cart Items (1)" showing a single item: "Red Carpet Premiere jacket" by Hollywood Rollers, priced at \$180.00, quantity 4, and a total of \$720.00. Below this are buttons for "Update Cart", "Clear Cart", and "Continue Shopping". On the right, an "Order Summary" table provides a breakdown of the costs:

| Subtotal | \$720.00 |
|---------------------|----------|
| Tax (9.5%) | \$72.00 |
| Shipping | \$0.00 |
| No shipping applied | |
| Total | \$832.16 |

Below the summary is a large yellow "Proceed to Checkout" button. At the bottom right of the page, there's a footer with the text "StarNets LA", "Skip the Chairs. Use the Force.", "Los Angeles California | Education & Entertainment", and "© 2003 StarNets LA Educational Project - CST 55A".

This is "shopping cart view".

The screenshot shows a secure checkout process. The top section, "Secure Checkout", contains a summary of the order items: "Red Carpet Premiere jacket" (Qty 4 = \$180.00) with a total of \$720.00. The "Order Summary" table is identical to the one in the shopping cart view.

The main form area is divided into sections: "Shipping Address", "Order Notes (Optional)", and "Payment Information".

- Shipping Address:** Fields include "Address Line 1" (123 Hollywood Blvd), "Address Line 2" (empty), "City" (Los Angeles), "State" (California), "ZIP Code" (90020), and "Country" (USA).
- Order Notes (Optional):** A text area containing "Special instructions or delivery notes...".
- Payment Information:** A section showing the payment method selected: "Payment Method: Invoice (Net 30)".

Here we are ready to place the order.

Analysis of Database Issues

The database constructed for StarNaVs LA aims to be in 2nd Normal Form and 3rd Normal Form, which will eliminate redundancy along with providing data integrity.

Normal Form:

2NF ensures that all non-key attributes are functionally dependent solely on the primary key. That is, there are no partial dependencies. For instance, in a "Orders" relation, a set of attributes like "order_status" would be dependent solely on "order_id."

To acquire 3NF, transitive dependencies are removed. For example, since Product Category data is maintained in a separate Categories table, it prevents redundant data of product categories being maintained in the Products table.

Redundancy and Reachability

Redundant data has been reduced using foreign keys. This means that every product, order, and category has a reference ID, making it easy to maintain. This ensures that scaling takes place with ease.

Foreign key constraints make it possible to maintain referential integrity in a database, for example, between Order Items and the Products table.

In conclusion, the database is normalized for redundancy elimination and for data to be accessible as well as maintainable. Every table in the database has a logical design structure that follows 2NF and 3NF.

Team Work Plan & Code Responsibilities

Nirav: Shopping cart flow, customer dashboard, checkout functionality, and documentations.

Vivek: Designing database schema, SQL scripts, normalization, management dashboard utility.

Amisha: Design of the front-end, product list pages, design, and responsiveness.

Swetha: Authentication mechanism, role-based access control, HR module, updating inventory. All of them were involved in writing PHP scripts, integration with databases, and also the UI testing.

Security Discussion:

A compulsory login landing page with no artwork, secure password storage with bcrypt, secure SQL injection protection with the prepared statements, authentication with sessions, and role-based access restriction. There are no stored credit card numbers. Only receipt-style confirmations are allowed per assignment rules.

Conclusion & Future Enhancements

With additional time, other functionalities like analytics, product recommendations using AI, enhanced media galleries, and a comprehensive CMS for Hollywood titles would also be incorporated. The project effectively provides an e-commerce solution for all requirements in a course.

Appendix Code Repository

INSERT ALL THE CODE HERE

Database.php

```
<?php

class Database {
    private $host = 'localhost';
    private $db_name = 'sahs2_starnavs_la';
    private $username = 'sahs2_navs';
    private $password = 'Starnavs123@';
    private $conn;

    public function getConnection() {
        $this->conn = null;

        try {
            $this->conn = new PDO(
                "mysql:host=" . $this->host . ";dbname=" . $this->db_name,
                $this->username,
                $this->password,
                array(
                    PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION,
                    PDO::ATTR_DEFAULT_FETCH_MODE => PDO::FETCH_ASSOC,
                    PDO::ATTR_EMULATE_PREPARES => false,
                    PDO::MYSQL_ATTR_INIT_COMMAND => "SET NAMES utf8mb4"
                )
            );
        } catch(PDOException $e) {
            error_log("Connection Error: " . $e->getMessage());
            die("Database connection failed. Please contact support.");
        }

        return $this->conn;
    }

    public function closeConnection() {
        $this->conn = null;
    }
}

?>
```

Config.php

```

<?php

error_reporting(E_ALL);
ini_set('display_errors', 1);

date_default_timezone_set('America/Los_Angeles');

define('APP_NAME', 'StarNaVs LA');
define('APP_TAGLINE', 'Shop the Charm. Live the Fame.');
define('APP_VERSION', '1.0.0');

define('BASE_URL', 'https://cyan.csam.montclair.edu/~sahs2/starly/');
define('ASSETS_URL', BASE_URL . 'assets/');
define('UPLOAD_URL', BASE_URL . 'uploads/');

define('ROOT_PATH', dirname(dirname(__FILE__)) . '/');
define('CONFIG_PATH', ROOT_PATH . 'config/');
define('MODEL_PATH', ROOT_PATH . 'models/');
define('VIEW_PATH', ROOT_PATH . 'views/');
define('CONTROLLER_PATH', ROOT_PATH . 'controllers/');
define('UPLOAD_PATH', ROOT_PATH . 'uploads/');

define('SESSION_LIFETIME', 3600);
define('SESSION_NAME', 'STARNAVS_SESSION');

define('BCRYPT_COST', 10);
define('PASSWORD_MIN_LENGTH', 8);
define('MAX_LOGIN_ATTEMPTS', 5);
define('LOGIN_LOCKOUT_TIME', 900);

define('PRODUCTS_PER_PAGE', 12);
define('ORDERS_PER_PAGE', 10);
define('ARTICLES_PER_PAGE', 6);

define('MAX_FILE_SIZE', 5242880); // 5MB in bytes
define('ALLOWED_IMAGE_TYPES', ['image/jpeg', 'image/png', 'image/gif', 'image/webp']);

define('TAX_RATE', 0.095); // 9.5% CA tax
define('FREE SHIPPING_THRESHOLD', 100.00);
define('STANDARD_SHIPPING_COST', 9.99);

define('SMTP_HOST', 'smtp.example.com');
define('SMTP_PORT', 587);
define('SMTP_USERNAME', 'noreply@starnavs.la');

```

```

define('SMTP_PASSWORD', '');
define('FROM_EMAIL', 'noreply@starnavs.la');
define('FROM_NAME', 'StarNaVs LA');

define('ROLE_DB_ADMIN', 1);
define('ROLE_BUSINESS_MANAGER', 2);
define('ROLE_INVENTORY_MANAGER', 3);
define('ROLE_ORDER_FULFILLMENT', 4);
define('ROLE_CONTENT_EDITOR', 5);
define('ROLE_CUSTOMER', 6);

define('ORDER_STATUS_PENDING', 'Pending');
define('ORDER_STATUS_PROCESSING', 'Processing');
define('ORDER_STATUS_SHIPPED', 'Shipped');
define('ORDER_STATUS_DELIVERED', 'Delivered');
define('ORDER_STATUS_CANCELLED', 'Cancelled');

require_once CONFIG_PATH . 'database.php';

spl_autoload_register(function($class) {
    $model_file = MODEL_PATH . $class . '.php';
    if (file_exists($model_file)) {
        require_once $model_file;
        return;
    }

    $controller_file = CONTROLLER_PATH . $class . '.php';
    if (file_exists($controller_file)) {
        require_once $controller_file;
        return;
    }
});

function h($string) {
    if ($string === null) {
        return '';
    }
    return htmlspecialchars($string, ENT_QUOTES, 'UTF-8');
}

function redirect($url) {
    header("Location: " . BASE_URL . $url);
    exit();
}

```

```

function isLoggedIn() {
    return isset($_SESSION['user_id']) && !empty($_SESSION['user_id']);
}

function getCurrentUserId() {
    return isset($_SESSION['user_id']) ? $_SESSION['user_id'] : null;
}

function getCurrentUserRole() {
    return isset($_SESSION['role_id']) ? $_SESSION['role_id'] : null;
}

function hasRole($role_id) {
    return getCurrentUserRole() === $role_id;
}

function isCustomer() {
    return hasRole(ROLE_CUSTOMER);
}

function isEmployee() {
    $role = getCurrentUserRole();
    return in_array($role, [
        ROLE_INVENTORY_MANAGER,
        ROLE_ORDER_FULFILLMENT,
        ROLE_CONTENT_EDITOR
    ]);
}

function isManager() {
    $role = getCurrentUserRole();
    return in_array($role, [ROLE_DB_ADMIN, ROLE_BUSINESS_MANAGER]);
}

function formatCurrency($amount) {
    if ($amount === null) {
        return '$0.00';
    }
    return '$' . number_format($amount, 2);
}

function formatDate($date, $format = 'M d, Y') {
    return date($format, strtotime($date));
}

```

```

}

function formatDate($datetime, $format = 'M d, Y g:i A') {
    return date($format, strtotime($datetime));
}

function generateCSRFToken() {
    if (!isset($_SESSION['csrf_token'])) {
        $_SESSION['csrf_token'] = bin2hex(random_bytes(32));
    }
    return $_SESSION['csrf_token'];
}

function verifyCSRFToken($token) {
    return isset($_SESSION['csrf_token']) && hash_equals($_SESSION['csrf_token'], $token);
}

function setFlashMessage($type, $message) {
    $_SESSION['flash_message'] = [
        'type' => $type,
        'message' => $message
    ];
}

function getFlashMessage() {
    if (isset($_SESSION['flash_message'])) {
        $message = $_SESSION['flash_message'];
        unset($_SESSION['flash_message']);
        return $message;
    }
    return null;
}

function logAudit($action, $table_name = null, $record_id = null, $old_value = null, $new_value = null) {
    try {
        $database = new Database();
        $db = $database->getConnection();

        $query = "INSERT INTO audit_log (user_id, action, table_name, record_id, old_value, new_value, ip_address, user_agent)
                  VALUES (:user_id, :action, :table_name, :record_id, :old_value, :new_value, :ip_address,
                  :user_agent)";

        $stmt = $db->prepare($query);
    }
}

```

```

$stmt->execute([
    ':user_id' => getCurrentUserId(),
    ':action' => $action,
    ':table_name' => $table_name,
    ':record_id' => $record_id,
    ':old_value' => $old_value,
    ':new_value' => $new_value,
    ':ip_address' => $_SERVER['REMOTE_ADDR'] ?? null,
    ':user_agent' => $_SERVER['HTTP_USER_AGENT'] ?? null
]);
} catch (Exception $e) {
    error_log("Audit log error: " . $e->getMessage());
}
}

?>

```

Index.php

```

<?php

require_once 'config/config.php';
require_once 'includes/session.php';

if (isLoggedIn()) {
    redirect('public/dashboard.php');
}

redirect('public/home.php');
?>

```

Articles.php

```

<?php

class Article {

    private $conn;
    private $table = 'articles';

    public function __construct($db) {
        $this->conn = $db;
    }

    public function getPublished($category = null, $limit = null, $offset = 0) {
        $query = "SELECT a.*, e.first_name, e.last_name,
            (SELECT COUNT(*) FROM article_comments WHERE article_id = a.article_id) as
        comment_count
    }
}
```

```

        FROM " . $this->table . " a
        JOIN employees e ON a.author_id = e.employee_id
        WHERE a.is_published = 1";

if ($category) {
$query .= " AND a.category = :category";
}

$query .= " ORDER BY a.published_at DESC";

if ($limit) {
$query .= " LIMIT :limit OFFSET :offset";
}

$stmt = $this->conn->prepare($query);

if ($category) {
$stmt->bindParam(':category', $category, PDO::PARAM_STR);
}

if ($limit) {
$stmt->bindParam(':limit', $limit, PDO::PARAM_INT);
$stmt->bindParam(':offset', $offset, PDO::PARAM_INT);
}

$stmt->execute();
return $stmt->fetchAll();
}

public function getAll($limit = null, $offset = 0) {
$query = "SELECT a.*, e.first_name, e.last_name,
(SELECT COUNT(*) FROM article_comments WHERE article_id = a.article_id) as
comment_count
        FROM " . $this->table . " a
        LEFT JOIN employees e ON a.author_id = e.employee_id
        ORDER BY a.created_at DESC";

if ($limit) {
$query .= " LIMIT :limit OFFSET :offset";
}

$stmt = $this->conn->prepare($query);

if ($limit) {

```

```

$stmt->bindParam(':limit', $limit, PDO::PARAM_INT);
$stmt->bindParam(':offset', $offset, PDO::PARAM_INT);
}

$stmt->execute();
return $stmt->fetchAll();
}

public function getById($article_id) {
$query = "SELECT a.*, e.first_name, e.last_name
    FROM " . $this->table . " a
    JOIN employees e ON a.author_id = e.employee_id
    WHERE a.article_id = :article_id";

$stmt = $this->conn->prepare($query);
$stmt->bindParam(':article_id', $article_id, PDO::PARAM_INT);
$stmt->execute();

return $stmt->fetch();
}

public function getBySlug($slug) {
$query = "SELECT a.*, e.first_name, e.last_name
    FROM " . $this->table . " a
    JOIN employees e ON a.author_id = e.employee_id
    WHERE a.slug = :slug AND a.is_published = 1";

$stmt = $this->conn->prepare($query);
$stmt->bindParam(':slug', $slug, PDO::PARAM_STR);
$stmt->execute();

return $stmt->fetch();
}

public function incrementViews($article_id) {
$query = "UPDATE " . $this->table . " SET view_count = view_count + 1 WHERE article_id =
'article_id';

$stmt = $this->conn->prepare($query);
$stmt->bindParam(':article_id', $article_id, PDO::PARAM_INT);

return $stmt->execute();
}

```

```

public function getComments($article_id) {
    $query = "SELECT ac.*, c.first_name, c.last_name
        FROM article_comments ac
        JOIN customers c ON ac.customer_id = c.customer_id
        WHERE ac.article_id = :article_id AND ac.is_approved = 1
        ORDER BY ac.created_at DESC";

    $stmt = $this->conn->prepare($query);
    $stmt->bindParam(':article_id', $article_id, PDO::PARAM_INT);
    $stmt->execute();

    return $stmt->fetchAll();
}

public function addComment($article_id, $customer_id, $comment_text) {
    try {
        $query = "INSERT INTO article_comments (article_id, customer_id, comment_text)
            VALUES (:article_id, :customer_id, :comment_text)";

        $stmt = $this->conn->prepare($query);
        $stmt->bindParam(':article_id', $article_id, PDO::PARAM_INT);
        $stmt->bindParam(':customer_id', $customer_id, PDO::PARAM_INT);
        $stmt->bindParam(':comment_text', $comment_text, PDO::PARAM_STR);

        return $stmt->execute();
    } catch (Exception $e) {
        error_log("Add comment error: " . $e->getMessage());
        return false;
    }
}

public function getPublishedCount($category = null) {
    $query = "SELECT COUNT(*) as count FROM " . $this->table . " WHERE is_published = 1";

    if ($category) {
        $query .= " AND category = :category";
    }

    $stmt = $this->conn->prepare($query);

    if ($category) {
        $stmt->bindParam(':category', $category, PDO::PARAM_STR);
    }
}

```

```

$stmt->execute();
$result = $stmt->fetch();

return $result['count'];
}

public function getCategories() {
$query = "SELECT DISTINCT category, COUNT(*) as count
    FROM " . $this->table . "
    WHERE is_published = 1 AND category IS NOT NULL
    GROUP BY category
    ORDER BY category";

$stmt = $this->conn->prepare($query);
$stmt->execute();

return $stmt->fetchAll();
}

public function create($data) {
$query = "INSERT INTO " . $this->table . "
    (title, slug, content, summary, author_id, category, featured_image, is_published)
    VALUES (:title, :slug, :content, :summary, :author_id, :category, :featured_image,
:is_published)";

$stmt = $this->conn->prepare($query);

$stmt->bindParam(':title', $data['title'], PDO::PARAM_STR);
$stmt->bindParam(':slug', $data['slug'], PDO::PARAM_STR);
$stmt->bindParam(':content', $data['content'], PDO::PARAM_STR);
$stmt->bindParam(':summary', $data['summary'], PDO::PARAM_STR);
$stmt->bindParam(':author_id', $data['author_id'], PDO::PARAM_INT);
$stmt->bindParam(':category', $data['category'], PDO::PARAM_STR);
$stmt->bindParam(':featured_image', $data['featured_image'], PDO::PARAM_STR);
$stmt->bindParam(':is_published', $data['is_published'], PDO::PARAM_INT);

if ($stmt->execute()) {
return $this->conn->lastInsertId();
}

return false;
}

?>

```

Cart.php

```
<?php

class Cart {
    private $conn;
    private $table = 'shopping_cart';

    public function __construct($db) {
        $this->conn = $db;
    }

    public function getCartItems($customer_id) {
        $query = "SELECT sc.*, p.product_name, p.price, p.image_url, p.sku,
                  i.quantity_available, c.category_name,
                  (sc.quantity * p.price) as item_total
            FROM " . $this->table . " sc
            JOIN products p ON sc.product_id = p.product_id
            JOIN categories c ON p.category_id = c.category_id
            LEFT JOIN inventory i ON p.product_id = i.product_id
            WHERE sc.customer_id = :customer_id
            ORDER BY sc.added_at DESC";

        $stmt = $this->conn->prepare($query);
        $stmt->bindParam(':customer_id', $customer_id, PDO::PARAM_INT);
        $stmt->execute();

        return $stmt->fetchAll();
    }

    public function addItem($customer_id, $product_id, $quantity = 1) {
        $check_query = "SELECT cart_id, quantity FROM " . $this->table . "
                      WHERE customer_id = :customer_id AND product_id = :product_id";

        $check_stmt = $this->conn->prepare($check_query);
        $check_stmt->bindParam(':customer_id', $customer_id, PDO::PARAM_INT);
        $check_stmt->bindParam(':product_id', $product_id, PDO::PARAM_INT);
        $check_stmt->execute();

        if ($existing = $check_stmt->fetch()) {
            $new_quantity = $existing['quantity'] + $quantity;
            return $this->updateQuantity($customer_id, $product_id, $new_quantity);
        } else {
            $query = "INSERT INTO " . $this->table . "
```

```

(customer_id, product_id, quantity)
VALUES (:customer_id, :product_id, :quantity)";

$stmt = $this->conn->prepare($query);
$stmt->bindParam(':customer_id', $customer_id, PDO::PARAM_INT);
$stmt->bindParam(':product_id', $product_id, PDO::PARAM_INT);
$stmt->bindParam(':quantity', $quantity, PDO::PARAM_INT);

return $stmt->execute();
}

}

public function updateQuantity($customer_id, $product_id, $quantity) {
if ($quantity <= 0) {
return $this->removeItem($customer_id, $product_id);
}

$query = "UPDATE " . $this->table . "
SET quantity = :quantity
WHERE customer_id = :customer_id AND product_id = :product_id";

$stmt = $this->conn->prepare($query);
$stmt->bindParam(':customer_id', $customer_id, PDO::PARAM_INT);
$stmt->bindParam(':product_id', $product_id, PDO::PARAM_INT);
$stmt->bindParam(':quantity', $quantity, PDO::PARAM_INT);

return $stmt->execute();
}

public function removeItem($customer_id, $product_id) {
$query = "DELETE FROM " . $this->table . "
WHERE customer_id = :customer_id AND product_id = :product_id";

$stmt = $this->conn->prepare($query);
$stmt->bindParam(':customer_id', $customer_id, PDO::PARAM_INT);
$stmt->bindParam(':product_id', $product_id, PDO::PARAM_INT);

return $stmt->execute();
}

public function clearCart($customer_id) {
$query = "DELETE FROM " . $this->table . " WHERE customer_id = :customer_id";

$stmt = $this->conn->prepare($query);

```

```

$stmt->bindParam(':customer_id', $customer_id, PDO::PARAM_INT);

return $stmt->execute();
}

public function getCartTotal($customer_id) {
$query = "SELECT SUM(sc.quantity * p.price) as total
    FROM " . $this->table . " sc
    JOIN products p ON sc.product_id = p.product_id
    WHERE sc.customer_id = :customer_id";

$stmt = $this->conn->prepare($query);
$stmt->bindParam(':customer_id', $customer_id, PDO::PARAM_INT);
$stmt->execute();

$result = $stmt->fetch();
return $result['total'] ?? 0;
}

public function getCartCount($customer_id) {
$query = "SELECT SUM(quantity) as count FROM " . $this->table . "
    WHERE customer_id = :customer_id";

$stmt = $this->conn->prepare($query);
$stmt->bindParam(':customer_id', $customer_id, PDO::PARAM_INT);
$stmt->execute();

$result = $stmt->fetch();
return $result['count'] ?? 0;
}

public function validateCart($customer_id) {
$query = "SELECT sc.product_id, sc.quantity, p.product_name, i.quantity_available
    FROM " . $this->table . " sc
    JOIN products p ON sc.product_id = p.product_id
    LEFT JOIN inventory i ON p.product_id = i.product_id
    WHERE sc.customer_id = :customer_id
    AND (i.quantity_available IS NULL OR sc.quantity > i.quantity_available)";

$stmt = $this->conn->prepare($query);
$stmt->bindParam(':customer_id', $customer_id, PDO::PARAM_INT);
$stmt->execute();

return $stmt->fetchAll();
}

```

```
    }
}
?>
```

Category.php

```
<?php

class Category {
    private $conn;
    private $table = 'categories';

    public function __construct($db) {
        $this->conn = $db;
    }

    public function getAll() {
        $query = "SELECT c.*, COUNT(p.product_id) as product_count
                  FROM " . $this->table . " c
                  LEFT JOIN products p ON c.category_id = p.category_id AND p.is_active = 1
                  WHERE c.is_active = 1
                  GROUP BY c.category_id
                  ORDER BY c.category_name";

        $stmt = $this->conn->prepare($query);
        $stmt->execute();

        return $stmt->fetchAll();
    }

    public function getById($category_id) {
        $query = "SELECT * FROM " . $this->table . " WHERE category_id = :category_id";

        $stmt = $this->conn->prepare($query);
        $stmt->bindParam(':category_id', $category_id, PDO::PARAM_INT);
        $stmt->execute();

        return $stmt->fetch();
    }

    public function create($data) {
        $query = "INSERT INTO " . $this->table . "
                  (category_name, description, image_url)
                  VALUES (:category_name, :description, :image_url)";

        $stmt = $this->conn->prepare($query);
        $stmt->execute($data);
    }
}
```

```

$stmt = $this->conn->prepare($query);

$stmt->bindParam(':category_name', $data['category_name'], PDO::PARAM_STR);
$stmt->bindParam(':description', $data['description'], PDO::PARAM_STR);
$stmt->bindParam(':image_url', $data['image_url'], PDO::PARAM_STR);

if ($stmt->execute()) {
    return $this->conn->lastInsertId();
}

return false;
}

public function update($category_id, $data) {
    $query = "UPDATE " . $this->table . "
        SET category_name = :category_name,
            description = :description,
            image_url = :image_url,
            is_active = :is_active
        WHERE category_id = :category_id";

    $stmt = $this->conn->prepare($query);

    $stmt->bindParam(':category_id', $category_id, PDO::PARAM_INT);
    $stmt->bindParam(':category_name', $data['category_name'], PDO::PARAM_STR);
    $stmt->bindParam(':description', $data['description'], PDO::PARAM_STR);
    $stmt->bindParam(':image_url', $data['image_url'], PDO::PARAM_STR);
    $stmt->bindParam(':is_active', $data['is_active'] ?? 1, PDO::PARAM_INT);

    return $stmt->execute();
}

public function delete($category_id) {
    $query = "UPDATE " . $this->table . "
        SET is_active = 0
        WHERE category_id = :category_id";

    $stmt = $this->conn->prepare($query);
    $stmt->bindParam(':category_id', $category_id, PDO::PARAM_INT);

    return $stmt->execute();
}

public function nameExists($category_name, $exclude_category_id = null) {
    $query = "SELECT category_id FROM " . $this->table . "
        WHERE category_name = :category_name";

```

```

if ($exclude_category_id) {
$query .= " AND category_id != :exclude_category_id";
}

$stmt = $this->conn->prepare($query);
$stmt->bindParam(':category_name', $category_name, PDO::PARAM_STR);

if ($exclude_category_id) {
$stmt->bindParam(':exclude_category_id', $exclude_category_id, PDO::PARAM_INT);
}

$stmt->execute();
return $stmt->rowCount() > 0;
}
?>

```

Customer.php

```

<?php

class Customer {
    private $conn;
    private $table = 'customers';

    public function __construct($db) {
        $this->conn = $db;
    }

    public function getUserId($user_id) {
        $query = "SELECT c.*, u.username, u.email
                  FROM " . $this->table . " c
                  JOIN users u ON c.user_id = u.user_id
                  WHERE c.user_id = :user_id";

        $stmt = $this->conn->prepare($query);
        $stmt->bindParam(':user_id', $user_id, PDO::PARAM_INT);
        $stmt->execute();

        return $stmt->fetch();
    }

    public function getId($customer_id) {
        $query = "SELECT c.*, u.username, u.email
                  FROM " . $this->table . " c
                  WHERE c.id = :customer_id";
    }
}
```

```

JOIN users u ON c.user_id = u.user_id
WHERE c.customer_id = :customer_id";

$stmt = $this->conn->prepare($query);
$stmt->bindParam(':customer_id', $customer_id, PDO::PARAM_INT);
$stmt->execute();

return $stmt->fetch();
}

public function create($user_id, $first_name, $last_name, $phone = null, $address_data = []) {
$query = "INSERT INTO " . $this->table . "
(user_id, first_name, last_name, phone, address_line1, address_line2, city, state, zip_code,
country)
VALUES (:user_id, :first_name, :last_name, :phone, :address_line1, :address_line2, :city, :state,
:zip_code, :country)";

$stmt = $this->conn->prepare($query);

$stmt->bindParam(':user_id', $user_id, PDO::PARAM_INT);
$stmt->bindParam(':first_name', $first_name, PDO::PARAM_STR);
$stmt->bindParam(':last_name', $last_name, PDO::PARAM_STR);
$stmt->bindParam(':phone', $phone, PDO::PARAM_STR);
$stmt->bindParam(':address_line1', $address_data['address_line1'] ?? null, PDO::PARAM_STR);
$stmt->bindParam(':address_line2', $address_data['address_line2'] ?? null, PDO::PARAM_STR);
$stmt->bindParam(':city', $address_data['city'] ?? null, PDO::PARAM_STR);
$stmt->bindParam(':state', $address_data['state'] ?? null, PDO::PARAM_STR);
$stmt->bindParam(':zip_code', $address_data['zip_code'] ?? null, PDO::PARAM_STR);
$stmt->bindParam(':country', $address_data['country'] ?? 'USA', PDO::PARAM_STR);

if ($stmt->execute()) {
return $this->conn->lastInsertId();
}

return false;
}

public function update($customer_id, $data) {
$query = "UPDATE " . $this->table . "
SET first_name = :first_name,
last_name = :last_name,
phone = :phone,
address_line1 = :address_line1,
address_line2 = :address_line2,

```

```

    city = :city,
    state = :state,
    zip_code = :zip_code,
    country = :country
    WHERE customer_id = :customer_id";

$stmt = $this->conn->prepare($query);

$stmt->bindParam(':customer_id', $customer_id, PDO::PARAM_INT);
$stmt->bindParam(':first_name', $data['first_name'], PDO::PARAM_STR);
$stmt->bindParam(':last_name', $data['last_name'], PDO::PARAM_STR);
$stmt->bindParam(':phone', $data['phone'], PDO::PARAM_STR);
$stmt->bindParam(':address_line1', $data['address_line1'], PDO::PARAM_STR);
$stmt->bindParam(':address_line2', $data['address_line2'], PDO::PARAM_STR);
$stmt->bindParam(':city', $data['city'], PDO::PARAM_STR);
$stmt->bindParam(':state', $data['state'], PDO::PARAM_STR);
$stmt->bindParam(':zip_code', $data['zip_code'], PDO::PARAM_STR);
$stmt->bindParam(':country', $data['country'], PDO::PARAM_STR);

return $stmt->execute();
}

public function getAll($limit = null, $offset = 0) {
$query = "SELECT c.*, u.username, u.email, u.is_active
    FROM " . $this->table . " c
    JOIN users u ON c.user_id = u.user_id
    ORDER BY c.created_at DESC";

if ($limit) {
$query .= " LIMIT :limit OFFSET :offset";
}

$stmt = $this->conn->prepare($query);

if ($limit) {
$stmt->bindParam(':limit', $limit, PDO::PARAM_INT);
$stmt->bindParam(':offset', $offset, PDO::PARAM_INT);
}

$stmt->execute();
return $stmt->fetchAll();
}

public function getCount() {

```

```

$query = "SELECT COUNT(*) as count FROM " . $this->table;

$stmt = $this->conn->prepare($query);
$stmt->execute();
$result = $stmt->fetch();

return $result['count'];
}

public function search($search_term) {
$query = "SELECT c.*, u.username, u.email
FROM " . $this->table . " c
JOIN users u ON c.user_id = u.user_id
WHERE c.first_name LIKE :search
OR c.last_name LIKE :search
OR u.email LIKE :search
OR u.username LIKE :search
ORDER BY c.last_name, c.first_name"; 

$stmt = $this->conn->prepare($query);
$search_param = '%' . $search_term . '%';
$stmt->bindParam(':search', $search_param, PDO::PARAM_STR);
$stmt->execute();

return $stmt->fetchAll();
}
}

?>

```

Inventory.php

```

<?php

class Inventory {
    private $conn;
    private $table = 'inventory';

    public function __construct($db) {
        $this->conn = $db;
    }

    public function getAll() {
        $query = "SELECT i.*, p.product_name, p.sku, p.price, c.category_name,
CASE
    WHEN i.quantity_available <= i.reorder_level THEN 'low'

```

```

        WHEN i.quantity_available <= (i.reorder_level * 2) THEN 'medium'
        ELSE 'good'
    END as stock_status
    FROM " . $this->table . " i
    JOIN products p ON i.product_id = p.product_id
    JOIN categories c ON p.category_id = c.category_id
    WHERE p.is_active = 1
    ORDER BY i.quantity_available ASC, p.product_name";

$stmt = $this->conn->prepare($query);
$stmt->execute();

return $stmt->fetchAll();
}

public function getByProductId($product_id) {
$query = "SELECT i.*, p.product_name, p.sku
        FROM " . $this->table . " i
        JOIN products p ON i.product_id = p.product_id
        WHERE i.product_id = :product_id";

$stmt = $this->conn->prepare($query);
$stmt->bindParam(':product_id', $product_id, PDO::PARAM_INT);
$stmt->execute();

return $stmt->fetch();
}

public function updateQuantity($product_id, $quantity, $user_id) {
$query = "UPDATE " . $this->table . "
        SET quantity_available = :quantity,
        last_updated_by = :user_id
        WHERE product_id = :product_id";

$stmt = $this->conn->prepare($query);
$stmt->bindParam(':quantity', $quantity, PDO::PARAM_INT);
$stmt->bindParam(':product_id', $product_id, PDO::PARAM_INT);
$stmt->bindParam(':user_id', $user_id, PDO::PARAM_INT);

return $stmt->execute();
}

public function adjustQuantity($product_id, $adjustment, $user_id) {
$query = "UPDATE " . $this->table . "

```

```

        SET quantity_available = quantity_available + :adjustment,
        last_updated_by = :user_id
        WHERE product_id = :product_id";

$stmt = $this->conn->prepare($query);
$stmt->bindParam(':adjustment', $adjustment, PDO::PARAM_INT);
$stmt->bindParam(':product_id', $product_id, PDO::PARAM_INT);
$stmt->bindParam(':user_id', $user_id, PDO::PARAM_INT);

return $stmt->execute();
}

public function getLowStock() {
$query = "SELECT i.*, p.product_name, p.sku, c.category_name
    FROM " . $this->table . " i
    JOIN products p ON i.product_id = p.product_id
    JOIN categories c ON p.category_id = c.category_id
    WHERE p.is_active = 1
    AND i.quantity_available <= i.reorder_level
    ORDER BY i.quantity_available ASC";

$stmt = $this->conn->prepare($query);
$stmt->execute();

return $stmt->fetchAll();
}

public function getStatistics() {
$query = "SELECT
    COUNT(*) as total_products,
    SUM(quantity_available) as total_stock,
    SUM(CASE WHEN quantity_available <= reorder_level THEN 1 ELSE 0 END) as
low_stock_count,
    SUM(CASE WHEN quantity_available = 0 THEN 1 ELSE 0 END) as out_of_stock_count,
    AVG(quantity_available) as avg_stock
    FROM " . $this->table . " i
    JOIN products p ON i.product_id = p.product_id
    WHERE p.is_active = 1";

$stmt = $this->conn->prepare($query);
$stmt->execute();

return $stmt->fetch();
}

```

```
}
```

Order.php

```
<?php

class Order {
    private $conn;
    private $table = 'orders';

    public function __construct($db) {
        $this->conn = $db;
    }

    public function createOrder($customer_id, $shipping_address, $notes = null) {
        try {
            $this->conn->beginTransaction();

            $cart_query = "SELECT sc.product_id, sc.quantity, p.price
                           FROM shopping_cart sc
                           JOIN products p ON sc.product_id = p.product_id
                           WHERE sc.customer_id = :customer_id";

            $cart_stmt = $this->conn->prepare($cart_query);
            $cart_stmt->bindParam(':customer_id', $customer_id, PDO::PARAM_INT);
            $cart_stmt->execute();
            $cart_items = $cart_stmt->fetchAll();

            if (empty($cart_items)) {
                throw new Exception('Cart is empty');
            }

            $total = 0;
            foreach ($cart_items as $item) {
                $total += $item['quantity'] * $item['price'];
            }

            $order_query = "INSERT INTO " . $this->table . "
                           (customer_id, total_amount, shipping_address_line1, shipping_address_line2,
                            shipping_city, shipping_state, shipping_zip, shipping_country, notes)
                           VALUES (:customer_id, :total, :addr1, :addr2, :city, :state, :zip, :country, :notes)";

            $order_stmt = $this->conn->prepare($order_query);
            $order_stmt->bindParam(':customer_id', $customer_id, PDO::PARAM_INT);
```

```

$order_stmt->bindParam(':total', $total);
$order_stmt->bindParam(':addr1', $shipping_address['address_line1']);
$order_stmt->bindParam(':addr2', $shipping_address['address_line2']);
$order_stmt->bindParam(':city', $shipping_address['city']);
$order_stmt->bindParam(':state', $shipping_address['state']);
$order_stmt->bindParam(':zip', $shipping_address['zip_code']);
$order_stmt->bindParam(':country', $shipping_address['country']);
$order_stmt->bindParam(':notes', $notes);
$order_stmt->execute();

$order_id = $this->conn->lastInsertId();

$item_query = "INSERT INTO order_items (order_id, product_id, quantity, unit_price, subtotal)
    VALUES (:order_id, :product_id, :quantity, :price, :subtotal)";

$inv_query = "UPDATE inventory SET quantity_available = quantity_available - :quantity
    WHERE product_id = :product_id";

$item_stmt = $this->conn->prepare($item_query);
$inv_stmt = $this->conn->prepare($inv_query);

foreach ($cart_items as $item) {
    $subtotal = $item['quantity'] * $item['price'];

    $item_stmt->bindParam(':order_id', $order_id, PDO::PARAM_INT);
    $item_stmt->bindParam(':product_id', $item['product_id'], PDO::PARAM_INT);
    $item_stmt->bindParam(':quantity', $item['quantity'], PDO::PARAM_INT);
    $item_stmt->bindParam(':price', $item['price']);
    $item_stmt->bindParam(':subtotal', $subtotal);
    $item_stmt->execute();

    $inv_stmt->bindParam(':quantity', $item['quantity'], PDO::PARAM_INT);
    $inv_stmt->bindParam(':product_id', $item['product_id'], PDO::PARAM_INT);
    $inv_stmt->execute();
}

$clear_cart = "DELETE FROM shopping_cart WHERE customer_id = :customer_id";
$clear_stmt = $this->conn->prepare($clear_cart);
$clear_stmt->bindParam(':customer_id', $customer_id, PDO::PARAM_INT);
$clear_stmt->execute();

$this->conn->commit();
return $order_id;

```

```

} catch (Exception $e) {
$this->conn->rollBack();
error_log("Order creation error: " . $e->getMessage());
return false;
}

public function getOrderById($order_id) {
$query = "SELECT o.*, c.first_name, c.last_name, c.phone, u.email
FROM " . $this->table . " o
JOIN customers c ON o.customer_id = c.customer_id
JOIN users u ON c.user_id = u.user_id
WHERE o.order_id = :order_id";

$stmt = $this->conn->prepare($query);
$stmt->bindParam(':order_id', $order_id, PDO::PARAM_INT);
$stmt->execute();

return $stmt->fetch();
}

public function getById($order_id) {
return $this->getOrderById($order_id);
}

public function getOrderItems($order_id) {
$query = "SELECT oi.*, p.product_name, p.sku, p.image_url
FROM order_items oi
JOIN products p ON oi.product_id = p.product_id
WHERE oi.order_id = :order_id";

$stmt = $this->conn->prepare($query);
$stmt->bindParam(':order_id', $order_id, PDO::PARAM_INT);
$stmt->execute();

return $stmt->fetchAll();
}

public function getCustomerOrders($customer_id, $limit = null) {
$query = "SELECT * FROM " . $this->table . "
WHERE customer_id = :customer_id
ORDER BY order_date DESC";

if ($limit) {

```

```

$query .= " LIMIT :limit";
}

$stmt = $this->conn->prepare($query);
$stmt->bindParam(':customer_id', $customer_id, PDO::PARAM_INT);

if ($limit) {
$stmt->bindParam(':limit', $limit, PDO::PARAM_INT);
}

$stmt->execute();
return $stmt->fetchAll();
}

public function getAllOrders($status = null, $limit = null, $offset = 0) {
$query = "SELECT o.*, c.first_name, c.last_name, c.phone
    FROM " . $this->table . " o
    JOIN customers c ON o.customer_id = c.customer_id";

if ($status) {
$query .= " WHERE o.status = :status";
}

$query .= " ORDER BY o.order_date DESC";

if ($limit) {
$query .= " LIMIT :limit OFFSET :offset";
}

$stmt = $this->conn->prepare($query);

if ($status) {
$stmt->bindParam(':status', $status, PDO::PARAM_STR);
}

if ($limit) {
$stmt->bindParam(':limit', $limit, PDO::PARAM_INT);
$stmt->bindParam(':offset', $offset, PDO::PARAM_INT);
}

$stmt->execute();
return $stmt->fetchAll();
}

```

```

public function updateStatus($order_id, $status, $employee_id = null) {
    $query = "UPDATE " . $this->table . " SET status = :status";

    if (($status === 'Delivered' || $status === 'Shipped') && $employee_id) {
        $query .= ", fulfilled_by = :employee_id, fulfilled_at = NOW()";
    }

    $query .= " WHERE order_id = :order_id";

    $stmt = $this->conn->prepare($query);
    $stmt->bindParam(':status', $status, PDO::PARAM_STR);
    $stmt->bindParam(':order_id', $order_id, PDO::PARAM_INT);

    if (($status === 'Delivered' || $status === 'Shipped') && $employee_id) {
        $stmt->bindParam(':employee_id', $employee_id, PDO::PARAM_INT);
    }

    return $stmt->execute();
}

public function getStatistics() {
    $query = "SELECT
        COUNT(*) as total_orders,
        SUM(CASE WHEN status = 'Pending' THEN 1 ELSE 0 END) as pending_orders,
        SUM(CASE WHEN status = 'Processing' THEN 1 ELSE 0 END) as processing_orders,
        SUM(CASE WHEN status = 'Shipped' THEN 1 ELSE 0 END) as shipped_orders,
        SUM(CASE WHEN status = 'Delivered' THEN 1 ELSE 0 END) as delivered_orders,
        SUM(total_amount) as total_revenue,
        AVG(total_amount) as average_order_value
        FROM " . $this->table;

    $stmt = $this->conn->prepare($query);
    $stmt->execute();

    return $stmt->fetch();
}
?>

```

Product.php

```

<?php

class Product {
    private $conn;

```

```

private $table = 'products';

public function __construct($db) {
    $this->conn = $db;
}

public function getAll($category_id = null, $limit = null, $offset = 0) {
    $query = "SELECT p.*, c.category_name, i.quantity_available
        FROM " . $this->table . " p
        JOIN categories c ON p.category_id = c.category_id
        LEFT JOIN inventory i ON p.product_id = i.product_id
        WHERE p.is_active = 1";

    if ($category_id) {
        $query .= " AND p.category_id = :category_id";
    }

    $query .= " ORDER BY p.created_at DESC";

    if ($limit) {
        $query .= " LIMIT :limit OFFSET :offset";
    }

    $stmt = $this->conn->prepare($query);

    if ($category_id) {
        $stmt->bindParam(':category_id', $category_id, PDO::PARAM_INT);
    }

    if ($limit) {
        $stmt->bindParam(':limit', $limit, PDO::PARAM_INT);
        $stmt->bindParam(':offset', $offset, PDO::PARAM_INT);
    }

    $stmt->execute();
    return $stmt->fetchAll();
}

public function getById($product_id) {
    $query = "SELECT p.*, c.category_name, i.quantity_available, i.quantity_reserved
        FROM " . $this->table . " p
        JOIN categories c ON p.category_id = c.category_id
        LEFT JOIN inventory i ON p.product_id = i.product_id
        WHERE p.product_id = :product_id";
}

```

```

$stmt = $this->conn->prepare($query);
$stmt->bindParam(':product_id', $product_id, PDO::PARAM_INT);
$stmt->execute();

return $stmt->fetch();
}

public function search($search_term) {
$query = "SELECT p.*, c.category_name, i.quantity_available
    FROM " . $this->table . " p
    JOIN categories c ON p.category_id = c.category_id
    LEFT JOIN inventory i ON p.product_id = i.product_id
    WHERE p.is_active = 1
    AND (p.product_name LIKE :search1
    OR p.description LIKE :search2
    OR p.sku LIKE :search3
    OR c.category_name LIKE :search4)
    ORDER BY p.product_name";

$stmt = $this->conn->prepare($query);
$search_param = '%' . $search_term . '%';
$stmt->execute([
':search1' => $search_param,
':search2' => $search_param,
':search3' => $search_param,
':search4' => $search_param
]);

return $stmt->fetchAll();
}

public function create($data) {
$query = "INSERT INTO " . $this->table . "
(category_id, product_name, description, price, sku, image_url, created_by)
VALUES (:category_id, :product_name, :description, :price, :sku, :image_url, :created_by)";

$stmt = $this->conn->prepare($query);

$stmt->bindParam(':category_id', $data['category_id'], PDO::PARAM_INT);
$stmt->bindParam(':product_name', $data['product_name'], PDO::PARAM_STR);
$stmt->bindParam(':description', $data['description'], PDO::PARAM_STR);
$stmt->bindParam(':price', $data['price'], PDO::PARAM_STR);
$stmt->bindParam(':sku', $data['sku'], PDO::PARAM_STR);

```

```

$stmt->bindParam(':image_url', $data['image_url'], PDO::PARAM_STR);
$stmt->bindParam(':created_by', $data['created_by'], PDO::PARAM_INT);

if ($stmt->execute()) {
    $product_id = $this->conn->lastInsertId();

    $inv_query = "INSERT INTO inventory (product_id, quantity_available, last_updated_by)
                  VALUES (:product_id, :quantity, :updated_by)";
    $inv_stmt = $this->conn->prepare($inv_query);
    $inv_stmt->bindParam(':product_id', $product_id, PDO::PARAM_INT);
    $inv_stmt->bindParam(':quantity', $data['initial_quantity'] ?? 0, PDO::PARAM_INT);
    $inv_stmt->bindParam(':updated_by', $data['created_by'], PDO::PARAM_INT);
    $inv_stmt->execute();

    return $product_id;
}

return false;
}

public function update($product_id, $data) {
    $query = "UPDATE " . $this->table . "
              SET category_id = :category_id,
                  product_name = :product_name,
                  description = :description,
                  price = :price,
                  sku = :sku,
                  image_url = :image_url,
                  is_active = :is_active
              WHERE product_id = :product_id";

    $stmt = $this->conn->prepare($query);

    $stmt->bindParam(':product_id', $product_id, PDO::PARAM_INT);
    $stmt->bindParam(':category_id', $data['category_id'], PDO::PARAM_INT);
    $stmt->bindParam(':product_name', $data['product_name'], PDO::PARAM_STR);
    $stmt->bindParam(':description', $data['description'], PDO::PARAM_STR);
    $stmt->bindParam(':price', $data['price'], PDO::PARAM_STR);
    $stmt->bindParam(':sku', $data['sku'], PDO::PARAM_STR);
    $stmt->bindParam(':image_url', $data['image_url'], PDO::PARAM_STR);
    $stmt->bindParam(':is_active', $data['is_active'] ?? 1, PDO::PARAM_INT);

    return $stmt->execute();
}

```

```

public function delete($product_id) {
    $query = "UPDATE " . $this->table . " SET is_active = 0 WHERE product_id = :product_id";

    $stmt = $this->conn->prepare($query);
    $stmt->bindParam(':product_id', $product_id, PDO::PARAM_INT);

    return $stmt->execute();
}

public function getCount($category_id = null) {
    $query = "SELECT COUNT(*) as count FROM " . $this->table . " WHERE is_active = 1";

    if ($category_id) {
        $query .= " AND category_id = :category_id";
    }

    $stmt = $this->conn->prepare($query);

    if ($category_id) {
        $stmt->bindParam(':category_id', $category_id, PDO::PARAM_INT);
    }

    $stmt->execute();
    $result = $stmt->fetch();

    return $result['count'];
}

public function skuExists($sku, $exclude_product_id = null) {
    $query = "SELECT product_id FROM " . $this->table . " WHERE sku = :sku";

    if ($exclude_product_id) {
        $query .= " AND product_id != :exclude_product_id";
    }

    $stmt = $this->conn->prepare($query);
    $stmt->bindParam(':sku', $sku, PDO::PARAM_STR);

    if ($exclude_product_id) {
        $stmt->bindParam(':exclude_product_id', $exclude_product_id, PDO::PARAM_INT);
    }

    $stmt->execute();
}

```

```

return $stmt->rowCount() > 0;
}

public function getLowStock() {
$query = "SELECT p.*, c.category_name, i.quantity_available, i.reorder_level
    FROM " . $this->table . " p
    JOIN categories c ON p.category_id = c.category_id
    JOIN inventory i ON p.product_id = i.product_id
    WHERE p.is_active = 1
    AND i.quantity_available <= i.reorder_level
    ORDER BY i.quantity_available ASC";

$stmt = $this->conn->prepare($query);
$stmt->execute();

return $stmt->fetchAll();
}
?>

```

User.php

```

<?php

class User {
    private $conn;
    private $table = 'users';

    public function __construct($db) {
$this->conn = $db;
    }

    public function getUserById($user_id) {
$query = "SELECT u.*, r.role_name
    FROM " . $this->table . " u
    JOIN roles r ON u.role_id = r.role_id
    WHERE u.user_id = :user_id";

$stmt = $this->conn->prepare($query);
$stmt->bindParam(':user_id', $user_id, PDO::PARAM_INT);
$stmt->execute();

return $stmt->fetch();
    }

```

```

public function getUserByUsername($username) {
    $query = "SELECT u.*, r.role_name
        FROM " . $this->table . " u
        JOIN roles r ON u.role_id = r.role_id
        WHERE (u.username = :username OR u.email = :username)
        AND u.is_active = 1";

    $stmt = $this->conn->prepare($query);
    $stmt->bindParam(':username', $username, PDO::PARAM_STR);
    $stmt->execute();

    return $stmt->fetch();
}

public function getByUsername($username) {
    $query = "SELECT u.*, r.role_name
        FROM " . $this->table . " u
        JOIN roles r ON u.role_id = r.role_id
        WHERE u.username = :username";

    $stmt = $this->conn->prepare($query);
    $stmt->bindParam(':username', $username, PDO::PARAM_STR);
    $stmt->execute();

    return $stmt->fetch();
}

public function getByEmail($email) {
    $query = "SELECT u.*, r.role_name
        FROM " . $this->table . " u
        JOIN roles r ON u.role_id = r.role_id
        WHERE u.email = :email";

    $stmt = $this->conn->prepare($query);
    $stmt->bindParam(':email', $email, PDO::PARAM_STR);
    $stmt->execute();

    return $stmt->fetch();
}

public function verifyPassword($password, $hash) {
    return password_verify($password, $hash);
}

```

```

public function createUser($username, $email, $password, $role_id = ROLE_CUSTOMER) {
    $query = "INSERT INTO " . $this->table . "
        (username, email, password_hash, role_id)
        VALUES (:username, :email, :password_hash, :role_id)";

    $stmt = $this->conn->prepare($query);

    $password_hash = password_hash($password, PASSWORD_BCRYPT, ['cost' => BCRYPT_COST]);

    $stmt->bindParam(':username', $username, PDO::PARAM_STR);
    $stmt->bindParam(':email', $email, PDO::PARAM_STR);
    $stmt->bindParam(':password_hash', $password_hash, PDO::PARAM_STR);
    $stmt->bindParam(':role_id', $role_id, PDO::PARAM_INT);

    if ($stmt->execute()) {
        return $this->conn->lastInsertId();
    }

    return false;
}

public function updatePassword($user_id, $new_password) {
    $query = "UPDATE " . $this->table . "
        SET password_hash = :password_hash
        WHERE user_id = :user_id";

    $stmt = $this->conn->prepare($query);

    $password_hash = password_hash($new_password, PASSWORD_BCRYPT, ['cost' => BCRYPT_COST]);

    $stmt->bindParam(':password_hash', $password_hash, PDO::PARAM_STR);
    $stmt->bindParam(':user_id', $user_id, PDO::PARAM_INT);

    return $stmt->execute();
}

public function usernameExists($username, $exclude_user_id = null) {
    $query = "SELECT user_id FROM " . $this->table . " WHERE username = :username";

    if ($exclude_user_id) {
        $query .= " AND user_id != :exclude_user_id";
    }
}

```

```

$stmt = $this->conn->prepare($query);
$stmt->bindParam(':username', $username, PDO::PARAM_STR);

if ($exclude_user_id) {
$stmt->bindParam(':exclude_user_id', $exclude_user_id, PDO::PARAM_INT);
}

$stmt->execute();
return $stmt->rowCount() > 0;
}

public function emailExists($email, $exclude_user_id = null) {
$query = "SELECT user_id FROM " . $this->table . " WHERE email = :email";

if ($exclude_user_id) {
$query .= " AND user_id != :exclude_user_id";
}

$stmt = $this->conn->prepare($query);
$stmt->bindParam(':email', $email, PDO::PARAM_STR);

if ($exclude_user_id) {
$stmt->bindParam(':exclude_user_id', $exclude_user_id, PDO::PARAM_INT);
}

$stmt->execute();
return $stmt->rowCount() > 0;
}

public function getAllUsers($role_id = null) {
$query = "SELECT u.*, r.role_name
FROM " . $this->table . " u
JOIN roles r ON u.role_id = r.role_id";

if ($role_id) {
$query .= " WHERE u.role_id = :role_id";
}

$query .= " ORDER BY u.created_at DESC";

$stmt = $this->conn->prepare($query);

if ($role_id) {
$stmt->bindParam(':role_id', $role_id, PDO::PARAM_INT);
}

```

```

        }

        $stmt->execute();
        return $stmt->fetchAll();
    }

    public function deactivateUser($user_id) {
        $query = "UPDATE " . $this->table . " SET is_active = 0 WHERE user_id = :user_id";

        $stmt = $this->conn->prepare($query);
        $stmt->bindParam(':user_id', $user_id, PDO::PARAM_INT);

        return $stmt->execute();
    }

    public function activateUser($user_id) {
        $query = "UPDATE " . $this->table . " SET is_active = 1 WHERE user_id = :user_id";

        $stmt = $this->conn->prepare($query);
        $stmt->bindParam(':user_id', $user_id, PDO::PARAM_INT);

        return $stmt->execute();
    }
}
?>

```

Dashboard.php

```

<?php

require_once '../config/config.php';
require_once '../includes/session.php';

requireRole(ROLE_BUSINESS_MANAGER);

$page_title = 'Admin Dashboard';

$database = new Database();
$db = $database->getConnection();

require_once MODEL_PATH . 'Order.php';
require_once MODEL_PATH . 'Product.php';
require_once MODEL_PATH . 'Customer.php';

$order_model = new Order($db);

```

```

$product_model = new Product($db);

$order_stats = $order_model->getStatistics();

$customer_query = "SELECT COUNT(*) as count FROM customers";
$customer_stmt = $db->prepare($customer_query);
$customer_stmt->execute();
$customer_count = $customer_stmt->fetch()['count'];

$revenue_query = "SELECT SUM(total_amount) as revenue FROM orders WHERE status != 'Cancelled'";
$revenue_stmt = $db->prepare($revenue_query);
$revenue_stmt->execute();
$total_revenue = $revenue_stmt->fetch()['revenue'] ?? 0;

$recent_orders = $order_model->getAllOrders(null, 5);

$low_stock_query =
    SELECT p.product_id, p.product_name, p.sku, i.quantity_available, i.reorder_level
    FROM products p
    INNER JOIN inventory i ON p.product_id = i.product_id
    WHERE i.quantity_available <= i.reorder_level
    ORDER BY i.quantity_available ASC
    LIMIT 10
";
$low_stock_stmt = $db->prepare($low_stock_query);
$low_stock_stmt->execute();
$low_stock = $low_stock_stmt->fetchAll();

$sales_query =
    SELECT
        DATE_FORMAT(order_date, '%Y-%m') as month,
        COUNT(*) as order_count,
        SUM(total_amount) as revenue
    FROM orders
    WHERE status != 'Cancelled'
    AND order_date >= DATE_SUB(NOW(), INTERVAL 6 MONTH)
    GROUP BY DATE_FORMAT(order_date, '%Y-%m')
    ORDER BY month ASC
";
$sales_stmt = $db->prepare($sales_query);
$sales_stmt->execute();
$monthly_sales = $sales_stmt->fetchAll();

include '../includes/header.php';

```

```

?>

<div class="container-fluid my-4">
    <h1 class="text-hollywood mb-4">đŸŽ¬ Admin Dashboard</h1>

    <!-- Key Metrics -->
    <div class="row mb-4">
        <div class="col-md-3">
            <div class="dashboard-card">
                <h3><?php echo formatCurrency($total_revenue); ?></h3>
                <p>Total Revenue</p>
                <small class="text-muted">All-time</small>
            </div>
        </div>
        <div class="col-md-3">
            <div class="dashboard-card" style="border-left-color: #28a745;">
                <h3><?php echo number_format($order_stats['total_orders']); ?></h3>
                <p>Total Orders</p>
                <small class="text-muted"><?php echo $order_stats['pending_orders']; ?> pending</small>
            </div>
        </div>
        <div class="col-md-3">
            <div class="dashboard-card" style="border-left-color: #17a2b8;">
                <h3><?php echo number_format($customer_count); ?></h3>
                <p>Total Customers</p>
            </div>
        </div>
        <div class="col-md-3">
            <div class="dashboard-card" style="border-left-color: #ffc107;">
                <h3><?php echo number_format(count($low_stock)); ?></h3>
                <p>Low Stock Items</p>
                <small class="text-muted">Needs attention</small>
            </div>
        </div>
    </div>

    <div class="row">
        <!-- Recent Orders -->
        <div class="col-md-6 mb-4">
            <div class="card card-hollywood">
                <div class="card-header card-header-hollywood">
                    <i class="fas fa-shopping-cart"></i> Recent Orders
                </div>
                <div class="card-body">

```

```

<?php if (empty($recent_orders)): ?>
<p class="text-muted text-center">No recent orders</p>
<?php else: ?>
<div class="table-responsive">
    <table class="table table-sm">
        <thead>
            <tr>
                <th>Order #</th>
                <th>Customer</th>
                <th>Total</th>
                <th>Status</th>
            </tr>
        </thead>
        <tbody>
            <?php foreach ($recent_orders as $order): ?>
            <tr>
                <td>
                    <a href="..../employee/order_view.php?id=<?php echo $order['order_id']; ?>">
                        #<?php echo $order['order_id']; ?>
                    </a>
                </td>
                <td><?php echo h($order['first_name'] . ' ' . $order['last_name']); ?></td>
                <td><?php echo formatCurrency($order['total_amount']); ?></td>
                <td>
                    <span class="badge status-<?php echo strtolower($order['status']); ?>">
                        <?php echo h($order['status']); ?>
                    </span>
                </td>
            </tr>
        <?php endforeach; ?>
    </tbody>
</table>
</div>
<a href="..../employee/orders.php" class="btn btn-sm btn-outline-hollywood">
    View All Orders <i class="fas fa-arrow-right"></i>
</a>
<?php endif; ?>
</div>
</div>

<!-- Low Stock Alert -->
<div class="col-md-6 mb-4">
<div class="card card-hollywood">

```

```

<div class="card-header card-header-hollywood">
<i class="fas fa-exclamation-triangle"></i> Low Stock Alert
</div>
<div class="card-body">
<?php if (empty($low_stock)): ?>
<p class="text-muted text-center">All products are well stocked!</p>
<?php else: ?>
<div class="table-responsive">
    <table class="table table-sm">
        <thead>
            <tr>
                <th>Product</th>
                <th>Current</th>
                <th>Reorder</th>
                <th>Status</th>
            </tr>
        </thead>
        <tbody>
            <?php foreach ($low_stock as $item): ?>
            <tr>
                <td>
                    <strong><?php echo h($item['product_name']); ?></strong><br>
                    <small class="text-muted"><?php echo h($item['sku']); ?></small>
                </td>
                <td><?php echo $item['quantity_available']; ?></td>
                <td><?php echo $item['reorder_level']; ?></td>
                <td>
                    <?php if ($item['quantity_available'] == 0): ?>
                        <span class="badge bg-danger">Out of Stock</span>
                    <?php else: ?>
                        <span class="badge bg-warning">Low Stock</span>
                    <?php endif; ?>
                </td>
            </tr>
        <?php endforeach; ?>
    </tbody>
</table>
</div>
<a href="../employee/inventory.php" class="btn btn-sm btn-outline-hollywood">
    Manage Inventory <i class="fas fa-arrow-right"></i>
</a>
<?php endif; ?>
</div>
</div>

```

```

</div>
</div>

<!-- Sales Chart -->
<div class="row">
<div class="col-12 mb-4">
<div class="card card-hollywood">
    <div class="card-header card-header-hollywood">
        <i class="fas fa-chart-line"></i> Sales Trend (Last 6 Months)
    </div>
    <div class="card-body">
        <?php if (empty($monthly_sales)): ?>
        <p class="text-muted text-center">No sales data available</p>
        <?php else: ?>
        <canvas id="salesChart" height="80"></canvas>
        <?php endif; ?>
    </div>
</div>
</div>
</div>
</div>

<?php if (!empty($monthly_sales)): ?>
<script src="https://cdn.jsdelivr.net/npm/chart.js@4.4.0/dist/chart.umd.min.js"></script>
<script>
const ctx = document.getElementById('salesChart');
const salesData = <?php echo json_encode($monthly_sales); ?>

new Chart(ctx, {
    type: 'line',
    data: {
        labels: salesData.map(d => d.month),
        datasets: [
            {
                label: 'Revenue ($)',
                data: salesData.map(d => parseFloat(d.revenue)),
                borderColor: '#d4af37',
                backgroundColor: 'rgba(212, 175, 55, 0.1)',
                tension: 0.4,
                fill: true,
                yAxisID: 'y'
            },
            {
                label: 'Orders',

```

```

        data: salesData.map(d => parseInt(d.order_count)),
        borderColor: '#17a2b8',
        backgroundColor: 'rgba(23, 162, 184, 0.1)',
        tension: 0.4,
        fill: true,
        yAxisID: 'y1'
    }
]
},
options: {
responsive: true,
maintainAspectRatio: true,
interaction: {
mode: 'index',
intersect: false,
},
scales: {
y: {
type: 'linear',
display: true,
position: 'left',
title: {
display: true,
text: 'Revenue ($)'
}
},
y1: {
type: 'linear',
display: true,
position: 'right',
title: {
display: true,
text: 'Number of Orders'
},
grid: {
drawOnChartArea: false,
}
}
}
}
}),
</script>
<?php endif; ?>
```

```

<?php include '../includes/footer.php'; ?>

Employee.php
<?php

require_once '../config/config.php';
require_once '../includes/session.php';

requireRole(ROLE_MANAGER);

$page_title = 'Employee Management';

$database = new Database();
$db = $database->getConnection();

if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['create_employee'])) {
    if (!verifyCSRFToken($_POST['csrf_token'] ?? '')) {
        setFlashMessage('error', 'Invalid security token.');
    } else {
        $username = trim($_POST['username']);
        $email = trim($_POST['email']);
        $password = $_POST['password'];
        $role_id = (int)$_POST['role_id'];
        $first_name = trim($_POST['first_name']);
        $last_name = trim($_POST['last_name']);
        $position = trim($_POST['position']);
        $department = trim($_POST['department']);
        $hire_date = $_POST['hire_date'];
        $salary = (float)$_POST['salary'];

        try {
            $db->beginTransaction();

            $password_hash = password_hash($password, PASSWORD_BCRYPT, ['cost' => 10]);
            $user_query = "INSERT INTO users (username, email, password_hash, role_id)
                          VALUES (:username, :email, :password, :role_id)";
            $user_stmt = $db->prepare($user_query);
            $user_stmt->execute([
                ':username' => $username,
                ':email' => $email,
                ':password' => $password_hash,
                ':role_id' => $role_id
            ]);
            $user_id = $db->lastInsertId();
        }
    }
}

```

```

$emp_query = "INSERT INTO employees (user_id, first_name, last_name, position,
    department, hire_date, salary, is_active)
VALUES (:uid, :fn, :ln, :pos, :dept, :hire, :sal, 1)";
$emp_stmt = $db->prepare($emp_query);
$emp_stmt->execute([
    ':uid' => $user_id,
    ':fn' => $first_name,
    ':ln' => $last_name,
    ':pos' => $position,
    ':dept' => $department,
    ':hire' => $hire_date,
    ':sal' => $salary
]);
$db->commit();
logAudit('CREATE_EMPLOYEE', 'employees', $user_id, null, "$first_name $last_name");
setFlashMessage('success', 'Employee created successfully!');
} catch (Exception $e) {
$db->rollBack();
setFlashMessage('error', 'Error: ' . $e->getMessage());
}
redirect('public/admin/employees.php');
}

if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['update_employee'])) {
    if (!verifyCSRFToken($_POST['csrf_token'] ?? '')) {
        setFlashMessage('error', 'Invalid security token.');
    } else {
        $employee_id = (int)$_POST['employee_id'];
        $first_name = trim($_POST['first_name']);
        $last_name = trim($_POST['last_name']);
        $position = trim($_POST['position']);
        $department = trim($_POST['department']);
        $salary = (float)$_POST['salary'];
        $is_active = isset($_POST['is_active']) ? 1 : 0;

        $query = "UPDATE employees SET
            first_name = :fn, last_name = :ln, position = :pos,
            department = :dept, salary = :sal, is_active = :active
            WHERE employee_id = :eid";
        $stmt = $db->prepare($query);

```

```

if ($stmt->execute([
    ':fn' => $first_name,
    ':ln' => $last_name,
    ':pos' => $position,
    ':dept' => $department,
    ':sal' => $salary,
    ':active' => $is_active,
    ':eid' => $employee_id
])) {
    logAudit('UPDATE_EMPLOYEE', 'employees', $employee_id, null, "$first_name $last_name");
    setFlashMessage('success', 'Employee updated successfully!');
} else {
    setFlashMessage('error', 'Failed to update employee.');
}
redirect('public/admin/employees.php');
}

$emp_query = "
    SELECT e.*, u.username, u.email, r.role_name
    FROM employees e
    INNER JOIN users u ON e.user_id = u.user_id
    INNER JOIN roles r ON u.role_id = r.role_id
    ORDER BY e.hire_date DESC
";
$emp_stmt = $db->prepare($emp_query);
$emp_stmt->execute();
$employees = $emp_stmt->fetchAll();

$role_query = "SELECT * FROM roles WHERE role_id >= 2 ORDER BY role_name";
$role_stmt = $db->prepare($role_query);
$role_stmt->execute();
$roles = $role_stmt->fetchAll();

include '../includes/header.php';
?>

<div class="container-fluid my-4">
    <div class="d-flex justify-content-between align-items-center mb-4">
        <h1 class="text-hollywood">Employee Management</h1>
        <button class="btn btn-hollywood" data-bs-toggle="modal" data-bs-target="#createEmployeeModal">
            <i class="fas fa-user-plus"></i> Add Employee
        </button>

```

```

</div>

<!-- Statistics -->
<div class="row mb-4">
<div class="col-md-4">
<div class="dashboard-card">
    <h3><?php echo number_format(count($employees)); ?></h3>
    <p>Total Employees</p>
</div>
</div>
<div class="col-md-4">
<div class="dashboard-card" style="border-left-color: #28a745;">
    <h3>
        <?php echo number_format(count(array_filter($employees, fn($e) => $e['is_active']))); ?>
    </h3>
    <p>Active Employees</p>
</div>
</div>
<div class="col-md-4">
<div class="dashboard-card" style="border-left-color: #dc3545;">
    <h3>
        <?php echo number_format(count(array_filter($employees, fn($e) => !$e['is_active']))); ?>
    </h3>
    <p>Inactive Employees</p>
</div>
</div>
</div>

<!-- Employees Table -->
<div class="card card-hollywood">
<div class="card-header card-header-hollywood">
<i class="fas fa-users"></i> Employees
</div>
<div class="card-body">
<div class="table-responsive">
    <table class="table table-hover">
        <thead class="table-light">
            <tr>
                <th>ID</th>
                <th>Name</th>
                <th>Position</th>
                <th>Department</th>
                <th>Role</th>
                <th>Hire Date</th>

```

```

<th>Salary</th>
<th>Status</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<?php foreach ($employees as $employee): ?>
<tr>
<td><?php echo $employee['employee_id']; ?></td>
<td>
<strong><?php echo h($employee['first_name']) . ' ' . $employee['last_name']); ?></strong><br>
<small class="text-muted"><?php echo h($employee['email']); ?></small>
</td>
<td><?php echo h($employee['position']); ?></td>
<td><?php echo h($employee['department']); ?></td>
<td>
<span class="badge bg-info">
    <?php echo h($employee['role_name']); ?>
</span>
</td>
<td><?php echo formatDate($employee['hire_date']); ?></td>
<td><?php echo formatCurrency($employee['salary']); ?></td>
<td>
<?php if ($employee['is_active']): ?>
    <span class="badge bg-success">Active</span>
<?php else: ?>
    <span class="badge bg-secondary">Inactive</span>
<?php endif; ?>
</td>
<td>
<button class="btn btn-sm btn-outline-hollywood"
    onclick="editEmployee(<?php echo htmlspecialchars(json_encode($employee)); ?>)">
    <i class="fas fa-edit"></i>
</button>
</td>
</tr>
<?php endforeach; ?>
</tbody>
</table>
</div>
</div>
</div>

```

```

</div>

<!-- Create Employee Modal -->
<div class="modal fade" id="createEmployeeModal" tabindex="-1">
    <div class="modal-dialog modal-lg">
        <div class="modal-content">
            <form method="POST" action="">
                <input type="hidden" name="csrf_token" value="<?php echo generateCSRFToken(); ?>">

                <div class="modal-header bg-hollywood text-white">
                    <h5 class="modal-title">Add New Employee</h5>
                    <button type="button" class="btn-close btn-close-white" data-bs-dismiss="modal"></button>
                </div>
                <div class="modal-body">
                    <h6 class="text-hollywood mb-3">User Account</h6>
                    <div class="row">
                        <div class="col-md-6 mb-3">
                            <label for="create_username" class="form-label">Username *</label>
                            <input type="text" class="form-control" name="username" id="create_username" required>
                        </div>
                        <div class="col-md-6 mb-3">
                            <label for="create_email" class="form-label">Email *</label>
                            <input type="email" class="form-control" name="email" id="create_email" required>
                        </div>
                    </div>
                    <div class="row">
                        <div class="col-md-6 mb-3">
                            <label for="create_password" class="form-label">Password *</label>
                            <input type="password" class="form-control" name="password" id="create_password" minlength="6" required>
                        </div>
                        <div class="col-md-6 mb-3">
                            <label for="create_role_id" class="form-label">Role *</label>
                            <select class="form-select" name="role_id" id="create_role_id" required>
                                <option value="">Select Role</option>
                                <?php foreach ($roles as $role): ?>
                                    <option value="<?php echo $role['role_id']; ?>">
                                        <?php echo h($role['role_name']); ?>
                                    </option>
                                <?php endforeach; ?>
                            </select>
                        </div>
                    </div>
                </div>
            </form>
        </div>
    </div>
</div>

```

```

<h6 class="text-hollywood mb-3 mt-4">Employee Details</h6>
<div class="row">
<div class="col-md-6 mb-3">
    <label for="create_first_name" class="form-label">First Name *</label>
    <input type="text" class="form-control" name="first_name" id="create_first_name"
required>
</div>
<div class="col-md-6 mb-3">
    <label for="create_last_name" class="form-label">Last Name *</label>
    <input type="text" class="form-control" name="last_name" id="create_last_name"
required>
</div>
</div>
<div class="row">
<div class="col-md-6 mb-3">
    <label for="create_position" class="form-label">Position *</label>
    <input type="text" class="form-control" name="position" id="create_position" required>
</div>
<div class="col-md-6 mb-3">
    <label for="create_department" class="form-label">Department *</label>
    <select class="form-select" name="department" id="create_department" required>
        <option value="">Select Department</option>
        <option value="Sales">Sales</option>
        <option value="Operations">Operations</option>
        <option value="Content">Content</option>
        <option value="Administration">Administration</option>
    </select>
</div>
</div>
<div class="row">
<div class="col-md-6 mb-3">
    <label for="create_hire_date" class="form-label">Hire Date *</label>
    <input type="date" class="form-control" name="hire_date" id="create_hire_date"
required>
</div>
<div class="col-md-6 mb-3">
    <label for="create_salary" class="form-label">Salary ($) *</label>
    <input type="number" class="form-control" name="salary" id="create_salary"
step="0.01" min="0" required>
</div>
</div>
</div>
<div class="modal-footer">

```

```

<button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Cancel</button>
<button type="submit" name="create_employee" class="btn btn-hollywood">
<i class="fas fa-user-plus"></i> Create Employee
</button>
</div>
</form>
</div>
</div>
</div>


<div class="modal fade" id="editEmployeeModal" tabindex="-1">
<div class="modal-dialog modal-lg">
<div class="modal-content">
<form method="POST" action="">
<input type="hidden" name="csrf_token" value="<?php echo generateCSRFToken(); ?>">
<input type="hidden" name="employee_id" id="edit_employee_id">

<div class="modal-header bg-hollywood text-white">
<h5 class="modal-title">Edit Employee</h5>
<button type="button" class="btn-close btn-close-white" data-bs-dismiss="modal"></button>
</div>
<div class="modal-body">
<div class="row">
<div class="col-md-6 mb-3">
<label for="edit_first_name" class="form-label">First Name *</label>
<input type="text" class="form-control" name="first_name" id="edit_first_name"
required>
</div>
<div class="col-md-6 mb-3">
<label for="edit_last_name" class="form-label">Last Name *</label>
<input type="text" class="form-control" name="last_name" id="edit_last_name"
required>
</div>
</div>
<div class="row">
<div class="col-md-6 mb-3">
<label for="edit_position" class="form-label">Position *</label>
<input type="text" class="form-control" name="position" id="edit_position" required>
</div>
<div class="col-md-6 mb-3">
<label for="edit_department" class="form-label">Department *</label>
<select class="form-select" name="department" id="edit_department" required>
<option value="">Select Department</option>

```

```

        <option value="Sales">Sales</option>
        <option value="Operations">Operations</option>
        <option value="Content">Content</option>
        <option value="Administration">Administration</option>
        </select>
    </div>
    </div>
    <div class="mb-3">
        <label for="edit_salary" class="form-label">Salary ($) *</label>
        <input type="number" class="form-control" name="salary" id="edit_salary"
            step="0.01" min="0" required>
    </div>
    <div class="mb-3 form-check">
        <input type="checkbox" class="form-check-input" name="is_active"
            id="edit_is_active" value="1">
        <label class="form-check-label" for="edit_is_active">
            Active Employee
        </label>
    </div>
    </div>
    <div class="modal-footer">
        <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Cancel</button>
        <button type="submit" name="update_employee" class="btn btn-hollywood">
            <i class="fas fa-save"></i> Update Employee
        </button>
    </div>
</form>
</div>
</div>
</div>

<script>
function editEmployee(employee) {
    document.getElementById('edit_employee_id').value = employee.employee_id;
    document.getElementById('edit_first_name').value = employee.first_name;
    document.getElementById('edit_last_name').value = employee.last_name;
    document.getElementById('edit_position').value = employee.position;
    document.getElementById('edit_department').value = employee.department;
    document.getElementById('edit_salary').value = employee.salary;
    document.getElementById('edit_is_active').checked = employee.is_active == 1;

    const modal = new bootstrap.Modal(document.getElementById('editEmployeeModal'));
    modal.show();
}

```

```

</script>

<?php include '../includes/footer.php'; ?>

Products.php
<?php

require_once '../../config/config.php';
require_once '../../includes/session.php';

requireRole(ROLE_MANAGER);

$page_title = 'Product Management';

$database = new Database();
$db = $database->getConnection();

require_once MODEL_PATH . 'Product.php';
require_once MODEL_PATH . 'Category.php';

$product_model = new Product($db);
$category_model = new Category($db);

if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['create_product'])) {
    if (!verifyCSRFToken($_POST['csrf_token'] ?? '')) {
        setFlashMessage('error', 'Invalid security token.');
    } else {
        $name = trim($_POST['name']);
        $description = trim($_POST['description']);
        $category_id = (int)$_POST['category_id'];
        $price = (float)$_POST['price'];
        $cost = (float)$_POST['cost'];
        $sku = trim($_POST['sku']);
        $quantity = (int)$_POST['quantity'];
        $reorder_level = (int)$_POST['reorder_level'];

        try {
            $db->beginTransaction();

            $product_id = $product_model->create($name, $description, $category_id, $price, $cost, $sku);

            if ($product_id) {
                $inv_query = "INSERT INTO inventory (product_id, quantity, reorder_level) VALUES (:pid, :qty, :rol)";

```

```

$inv_stmt = $db->prepare($inv_query);
$inv_stmt->execute([
    ':pid' => $product_id,
    ':qty' => $quantity,
    ':rol' => $reorder_level
]);
$db->commit();
logAudit('CREATE_PRODUCT', 'products', $product_id, null, $name);
setFlashMessage('success', 'Product created successfully!');

} else {
    $db->rollBack();
    setFlashMessage('error', 'Failed to create product.');
}
} catch (Exception $e) {
    $db->rollBack();
    setFlashMessage('error', 'Error: ' . $e->getMessage());
}

redirect('public/admin/products.php');
}

}

if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['update_product'])) {
    if (!verifyCSRFToken($_POST['csrf_token'] ?? '')) {
        setFlashMessage('error', 'Invalid security token.');
    } else {
        $product_id = (int)$_POST['product_id'];
        $name = trim($_POST['name']);
        $description = trim($_POST['description']);
        $category_id = (int)$_POST['category_id'];
        $price = (float)$_POST['price'];
        $cost = (float)$_POST['cost'];
        $sku = trim($_POST['sku']);
        $is_active = isset($_POST['is_active']) ? 1 : 0;

        if ($product_model->update($product_id, $name, $description, $category_id, $price, $cost, $sku,
$is_active)) {
            logAudit('UPDATE_PRODUCT', 'products', $product_id, null, $name);
            setFlashMessage('success', 'Product updated successfully!');
        } else {
            setFlashMessage('error', 'Failed to update product.');
        }
        redirect('public/admin/products.php');
    }
}

```

```

        }

}

if($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['delete_product'])) {
    if (!verifyCSRFToken($_POST['csrf_token'] ?? '')) {
        setFlashMessage('error', 'Invalid security token.');
    } else {
        $product_id = (int)$_POST['product_id'];

        if ($product_model->delete($product_id)) {
            logAudit('DELETE_PRODUCT', 'products', $product_id);
            setFlashMessage('success', 'Product deleted successfully!');
        } else {
            setFlashMessage('error', 'Failed to delete product. Product may be in orders.');
        }
        redirect('public/admin/products.php');
    }
}

$products_query =
    "SELECT p.*, c.name as category_name, i.quantity, i.reorder_level
     FROM products p
     LEFT JOIN categories c ON p.category_id = c.category_id
     LEFT JOIN inventory i ON p.product_id = i.product_id
     ORDER BY p.created_at DESC
";
$products_stmt = $db->prepare($products_query);
$products_stmt->execute();
$products = $products_stmt->fetchAll();

$categories = $category_model->getAll();

include '../includes/header.php';
?>

<div class="container-fluid my-4">
    <div class="d-flex justify-content-between align-items-center mb-4">
        <h1 class="text-hollywood">δÝ“ Product Management</h1>
        <button class="btn btn-hollywood" data-bs-toggle="modal" data-bs-target="#createProductModal">
            <i class="fas fa-plus"></i> Add Product
        </button>
    </div>

    <!-- Products Table -->

```

```

<div class="card card-hollywood">
<div class="card-header card-header-hollywood">
<i class="fas fa-box"></i> Products (<?php echo count($products); ?>
</div>
<div class="card-body">
<div class="table-responsive">
    <table class="table table-hover">
        <thead class="table-light">
            <tr>
                <th>SKU</th>
                <th>Name</th>
                <th>Category</th>
                <th>Price</th>
                <th>Cost</th>
                <th>Stock</th>
                <th>Status</th>
                <th>Actions</th>
            </tr>
        </thead>
        <tbody>
            <?php foreach ($products as $product): ?>
            <tr>
                <td><code><?php echo h($product['sku']); ?></code></td>
                <td><strong><?php echo h($product['name']); ?></strong></td>
                <td><?php echo h($product['category_name']); ?></td>
                <td><?php echo formatCurrency($product['price']); ?></td>
                <td><?php echo formatCurrency($product['cost']); ?></td>
                <td>
                    <?php echo $product['quantity']; ?>
                    <?php if ($product['quantity'] <= $product['reorder_level']): ?>
                        <i class="fas fa-exclamation-triangle text-warning"
                            title="Low stock"></i>
                    <?php endif; ?>
                </td>
                <td>
                    <?php if ($product['is_active']): ?>
                        <span class="badge bg-success">Active</span>
                    <?php else: ?>
                        <span class="badge bg-secondary">Inactive</span>
                    <?php endif; ?>
                </td>
                <td>
                    <button class="btn btn-sm btn-outline-hollywood">

```

```

        onclick="editProduct(<?php echo htmlspecialchars(json_encode($product));
?>)">
            <i class="fas fa-edit"></i>
        </button>
        <button class="btn btn-sm btn-outline-danger"
            onclick="deleteProduct(<?php echo $product['product_id']; ?>, '<?php echo
h($product['name']); ?>')">
            <i class="fas fa-trash"></i>
        </button>
    </td>
</tr>
<?php endforeach; ?>
</tbody>
</table>
</div>
</div>
</div>
</div>
<!-- Create Product Modal -->
<div class="modal fade" id="createProductModal" tabindex="-1">
    <div class="modal-dialog modal-lg">
        <div class="modal-content">
            <form method="POST" action="">
                <input type="hidden" name="csrf_token" value="<?php echo generateCSRFToken(); ?>">

                <div class="modal-header bg-hollywood text-white">
                    <h5 class="modal-title">Add New Product</h5>
                    <button type="button" class="btn-close btn-close-white" data-bs-dismiss="modal"></button>
                </div>
                <div class="modal-body">
                    <div class="row">
                        <div class="col-md-6 mb-3">
                            <label for="create_name" class="form-label">Product Name *</label>
                            <input type="text" class="form-control" name="name" id="create_name" required>
                        </div>
                        <div class="col-md-6 mb-3">
                            <label for="create_sku" class="form-label">SKU *</label>
                            <input type="text" class="form-control" name="sku" id="create_sku" required>
                        </div>
                    </div>
                    <div class="mb-3">
                        <label for="create_description" class="form-label">Description *</label>
                        <textarea class="form-control" name="description" id="create_description">
                    </div>
                </div>
            </form>
        </div>
    </div>

```

```

        rows="3" required></textarea>
</div>
<div class="row">
<div class="col-md-6 mb-3">
    <label for="create_category_id" class="form-label">Category *</label>
    <select class="form-select" name="category_id" id="create_category_id" required>
        <option value="">Select Category</option>
        <?php foreach ($categories as $cat): ?>
        <option value="<?php echo $cat['category_id']; ?>">
            <?php echo h($cat['name']); ?>
        </option>
        <?php endforeach; ?>
    </select>
</div>
<div class="col-md-3 mb-3">
    <label for="create_price" class="form-label">Price ($) *</label>
    <input type="number" class="form-control" name="price" id="create_price"
        step="0.01" min="0" required>
</div>
<div class="col-md-3 mb-3">
    <label for="create_cost" class="form-label">Cost ($) *</label>
    <input type="number" class="form-control" name="cost" id="create_cost"
        step="0.01" min="0" required>
</div>
</div>
<div class="row">
<div class="col-md-6 mb-3">
    <label for="create_quantity" class="form-label">Initial Stock *</label>
    <input type="number" class="form-control" name="quantity" id="create_quantity"
        min="0" required>
</div>
<div class="col-md-6 mb-3">
    <label for="create_reorder_level" class="form-label">Reorder Level *</label>
    <input type="number" class="form-control" name="reorder_level"
        id="create_reorder_level" min="0" required>
</div>
</div>
</div>
<div class="modal-footer">
    <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Cancel</button>
    <button type="submit" name="create_product" class="btn btn-hollywood">
        <i class="fas fa-save"></i> Create Product
    </button>
</div>

```

```

</form>
</div>
</div>
</div>


<div class="modal fade" id="editProductModal" tabindex="-1">
    <div class="modal-dialog modal-lg">
        <div class="modal-content">
            <form method="POST" action="">
                <input type="hidden" name="csrf_token" value="<?php echo generateCSRFToken(); ?>">
                <input type="hidden" name="product_id" id="edit_product_id">

                <div class="modal-header bg-hollywood text-white">
                    <h5 class="modal-title">Edit Product</h5>
                    <button type="button" class="btn-close btn-close-white" data-bs-dismiss="modal"></button>
                </div>
                <div class="modal-body">
                    <div class="row">
                        <div class="col-md-6 mb-3">
                            <label for="edit_name" class="form-label">Product Name *</label>
                            <input type="text" class="form-control" name="name" id="edit_name" required>
                        </div>
                        <div class="col-md-6 mb-3">
                            <label for="edit_sku" class="form-label">SKU *</label>
                            <input type="text" class="form-control" name="sku" id="edit_sku" required>
                        </div>
                    </div>
                    <div class="mb-3">
                        <label for="edit_description" class="form-label">Description *</label>
                        <textarea class="form-control" name="description" id="edit_description" rows="3" required></textarea>
                    </div>
                    <div class="row">
                        <div class="col-md-6 mb-3">
                            <label for="edit_category_id" class="form-label">Category *</label>
                            <select class="form-select" name="category_id" id="edit_category_id" required>
                                <option value="">Select Category</option>
                                <?php foreach ($categories as $cat): ?>
                                    <option value="<?php echo $cat['category_id']; ?>">
                                        <?php echo h($cat['name']); ?>
                                </option>
                            <?php endforeach; ?>
                            </select>
                        </div>
                    </div>
                </div>
            </form>
        </div>
    </div>
</div>

```

```

</div>
<div class="col-md-3 mb-3">
    <label for="edit_price" class="form-label">Price ($) *</label>
    <input type="number" class="form-control" name="price" id="edit_price"
        step="0.01" min="0" required>
</div>
<div class="col-md-3 mb-3">
    <label for="edit_cost" class="form-label">Cost ($) *</label>
    <input type="number" class="form-control" name="cost" id="edit_cost"
        step="0.01" min="0" required>
</div>
<div class="mb-3 form-check">
<input type="checkbox" class="form-check-input" name="is_active"
    id="edit_is_active" value="1">
<label class="form-check-label" for="edit_is_active">
    Active (visible to customers)
</label>
</div>
</div>
<div class="modal-footer">
    <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Cancel</button>
    <button type="submit" name="update_product" class="btn btn-hollywood">
        <i class="fas fa-save"></i> Update Product
    </button>
</div>
</form>
</div>
</div>
</div>

```

```


<div class="modal fade" id="deleteProductModal" tabindex="-1">
    <div class="modal-dialog">
        <div class="modal-content">
            <form method="POST" action="">
                <input type="hidden" name="csrf_token" value="<?php echo generateCSRFToken(); ?>">
                <input type="hidden" name="product_id" id="delete_product_id">

                <div class="modal-header bg-danger text-white">
                    <h5 class="modal-title">Confirm Deletion</h5>
                    <button type="button" class="btn-close btn-close-white" data-bs-dismiss="modal"></button>
                </div>
                <div class="modal-body">

```

```

<p>Are you sure you want to delete this product?</p>
<p class="text-muted" id="delete_product_name"></p>
<p class="text-danger"><strong>This action cannot be undone!</strong></p>
</div>
<div class="modal-footer">
<button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Cancel</button>
<button type="submit" name="delete_product" class="btn btn-danger">
<i class="fas fa-trash"></i> Delete Product
</button>
</div>
</form>
</div>
</div>
</div>

<script>
function editProduct(product) {
    document.getElementById('edit_product_id').value = product.product_id;
    document.getElementById('edit_name').value = product.name;
    document.getElementById('edit_sku').value = product.sku;
    document.getElementById('edit_description').value = product.description;
    document.getElementById('edit_category_id').value = product.category_id;
    document.getElementById('edit_price').value = product.price;
    document.getElementById('edit_cost').value = product.cost;
    document.getElementById('edit_is_active').checked = product.is_active == 1;

    const modal = new bootstrap.Modal(document.getElementById('editProductModal'));
    modal.show();
}

function deleteProduct(productId, name) {
    document.getElementById('delete_product_id').value = productId;
    document.getElementById('delete_product_name').textContent = name;

    const modal = new bootstrap.Modal(document.getElementById('deleteProductModal'));
    modal.show();
}
</script>

```

<?php include '../includes/footer.php'; ?>

Reports.php

<?php

```

require_once '../config/config.php';
require_once '../includes/session.php';

requireRole(ROLE_MANAGER);

$page_title = 'Product Management';

$database = new Database();
$db = $database->getConnection();

require_once MODEL_PATH . 'Product.php';
require_once MODEL_PATH . 'Category.php';

$product_model = new Product($db);
$category_model = new Category($db);

if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['create_product'])) {
    if (!verifyCSRFToken($_POST['csrf_token'] ?? '')) {
        setFlashMessage('error', 'Invalid security token.');
    } else {
        $name = trim($_POST['name']);
        $description = trim($_POST['description']);
        $category_id = (int)$_POST['category_id'];
        $price = (float)$_POST['price'];
        $cost = (float)$_POST['cost'];
        $sku = trim($_POST['sku']);
        $quantity = (int)$_POST['quantity'];
        $reorder_level = (int)$_POST['reorder_level'];

        try {
            $db->beginTransaction();

            $product_id = $product_model->create($name, $description, $category_id, $price, $cost, $sku);

            if ($product_id) {
                $inv_query = "INSERT INTO inventory (product_id, quantity, reorder_level) VALUES (:pid, :qty, :rol)";
                $inv_stmt = $db->prepare($inv_query);
                $inv_stmt->execute([
                    ':pid' => $product_id,
                    ':qty' => $quantity,
                    ':rol' => $reorder_level
                ]);
            }
        }
    }
}

```

```

        $db->commit();
        logAudit('CREATE_PRODUCT', 'products', $product_id, null, $name);
        setFlashMessage('success', 'Product created successfully!');
    } else {
        $db->rollBack();
        setFlashMessage('error', 'Failed to create product.');
    }
}

} catch (Exception $e) {
    $db->rollBack();
    setFlashMessage('error', 'Error: ' . $e->getMessage());
}

redirect('public/admin/products.php');
}

}

if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['update_product'])) {
    if (!verifyCSRFToken($_POST['csrf_token'] ?? '')) {
        setFlashMessage('error', 'Invalid security token.');
    } else {
        $product_id = (int)$_POST['product_id'];
        $name = trim($_POST['name']);
        $description = trim($_POST['description']);
        $category_id = (int)$_POST['category_id'];
        $price = (float)$_POST['price'];
        $cost = (float)$_POST['cost'];
        $sku = trim($_POST['sku']);
        $is_active = isset($_POST['is_active']) ? 1 : 0;

        if ($product_model->update($product_id, $name, $description, $category_id, $price, $cost, $sku,
$is_active)) {
            logAudit('UPDATE_PRODUCT', 'products', $product_id, null, $name);
            setFlashMessage('success', 'Product updated successfully!');
        } else {
            setFlashMessage('error', 'Failed to update product.');
        }
        redirect('public/admin/products.php');
    }
}

if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['delete_product'])) {
    if (!verifyCSRFToken($_POST['csrf_token'] ?? '')) {
        setFlashMessage('error', 'Invalid security token.');
    } else {

```

```

$product_id = (int)$_POST['product_id'];

if ($product_model->delete($product_id)) {
    logAudit('DELETE_PRODUCT', 'products', $product_id);
    setFlashMessage('success', 'Product deleted successfully!');
} else {
    setFlashMessage('error', 'Failed to delete product. Product may be in orders.');
}
redirect('public/admin/products.php');
}

$products_query =
    "SELECT p.*, c.name as category_name, i.quantity, i.reorder_level
     FROM products p
     LEFT JOIN categories c ON p.category_id = c.category_id
     LEFT JOIN inventory i ON p.product_id = i.product_id
     ORDER BY p.created_at DESC
";
$products_stmt = $db->prepare($products_query);
$products_stmt->execute();
$products = $products_stmt->fetchAll();

$categories = $category_model->getAll();

include '../includes/header.php';
?>

<div class="container-fluid my-4">
    <div class="d-flex justify-content-between align-items-center mb-4">
        <h1 class="text-hollywood">Product Management</h1>
        <button class="btn btn-hollywood" data-bs-toggle="modal" data-bs-target="#createProductModal">
            <i class="fas fa-plus"></i> Add Product
        </button>
    </div>

    <!-- Products Table -->
    <div class="card card-hollywood">
        <div class="card-header card-header-hollywood">
            <i class="fas fa-box"></i> Products (<?php echo count($products); ?>)
        </div>
        <div class="card-body">
            <div class="table-responsive">
                <table class="table table-hover">

```

```

<thead class="table-light">
<tr>
    <th>SKU</th>
    <th>Name</th>
    <th>Category</th>
    <th>Price</th>
    <th>Cost</th>
    <th>Stock</th>
    <th>Status</th>
    <th>Actions</th>
</tr>
</thead>
<tbody>
<?php foreach ($products as $product): ?>
    <tr>
        <td><code><?php echo h($product['sku']); ?></code></td>
        <td><strong><?php echo h($product['name']); ?></strong></td>
        <td><?php echo h($product['category_name']); ?></td>
        <td><?php echo formatCurrency($product['price']); ?></td>
        <td><?php echo formatCurrency($product['cost']); ?></td>
        <td>
            <?php echo $product['quantity']; ?>
            <?php if ($product['quantity'] <= $product['reorder_level']): ?>
                <i class="fas fa-exclamation-triangle text-warning" title="Low stock"></i>
            <?php endif; ?>
        </td>
        <td>
            <?php if ($product['is_active']): ?>
                <span class="badge bg-success">Active</span>
            <?php else: ?>
                <span class="badge bg-secondary">Inactive</span>
            <?php endif; ?>
        </td>
        <td>
            <button class="btn btn-sm btn-outline-hollywood"
                   onclick="editProduct(<?php echo htmlspecialchars(json_encode($product)); ?>)">
                <i class="fas fa-edit"></i>
            </button>
            <button class="btn btn-sm btn-outline-danger"
                   onclick="deleteProduct(<?php echo $product['product_id']; ?>, '<?php echo h($product['name']); ?>')">
                <i class="fas fa-trash"></i>
            </button>
        </td>
    </tr>
</tbody>

```

```

        </button>
    </td>
    </tr>
<?php endforeach; ?>
</tbody>
</table>
</div>
</div>
</div>
</div>


<div class="modal fade" id="createProductModal" tabindex="-1">
    <div class="modal-dialog modal-lg">
        <div class="modal-content">
            <form method="POST" action="">
                <input type="hidden" name="csrf_token" value="<?php echo generateCSRFToken(); ?>">

                <div class="modal-header bg-hollywood text-white">
                    <h5 class="modal-title">Add New Product</h5>
                    <button type="button" class="btn-close btn-close-white" data-bs-dismiss="modal"></button>
                </div>
                <div class="modal-body">
                    <div class="row">
                        <div class="col-md-6 mb-3">
                            <label for="create_name" class="form-label">Product Name *</label>
                            <input type="text" class="form-control" name="name" id="create_name" required>
                        </div>
                        <div class="col-md-6 mb-3">
                            <label for="create_sku" class="form-label">SKU *</label>
                            <input type="text" class="form-control" name="sku" id="create_sku" required>
                        </div>
                    </div>
                    <div class="mb-3">
                        <label for="create_description" class="form-label">Description *</label>
                        <textarea class="form-control" name="description" id="create_description" rows="3" required></textarea>
                    </div>
                    <div class="row">
                        <div class="col-md-6 mb-3">
                            <label for="create_category_id" class="form-label">Category *</label>
                            <select class="form-select" name="category_id" id="create_category_id" required>
                                <option value="">Select Category</option>
                                <?php foreach ($categories as $cat): ?>

```

```

        <option value="<?php echo $cat['category_id']; ?>">
            <?php echo h($cat['name']); ?>
        </option>
    <?php endforeach; ?>
</select>
</div>
<div class="col-md-3 mb-3">
    <label for="create_price" class="form-label">Price ($) *</label>
    <input type="number" class="form-control" name="price" id="create_price"
        step="0.01" min="0" required>
</div>
<div class="col-md-3 mb-3">
    <label for="create_cost" class="form-label">Cost ($) *</label>
    <input type="number" class="form-control" name="cost" id="create_cost"
        step="0.01" min="0" required>
</div>
</div>
<div class="row">
<div class="col-md-6 mb-3">
    <label for="create_quantity" class="form-label">Initial Stock *</label>
    <input type="number" class="form-control" name="quantity" id="create_quantity"
        min="0" required>
</div>
<div class="col-md-6 mb-3">
    <label for="create_reorder_level" class="form-label">Reorder Level *</label>
    <input type="number" class="form-control" name="reorder_level"
        id="create_reorder_level" min="0" required>
</div>
</div>
<div class="modal-footer">
    <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Cancel</button>
    <button type="submit" name="create_product" class="btn btn-hollywood">
        <i class="fas fa-save"></i> Create Product
    </button>
</div>
</form>
</div>
</div>
</div>
<!-- Edit Product Modal -->
<div class="modal fade" id="editProductModal" tabindex="-1">
    <div class="modal-dialog modal-lg">

```

```

<div class="modal-content">
<form method="POST" action="">
    <input type="hidden" name="csrf_token" value="<?php echo generateCSRFToken(); ?>">
    <input type="hidden" name="product_id" id="edit_product_id">

    <div class="modal-header bg-hollywood text-white">
        <h5 class="modal-title">Edit Product</h5>
        <button type="button" class="btn-close btn-close-white" data-bs-dismiss="modal"></button>
    </div>
    <div class="modal-body">
        <div class="row">
            <div class="col-md-6 mb-3">
                <label for="edit_name" class="form-label">Product Name *</label>
                <input type="text" class="form-control" name="name" id="edit_name" required>
            </div>
            <div class="col-md-6 mb-3">
                <label for="edit_sku" class="form-label">SKU *</label>
                <input type="text" class="form-control" name="sku" id="edit_sku" required>
            </div>
        </div>
        <div class="mb-3">
            <label for="edit_description" class="form-label">Description *</label>
            <textarea class="form-control" name="description" id="edit_description" rows="3" required></textarea>
        </div>
        <div class="row">
            <div class="col-md-6 mb-3">
                <label for="edit_category_id" class="form-label">Category *</label>
                <select class="form-select" name="category_id" id="edit_category_id" required>
                    <option value="">Select Category</option>
                    <?php foreach ($categories as $cat): ?>
                    <option value="<?php echo $cat['category_id']; ?>">
                        <?php echo h($cat['name']); ?>
                    </option>
                <?php endforeach; ?>
                </select>
            </div>
            <div class="col-md-3 mb-3">
                <label for="edit_price" class="form-label">Price ($) *</label>
                <input type="number" class="form-control" name="price" id="edit_price" step="0.01" min="0" required>
            </div>
            <div class="col-md-3 mb-3">
                <label for="edit_cost" class="form-label">Cost ($) *</label>

```

```

        <input type="number" class="form-control" name="cost" id="edit_cost"
        step="0.01" min="0" required>
    </div>
    </div>
    <div class="mb-3 form-check">
        <input type="checkbox" class="form-check-input" name="is_active"
            id="edit_is_active" value="1">
        <label class="form-check-label" for="edit_is_active">
            Active (visible to customers)
        </label>
    </div>
    </div>
    <div class="modal-footer">
        <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Cancel</button>
        <button type="submit" name="update_product" class="btn btn-hollywood">
            <i class="fas fa-save"></i> Update Product
        </button>
    </div>
</form>
</div>
</div>
</div>


<div class="modal fade" id="deleteProductModal" tabindex="-1">
    <div class="modal-dialog">
        <div class="modal-content">
            <form method="POST" action="">
                <input type="hidden" name="csrf_token" value="<?php echo generateCSRFToken(); ?>">
                <input type="hidden" name="product_id" id="delete_product_id">

                <div class="modal-header bg-danger text-white">
                    <h5 class="modal-title">Confirm Deletion</h5>
                    <button type="button" class="btn-close btn-close-white" data-bs-dismiss="modal"></button>
                </div>
                <div class="modal-body">
                    <p>Are you sure you want to delete this product?</p>
                    <p class="text-muted" id="delete_product_name"></p>
                    <p class="text-danger"><strong>This action cannot be undone!</strong></p>
                </div>
                <div class="modal-footer">
                    <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Cancel</button>
                    <button type="submit" name="delete_product" class="btn btn-danger">
                        <i class="fas fa-trash"></i> Delete Product
                    </button>
                </div>
            </form>
        </div>
    </div>
</div>

```

```

        </button>
    </div>
</form>
</div>
</div>
</div>

<script>
function editProduct(product) {
    document.getElementById('edit_product_id').value = product.product_id;
    document.getElementById('edit_name').value = product.name;
    document.getElementById('edit_sku').value = product.sku;
    document.getElementById('edit_description').value = product.description;
    document.getElementById('edit_category_id').value = product.category_id;
    document.getElementById('edit_price').value = product.price;
    document.getElementById('edit_cost').value = product.cost;
    document.getElementById('edit_is_active').checked = product.is_active == 1;

    const modal = new bootstrap.Modal(document.getElementById('editProductModal'));
    modal.show();
}

function deleteProduct(productId, name) {
    document.getElementById('delete_product_id').value = productId;
    document.getElementById('delete_product_name').textContent = name;

    const modal = new bootstrap.Modal(document.getElementById('deleteProductModal'));
    modal.show();
}
</script>

```

<?php include '../includes/footer.php'; ?>

Article.php

```

<?php

require_once '../../config/config.php';
require_once '../../includes/session.php';

```

```

$database = new Database();
$db = $database->getConnection();

```

```

require_once MODEL_PATH . 'Article.php';

```

```

$article_model = new Article($db);

$slug = $_GET['slug'] ?? '';

if (empty($slug)) {
    setFlashMessage('error', 'Article not found.');
    redirect('public/customer/hollywood101.php');
}

if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['add_comment'])) {
    if (!isLoggedIn()) {
        $_SESSION['return_url'] = $_SERVER['REQUEST_URI'];
        setFlashMessage('error', 'Please login to leave a comment.');
        redirect('public/login.php');
    }

    if (!verifyCSRFToken($_POST['csrf_token'] ?? '')) {
        setFlashMessage('error', 'Invalid security token.');
    } else {
        $comment_text = trim($_POST['comment_text'] ?? '');
    }

    if (empty($comment_text)) {
        setFlashMessage('error', 'Comment cannot be empty.');
    } elseif (strlen($comment_text) < 10) {
        setFlashMessage('error', 'Comment must be at least 10 characters.');
    } else {
        require_once MODEL_PATH . 'Customer.php';
        $customer_model = new Customer($db);
        $customer = $customer_model->getById(getCurrentUserId());

        if (!$customer) {
            setFlashMessage('error', 'Customer account not found.');
            redirect('public/customer/article.php?slug=' . $slug);
        }

        $article = $article_model->getBySlug($slug);
        if ($article_model->addComment($article['article_id'], $customer['customer_id'], $comment_text)) {
            logAudit('ADD_COMMENT', 'article_comments', $article['article_id']);
            setFlashMessage('success', 'Comment posted successfully!');
            redirect('public/customer/article.php?slug=' . $slug);
        } else {
            setFlashMessage('error', 'Failed to post comment.');
        }
    }
}

```

```

        }
    }
}

$article = $article_model->getBySlug($slug);

if (!$article) {
    setFlashMessage('error', 'Article not found.');
    redirect('public/customer/hollywood101.php');
}

$article_model->incrementViews($article['article_id']);

$comments = $article_model->getComments($article['article_id']);

$page_title = $article['title'];

include '../includes/header.php';
?>

<div class="container my-4">
    <nav aria-label="breadcrumb">
        <ol class="breadcrumb">
            <li class="breadcrumb-item"><a href="hollywood101.php">Hollywood 101</a></li>
            <li class="breadcrumb-item"><a href="hollywood101.php?category=<?php echo urlencode($article['category']); ?>">
                <?php echo h($article['category']); ?>
            </a></li>
            <li class="breadcrumb-item active"><?php echo h($article['title']); ?></li>
        </ol>
    </nav>

    <div class="row">
        <div class="col-md-8 mx-auto">
            <!-- Article Header -->
            <div class="card card-hollywood mb-4">
                <?php if ($article['featured_image']): ?>
                    
                    class="card-img-top" alt="<?php echo h($article['title']); ?>">
                    style="max-height: 400px; object-fit: cover;">
                <?php endif; ?>

                <div class="card-body">
                    <div class="mb-3">

```

```

<span class="badge badge-hollywood">
    <?php echo h($article['category']); ?>
</span>
<span class="text-muted ms-2">
    <i class="fas fa-eye"></i> <?php echo $article['view_count']; ?> views
</span>
</div>

<h1 class="display-5 mb-3"><?php echo h($article['title']); ?></h1>

<div class="d-flex align-items-center mb-4 text-muted">
<div class="me-4">
    <i class="fas fa-user"></i>
    By <?php echo h($article['first_name'] . ' ' . $article['last_name']); ?>
</div>
<div>
    <i class="fas fa-calendar"></i>
    <?php echo formatDateTime($article['published_at']); ?>
</div>
</div>

<hr>

<div class="article-content">
<?php echo $article['content']; ?>
</div>

<hr class="my-4">

<div class="d-flex justify-content-between align-items-center">
<a href="hollywood101.php" class="btn btn-outline-secondary">
    <i class="fas fa-arrow-left"></i> Back to Articles
</a>
<div>
    <button onclick="window.print()" class="btn btn-outline-hollywood">
        <i class="fas fa-print"></i> Print
    </button>
</div>
</div>
</div>
</div>

<!-- Comments Section -->
<div class="card card-hollywood mb-4">

```

```

<div class="card-header card-header-hollywood">
<i class="fas fa-comments"></i> Comments (<?php echo count($comments); ?>
</div>
<div class="card-body">
<!-- Add Comment Form -->
<div class="mb-4">
<h5>Leave a Comment</h5>
<form method="POST" action="">
    <input type="hidden" name="csrf_token" value="<?php echo generateCSRFToken(); ?>">
    <div class="mb-3">
        <textarea class="form-control" name="comment_text" rows="4"
            placeholder="Share your thoughts..." required minlength="10"></textarea>
        <small class="text-muted">Minimum 10 characters</small>
    </div>
    <button type="submit" name="add_comment" class="btn btn-hollywood">
        <i class="fas fa-paper-plane"></i> Post Comment
    </button>
</form>
</div>

<hr>

<!-- Display Comments -->
<?php if (empty($comments)): ?>
<div class="text-center text-muted py-4">
    <i class="fas fa-comments fa-3x mb-3"></i>
    <p>No comments yet. Be the first to comment!</p>
</div>
<?php else: ?>
<div class="comments-list">
    <?php foreach ($comments as $comment): ?>
    <div class="comment-item mb-3 pb-3 border-bottom">
        <div class="d-flex align-items-start">
            <div class="flex-shrink-0">
                <div style="width: 50px; height: 50px; background: linear-gradient(135deg,
#d4af37, #c7a229); border-radius: 50%; display: flex; align-items: center; justify-content: center; color: white; font-weight: bold; font-size: 1.2rem;">
                    <?php echo strtoupper(substr($comment['first_name'], 0, 1)); ?>
                </div>
            </div>
            <div class="flex-grow-1 ms-3">
                <h6 class="mb-1">
                    <?php echo h($comment['first_name'] . ' ' . $comment['last_name']); ?>
                </h6>
            </div>
        </div>
    </div>
</div>

```

```

<small class="text-muted ms-2">
    <?php echo formatDate($comment['created_at']); ?>
</small>
</h6>
<p class="mb-0"><?php echo nl2br(h($comment['comment_text'])); ?></p>
</div>
</div>
</div>
<?php endforeach; ?>
</div>
<?php endif; ?>
</div>
</div>

<!-- Related Articles (placeholder) -->
<div class="card card-hollywood">
    <div class="card-header card-header-hollywood">
        <i class="fas fa-bookmark"></i> More Articles in <?php echo h($article['category']); ?>
    </div>
    <div class="card-body">
        <a href="hollywood101.php?category=<?php echo urlencode($article['category']); ?>">
            class="btn btn-outline-hollywood">
                Browse More Articles <i class="fas fa-arrow-right"></i>
            </a>
        </div>
    </div>
    </div>
    </div>
</div>

<style>
    .article-content h2 {
        color: #d4af37;
        margin-top: 1.5rem;
        margin-bottom: 1rem;
    }

    .article-content p {
        line-height: 1.8;
        margin-bottom: 1rem;
    }

    .article-content ul, .article-content ol {
        margin-bottom: 1rem;
    }

```

```

padding-left: 2rem;
}

@media print {
    .btn, nav, footer, .navbar, .comments-list, form { display: none !important; }
    .container { max-width: 100% !important; }
}

```

</style>

```

<?php include '../includes/footer.php'; ?>

Cart.php
<?php

require_once '../../config/config.php';
require_once '../../includes/session.php';

requireLoginForAction('view your cart');

$page_title = 'Shopping Cart';

$database = new Database();
$db = $database->getConnection();

require_once MODEL_PATH . 'Cart.php';

$cart = new Cart($db);
$customer_id = $_SESSION['customer_id'];

if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    if (!verifyCSRFToken($_POST['csrf_token'] ?? '')) {
        setFlashMessage('error', 'Invalid security token.');
        redirect('public/customer/cart.php');
    }

    if (isset($_POST['update_cart'])) {
        foreach ($_POST['quantity'] as $product_id => $quantity) {
            $cart->updateQuantity($customer_id, $product_id, (int)$quantity);
        }
        setFlashMessage('success', 'Cart updated successfully!');
        redirect('public/customer/cart.php');
    }
}

if (isset($_POST['remove_item'])) {

```

```

$product_id = (int)$_POST['product_id'];
if ($cart->removeItem($customer_id, $product_id)) {
    logAudit('REMOVE_FROM_CART', 'shopping_cart', $product_id);
    setFlashMessage('success', 'Item removed from cart.');
}
redirect('public/customer/cart.php');
}

if (isset($_POST['clear_cart'])) {
    if ($cart->clearCart($customer_id)) {
        setFlashMessage('info', 'Cart cleared.');
    }
    redirect('public/customer/cart.php');
}
}

$cart_items = $cart->getCartItems($customer_id);
$cart_total = $cart->getCartTotal($customer_id);

$validation_errors = $cart->validateCart($customer_id);

include '../includes/header.php';
?>

<div class="container my-4">
    <h1 class="text-hollywood mb-4">δŸ› Shopping Cart</h1>

    <?php if (!empty($validation_errors)): ?>
    <div class="alert alert-danger">
        <h5><i class="fas fa-exclamation-triangle"></i> Cart Validation Issues</h5>
        <ul class="mb-0">
            <?php foreach ($validation_errors as $error): ?>
            <li>
                <?php echo h($error['product_name']); ?>:
                Requested <?php echo $error['quantity']; ?>,
                but only <?php echo $error['quantity_available'] ?? 0; ?> available in stock.
            </li>
            <?php endforeach; ?>
        </ul>
    </div>
    <?php endif; ?>

    <?php if (empty($cart_items)): ?>
    <div class="row">

```

```

<div class="col-md-8 mx-auto">
    <div class="card card-hollywood text-center py-5">
        <div class="card-body">
            <i class="fas fa-shopping-cart fa-4x text-muted mb-3"></i>
            <h3>Your cart is empty</h3>
            <p class="text-muted">Start shopping and add some Hollywood magic to your cart!</p>
            <a href="shop.php" class="btn btn-hollywood btn-lg mt-3">
                <i class="fas fa-shopping-bag"></i> Start Shopping
            </a>
        </div>
    </div>
<?php else: ?>
<form method="POST" action="">
<input type="hidden" name="csrf_token" value="<?php echo generateCSRFToken(); ?>">

<div class="row">
    <div class="col-md-8">
        <div class="card card-hollywood">
            <div class="card-header card-header-hollywood">
                Cart Items (<?php echo count($cart_items); ?>)
            </div>
            <div class="card-body p-0">
                <?php foreach ($cart_items as $item): ?>
                <div class="cart-item">
                    <div class="row align-items-center">
                        <div class="col-md-2">
                            <div style="width: 80px; height: 80px; background: #f0f0f0; display: flex; align-items: center; justify-content: center; border-radius: 8px;">
                                <?php if ($item['image_url']): ?>
                                    ">
                                <?php else: ?>
                                    <span style="font-size: 2rem;">δΥΖ¬</span>
                                <?php endif; ?>
                            </div>
                        </div>
                        <div class="col-md-4">
                            <h6 class="mb-1">
                                <a href="product_detail.php?id=<?php echo $item['product_id']; ?>">
                                    <?php echo h($item['product_name']); ?>
                                </a>
                            </h6>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
</div>

```

```

        </h6>
        <small class="text-muted"><?php echo h($item['category_name']); ?></small>
    ?></small><br>
        <small class="text-muted">SKU: <?php echo h($item['sku']); ?></small>
    </div>
        <div class="col-md-2">
            <strong><?php echo formatCurrency($item['price']); ?></strong>
        </div>
        <div class="col-md-2">
            <input type="number" class="form-control"
                name="quantity[<?php echo $item['product_id']; ?>]"
                value="<?php echo $item['quantity']; ?>"
                min="1" max="<?php echo $item['quantity_available']; ?>"
                style="width: 80px;">
        </div>
        <div class="col-md-2 text-end">
            <strong class="text-hollywood">
                <?php echo formatCurrency($item['item_total']); ?>
            </strong>
            <form method="POST" action="" class="d-inline">
                <input type="hidden" name="csrf_token" value="<?php echo
generateCSRFToken(); ?>">
                <input type="hidden" name="product_id" value="<?php echo
$item['product_id']; ?>">
                <button type="submit" name="remove_item" class="btn btn-sm btn-link
text-danger"
                    onclick="return confirm('Remove this item from cart?')">
                    <i class="fas fa-trash"></i>
                </button>
            </form>
        </div>
    </div>
    <?php endforeach; ?>
</div>
<div class="card-footer">
    <div class="row">
        <div class="col-md-6">
            <button type="submit" name="update_cart" class="btn btn-outline-hollywood">
                <i class="fas fa-sync"></i> Update Cart
            </button>
            <button type="submit" name="clear_cart" class="btn btn-outline-danger"
                onclick="return confirm('Clear entire cart?')">
                <i class="fas fa-trash"></i> Clear Cart
            </button>
        </div>
    </div>
</div>

```

```

        </button>
    </div>
    <div class="col-md-6 text-end">
        <a href="shop.php" class="btn btn-outline-secondary">
            <i class="fas fa-arrow-left"></i> Continue Shopping
        </a>
    </div>
</div>
</div>
</div>

<div class="col-md-4">
    <div class="cart-summary">
        <h4 class="text-hollywood mb-3">Order Summary</h4>

        <div class="d-flex justify-content-between mb-2">
            <span>Subtotal:</span>
            <strong><?php echo formatCurrency($cart_total); ?></strong>
        </div>

        <div class="d-flex justify-content-between mb-2">
            <span>Tax (<?php echo (TAX_RATE * 100); ?>%):</span>
            <strong><?php echo formatCurrency($cart_total * TAX_RATE); ?></strong>
        </div>

        <div class="d-flex justify-content-between mb-2">
            <span>Shipping:</span>
            <strong>
                <?php
                    $shipping = $cart_total >= FREE_SHIPPING_THRESHOLD ? 0 :
                STANDARD_SHIPPING_COST;
                    echo formatCurrency($shipping);
                ?>
            </strong>
        </div>

        <?php if ($cart_total >= FREE_SHIPPING_THRESHOLD): ?>
            <div class="alert alert-success py-2 small">
                <i class="fas fa-truck"></i> Free shipping applied!
            </div>
        <?php else: ?>
            <div class="alert alert-info py-2 small">
                <i class="fas fa-info-circle"></i>

```

Add <?php echo formatCurrency(FREE_SHIPPING_THRESHOLD - \$cart_total); ?>
more for free shipping!

```
</div>
<?php endif; ?>

<hr>

<div class="d-flex justify-content-between mb-3">
    <strong>Total:</strong>
    <h4 class="text-hollywood mb-0">
        <?php
            $total = $cart_total + ($cart_total * TAX_RATE) + $shipping;
            echo formatCurrency($total);
        ?>
        </h4>
</div>

<?php if (empty($validation_errors)): ?>
    <a href="checkout.php" class="btn btn-hollywood btn-lg w-100">
        <i class="fas fa-lock"></i> Proceed to Checkout
    </a>
<?php else: ?>
    <button type="button" class="btn btn-secondary btn-lg w-100" disabled>
        <i class="fas fa-exclamation-triangle"></i> Fix Cart Issues
    </button>
<?php endif; ?>

<div class="text-center mt-3">
    <small class="text-muted">
        <i class="fas fa-lock"></i> Secure Checkout
    </small>
</div>
</div>
</div>
<?php endif; ?>
</div>

<?php include '../includes/footer.php'; ?>
```

Checkout.php

```
<?php
```

```

require_once '../config/config.php';
require_once '../includes/session.php';

requireLoginForAction('checkout');

$page_title = 'Checkout';

$database = new Database();
$db = $database->getConnection();

require_once MODEL_PATH . 'Cart.php';
require_once MODEL_PATH . 'Order.php';
require_once MODEL_PATH . 'Customer.php';

$cart = new Cart($db);
$order_model = new Order($db);
$customer_model = new Customer($db);

$customer_id = $_SESSION['customer_id'];

$cart_items = $cart->getCartItems($customer_id);
$cart_total = $cart->getCartTotal($customer_id);
$validation_errors = $cart->validateCart($customer_id);

if (empty($cart_items)) {
    setFlashMessage('warning', 'Your cart is empty.');
    redirect('public/customer/shop.php');
}

if (!empty($validation_errors)) {
    setFlashMessage('error', 'Please fix cart issues before checkout.');
    redirect('public/customer/cart.php');
}

$customer = $customer_model->getById($customer_id);

if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['place_order'])) {
    if (!verifyCSRFToken($_POST['csrf_token'] ?? '')) {
        setFlashMessage('error', 'Invalid security token.');
        redirect('public/customer/checkout.php');
    }

    $shipping_address = [
        'address_line1' => trim($_POST['address_line1'] ?? ''),

```

```

'address_line2' => trim($_POST['address_line2'] ?? ""),
'city' => trim($_POST['city'] ?? ""),
'state' => trim($_POST['state'] ?? ""),
'zip_code' => trim($_POST['zip_code'] ?? ""),
'country' => trim($_POST['country'] ?? 'USA')
];

$notes = trim($_POST['notes']);

$errors = [];
if (empty($shipping_address['address_line1'])) $errors[] = 'Address line 1 is required';
if (empty($shipping_address['city'])) $errors[] = 'City is required';
if (empty($shipping_address['state'])) $errors[] = 'State is required';
if (empty($shipping_address['zip_code'])) $errors[] = 'ZIP code is required';

if (empty($errors)) {
    $order_id = $order_model->createOrder($customer_id, $shipping_address, $notes);

    if ($order_id) {
        logAudit('PLACE_ORDER', 'orders', $order_id);
        setFlashMessage('success', 'Order placed successfully! Order #' . $order_id);
        redirect('public/customer/order_detail.php?id=' . $order_id);
    } else {
        setFlashMessage('error', 'Failed to create order. Please try again.');
    }
} else {
    setFlashMessage('error', implode(', ', $errors));
}
}

$subtotal = $cart_total;
$tax = $subtotal * TAX_RATE;
$shipping = $subtotal >= FREE_SHIPPING_THRESHOLD ? 0 : STANDARD_SHIPPING_COST;
$total = $subtotal + $tax + $shipping;

include '../includes/header.php';
?>

<div class="container my-4">
    <h1 class="text-hollywood mb-4">ðŸ”“ Secure Checkout</h1>

    <div class="row">
        <div class="col-md-8">
            <div class="card card-hollywood mb-4">

```

```

<div class="card-header card-header-hollywood">
<i class="fas fa-shopping-bag"></i> Order Items
</div>
<div class="card-body">
<?php foreach ($cart_items as $item): ?>
<div class="d-flex align-items-center mb-3 pb-3 border-bottom">
    <div style="width: 60px; height: 60px; background: #f0f0f0; border-radius: 5px; display:
flex; align-items: center; justify-content: center; margin-right: 15px;">
        &#8226;
    </div>
    <div class="flex-grow-1">
        <h6 class="mb-0"><?php echo h($item['product_name']); ?></h6>
        <small class="text-muted">Qty: <?php echo $item['quantity']; ?> — <?php echo
formatCurrency($item['price']); ?></small>
    </div>
    <div>
        <strong><?php echo formatCurrency($item['item_total']); ?></strong>
    </div>
    </div>
    <?php endforeach; ?>
</div>
</div>

<form method="POST" action="">
    <input type="hidden" name="csrf_token" value="<?php echo generateCSRFToken(); ?>">

    <div class="card card-hollywood mb-4">
        <div class="card-header card-header-hollywood">
            <i class="fas fa-shipping-fast"></i> Shipping Address
        </div>
        <div class="card-body">
            <div class="row">
                <div class="col-md-12 mb-3">
                    <label for="address_line1" class="form-label">Address Line 1 *</label>
                    <input type="text" class="form-control" id="address_line1" name="address_line1"
value="<?php echo h($customer['address_line1'] ?? ""); ?>" required>
                </div>
                <div class="col-md-12 mb-3">
                    <label for="address_line2" class="form-label">Address Line 2</label>
                    <input type="text" class="form-control" id="address_line2" name="address_line2"
value="<?php echo h($customer['address_line2'] ?? ""); ?>">
                </div>
                <div class="col-md-6 mb-3">
                    <label for="city" class="form-label">City *</label>

```

```

<input type="text" class="form-control" id="city" name="city"
value=<?php echo h($customer['city'] ?? ""); ?>" required>
</div>
<div class="col-md-3 mb-3">
<label for="state" class="form-label">State *</label>
<input type="text" class="form-control" id="state" name="state"
value=<?php echo h($customer['state'] ?? ""); ?>" required>
</div>
<div class="col-md-3 mb-3">
<label for="zip_code" class="form-label">ZIP Code *</label>
<input type="text" class="form-control" id="zip_code" name="zip_code"
value=<?php echo h($customer['zip_code'] ?? ""); ?>" required>
</div>
<div class="col-md-12 mb-3">
<label for="country" class="form-label">Country</label>
<input type="text" class="form-control" id="country" name="country"
value=<?php echo h($customer['country'] ?? 'USA'); ?>">
</div>
<div class="col-md-12 mb-3">
<label for="notes" class="form-label">Order Notes (Optional)</label>
<textarea class="form-control" id="notes" name="notes" rows="3"
placeholder="Special instructions or delivery notes..."></textarea>
</div>
</div>
</div>
</div>

<div class="card card-hollywood mb-4">
<div class="card-header card-header-hollywood">
<i class="fas fa-credit-card"></i> Payment Information
</div>
<div class="card-body">
<div class="alert alert-info">
<i class="fas fa-info-circle"></i>
<strong>Invoice System:</strong> This is an educational project.
No actual payment processing occurs. Your order will be recorded as an invoice.
</div>
<p class="mb-0"><i class="fas fa-file-invoice"></i> Payment Method: <strong>Invoice (Net
30)</strong></p>
</div>
</div>

<div class="d-flex justify-content-between">
<a href="cart.php" class="btn btn-outline-secondary">

```

```

<i class="fas fa-arrow-left"></i> Back to Cart
</a>
<button type="submit" name="place_order" class="btn btn-hollywood btn-lg">
<i class="fas fa-check-circle"></i> Place Order
</button>
</div>
</form>
</div>

<div class="col-md-4">
<div class="cart-summary">
<h4 class="text-hollywood mb-3">Order Summary</h4>

<div class="d-flex justify-content-between mb-2">
<span>Subtotal:</span>
<strong><?php echo formatCurrency($subtotal); ?></strong>
</div>

<div class="d-flex justify-content-between mb-2">
<span>Tax (<?php echo (TAX_RATE * 100); ?>%):</span>
<strong><?php echo formatCurrency($tax); ?></strong>
</div>

<div class="d-flex justify-content-between mb-2">
<span>Shipping:</span>
<strong><?php echo formatCurrency($shipping); ?></strong>
</div>

<?php if ($shipping == 0): ?>
<div class="alert alert-success py-2 small">
<i class="fas fa-truck"></i> Free shipping!
</div>
<?php endif; ?>

<hr>

<div class="d-flex justify-content-between mb-3">
<strong>Total:</strong>
<h4 class="text-hollywood mb-0">
<?php echo formatCurrency($total); ?>
</h4>
</div>

<div class="alert alert-secondary small">

```

```

        <i class="fas fa-shield-alt"></i> Your information is secure and encrypted
    </div>
</div>
</div>
</div>

<?php include '../includes/footer.php'; ?>

Hollywood101.php
<?php

require_once '../config/config.php';
require_once '../includes/session.php';

$page_title = 'Hollywood 101';

$database = new Database();
$db = $database->getConnection();

require_once MODEL_PATH . 'Article.php';

$article_model = new Article($db);

$category = $_GET['category'] ?? null;
$page = isset($_GET['page']) ? (int)$_GET['page'] : 1;
$limit = ARTICLES_PER_PAGE;
$offset = ($page - 1) * $limit;

$articles = $article_model->getPublished($category, $limit, $offset);
$total_articles = $article_model->getPublishedCount($category);
$categories = $article_model->getCategories();

$total_pages = ceil($total_articles / $limit);

include '../includes/header.php';
?>

<div class="container my-4">
    <div class="row mb-4">
        <div class="col-md-12">
            <h1 class="text-hollywood">δŸŽ¬ Hollywood 101</h1>

```

```

<p class="lead">Explore the rich history of Hollywood, Los Angeles, and the entertainment
industry</p>
</div>
</div>

<div class="row">
<!-- Sidebar -->
<div class="col-md-3">
<div class="card card-hollywood mb-4">
    <div class="card-header card-header-hollywood">
        <i class="fas fa-list"></i> Categories
    </div>
    <div class="list-group list-group-flush">
        <a href="hollywood101.php" class="list-group-item list-group-item-action <?php echo
!$category ? 'active' : ''; ?>">
            All Articles (<?php echo $total_articles; ?>)
        </a>
        <?php foreach ($categories as $cat): ?>
            <a href="hollywood101.php?category=<?php echo urlencode($cat['category']); ?>">
                <?php echo h($cat['category']); ?> (<?php echo $cat['count']; ?>)
            </a>
        <?php endforeach; ?>
    </div>
</div>

<div class="card card-hollywood">
    <div class="card-body">
        <h6 class="text-hollywood">ðŸŒŸ Learn About</h6>
        <ul class="list-unstyled small">
            <li>â€¢ Hollywood History</li>
            <li>â€¢ Famous Landmarks</li>
            <li>â€¢ Film Industry Insights</li>
            <li>â€¢ Los Angeles Culture</li>
            <li>â€¢ Behind the Scenes</li>
        </ul>
    </div>
</div>

<!-- Articles Grid -->
<div class="col-md-9">
<?php if ($category): ?>

```

```

<div class="alert alert-info">
<i class="fas fa-filter"></i> Showing articles in: <strong><?php echo h($category); ?></strong>
<a href="hollywood101.php" class="btn btn-sm btn-outline-secondary float-end">Clear
Filter</a>
</div>
<?php endif; ?>

<?php if (empty($articles)): ?>
    <div class="alert alert-warning">
        <i class="fas fa-exclamation-triangle"></i> No articles found.
    </div>
<?php else: ?>
    <div class="row row-cols-1 row-cols-md-2 g-4">
        <?php foreach ($articles as $article): ?>
            <div class="col">
                <div class="card article-card h-100">
                    <div class="article-image">
                        <?php if ($article['featured_image']): ?>
                            
                            class="card-img-top" alt="<?php echo h($article['title']); ?>">
                            style="height: 200px; object-fit: cover;">
                        <?php else: ?>
                            δŶŽ¥
                        <?php endif; ?>
                    </div>
                    <div class="card-body d-flex flex-column">
                        <div class="mb-2">
                            <span class="badge badge-hollywood">
                                <?php echo h($article['category']); ?>
                            </span>
                            <small class="text-muted float-end">
                                <i class="fas fa-eye"></i> <?php echo $article['view_count']; ?> views
                            </small>
                        </div>
                        <h5 class="card-title">
                            <a href="article.php?slug=<?php echo h($article['slug']); ?>">
                                <?php echo h($article['title']); ?>
                            </a>
                        </h5>
                        <p class="card-text text-muted flex-grow-1">

```

```

        <?php echo h($article['summary']) ?? substr(strip_tags($article['content']), 0, 150) .  

'...'); ?>  

</p>  
  

<div class="mt-auto">  

    <div class="d-flex justify-content-between align-items-center">  

        <small class="text-muted">  

            By <?php echo h($article['first_name'] . ' ' . $article['last_name']); ?><br>  

            <?php echo formatDate($article['published_at']); ?>  

        </small>  

        <a href="article.php?slug=<?php echo h($article['slug']); ?>" class="btn btn-sm  

        btn-hollywood">  

            Read More <i class="fas fa-arrow-right"></i>  

        </a>  

    </div>  

</div>  

<div class="card-footer bg-light">  

    <small class="text-muted">  

        <i class="fas fa-comments"></i> <?php echo $article['comment_count']; ?>  

    </small>  

</div>  

<?php endforeach; ?>  

</div>  
  

<!-- Pagination -->  

<?php if ($total_pages > 1): ?>  

<nav aria-label="Article pagination" class="mt-4">  

    <ul class="pagination justify-content-center">  

        <?php if ($page > 1): ?>  

        <li class="page-item">  

            <a class="page-link" href=?page=<?php echo $page-1; ?><?php echo $category ?  

'&category=' . urlencode($category) : ''; ?>">  

                Previous  

            </a>  

        </li>  

        <?php endif; ?>  

        <?php for ($i = 1; $i <= $total_pages; $i++): ?>  

        <li class="page-item <?php echo $i == $page ? 'active' : ''; ?>">

```

```

<a class="page-link" href="?page=<?php echo $i; ?><?php echo $category ?
'&category=' . urlencode($category) : ''; ?>">
    <?php echo $i; ?>
</a>
</li>
<?php endfor; ?>

<?php if ($page < $total_pages): ?>
<li class="page-item">
    <a class="page-link" href="?page=<?php echo $page+1; ?><?php echo $category ?
'&category=' . urlencode($category) : ''; ?>">
        Next
    </a>
</li>
<?php endif; ?>
</ul>
</nav>
<?php endif; ?>
<?php endif; ?>
</div>
</div>
</div>

```

<?php include '../includes/footer.php'; ?>

Order_detail.php

```

<?php

require_once '../config/config.php';
require_once '../includes/session.php';

requireCustomer();

```

```

$database = new Database();
$db = $database->getConnection();

require_once MODEL_PATH . 'Order.php';

```

```

$order_model = new Order($db);
$customer_id = $_SESSION['customer_id'];

$order_id = isset($_GET['id']) ? (int)$_GET['id'] : 0;

```

```

if (!$order_id) {

```

```

        setFlashMessage('error', 'Invalid order.');
        redirect('public/customer/orders.php');
    }

$order = $order_model->getOrderById($order_id);

if (!$order || $order['customer_id'] != $customer_id) {
    setFlashMessage('error', 'Order not found or access denied.');
    redirect('public/customer/orders.php');
}

$order_items = $order_model->getOrderItems($order_id);

$page_title = 'Order #' . $order_id;

$subtotal = 0;
foreach ($order_items as $item) {
    $subtotal += $item['subtotal'];
}
$tax = $subtotal * TAX_RATE;
$shipping = ($subtotal >= FREE_SHIPPING_THRESHOLD) ? 0 : STANDARD_SHIPPING_COST;

include '../includes/header.php';
?>

<div class="container my-4">
    <div class="d-flex justify-content-between align-items-center mb-4">
        <h1 class="text-hollywood">đŸ“, Order #<?php echo $order_id; ?></h1>
        <div>
            <button onclick="window.print()" class="btn btn-outline-hollywood">
                <i class="fas fa-print"></i> Print Receipt
            </button>
            <a href="orders.php" class="btn btn-outline-secondary">
                <i class="fas fa-arrow-left"></i> Back to Orders
            </a>
        </div>
    </div>

    <div class="row">
        <div class="col-md-8">
            <!-- Order Status -->
            <div class="card card-hollywood mb-4">
                <div class="card-body">
                    <div class="row">

```

```

<div class="col-md-6">
    <h5>Order Status</h5>
    <p class="mb-1">
        <span class="badge status-<?php echo strtolower($order['status']); ?> fs-6">
            <?php echo h($order['status']); ?>
        </span>
    </p>
    <small class="text-muted">
        Order Date: <?php echo formatDateTime($order['order_date']); ?>
    </small>
</div>
<div class="col-md-6">
    <h5>Customer Information</h5>
    <p class="mb-0">
        <strong><?php echo h($order['first_name'] . ' ' . $order['last_name']); ?></strong><br>
        <?php echo h($order['email']); ?><br>
        <?php echo h($order['phone']); ?>
    </p>
</div>
</div>
</div>

<!-- Shipping Address -->
<div class="card card-hollywood mb-4">
    <div class="card-header card-header-hollywood">
        <i class="fas fa-shipping-fast"></i> Shipping Address
    </div>
    <div class="card-body">
        <address class="mb-0">
            <?php echo h($order['shipping_address_line1']); ?><br>
            <?php if ($order['shipping_address_line2']): ?>
                <?php echo h($order['shipping_address_line2']); ?><br>
            <?php endif; ?>
            <?php echo h($order['shipping_city']); ?>,
            <?php echo h($order['shipping_state']); ?>
            <?php echo h($order['shipping_zip']); ?><br>
            <?php echo h($order['shipping_country']); ?>
        </address>
        <?php if ($order['notes']): ?>
        <hr>
        <p class="mb-0"><strong>Notes:</strong> <?php echo nl2br(h($order['notes'])); ?></p>
        <?php endif; ?>
    </div>

```

```

</div>

<!-- Order Items -->
<div class="card card-hollywood mb-4">
    <div class="card-header card-header-hollywood">
        <i class="fas fa-box"></i> Order Items
    </div>
    <div class="card-body p-0">
        <div class="table-responsive">
            <table class="table table-hover mb-0">
                <thead class="table-light">
                    <tr>
                        <th>Product</th>
                        <th class="text-center">Quantity</th>
                        <th class="text-end">Unit Price</th>
                        <th class="text-end">Subtotal</th>
                    </tr>
                </thead>
                <tbody>
                    <?php foreach ($order_items as $item): ?>
                    <tr>
                        <td>
                            <div class="d-flex align-items-center">
                                <div style="width: 50px; height: 50px; background: #f0f0f0; border-radius: 5px;
display: flex; align-items: center; justify-content: center; margin-right: 10px;">
                                    &#8226;
                                </div>
                            <div>
                                <strong><?php echo h($item['product_name']); ?></strong><br>
                                <small class="text-muted">SKU: <?php echo h($item['sku']); ?></small>
                            </div>
                            </div>
                        </td>
                        <td class="text-center align-middle">
                            <?php echo $item['quantity']; ?>
                        </td>
                        <td class="text-end align-middle">
                            <?php echo formatCurrency($item['unit_price']); ?>
                        </td>
                        <td class="text-end align-middle">
                            <strong><?php echo formatCurrency($item['subtotal']); ?></strong>
                        </td>
                    </tr>
                <?php endforeach; ?>

```

```

        </tbody>
    </table>
</div>
</div>

</div>
</div>

<div class="col-md-4">
<!-- Order Summary --&gt;
&lt;div class="card card-hollywood mb-4"&gt;
    &lt;div class="card-header card-header-hollywood"&gt;
        &lt;i class="fas fa-calculator"&gt;&lt;/i&gt; Order Summary
    &lt;/div&gt;
    &lt;div class="card-body"&gt;
        &lt;div class="d-flex justify-content-between mb-2"&gt;
            &lt;span&gt;Subtotal:&lt;/span&gt;
            &lt;strong&gt;&lt;?php echo formatCurrency($subtotal); ?&gt;&lt;/strong&gt;
        &lt;/div&gt;

        &lt;div class="d-flex justify-content-between mb-2"&gt;
            &lt;span&gt;Tax (&lt;?php echo (TAX_RATE * 100); ?&gt;%):&lt;/span&gt;
            &lt;strong&gt;&lt;?php echo formatCurrency($tax); ?&gt;&lt;/strong&gt;
        &lt;/div&gt;

        &lt;div class="d-flex justify-content-between mb-2"&gt;
            &lt;span&gt;Shipping:&lt;/span&gt;
            &lt;strong&gt;&lt;?php echo formatCurrency($shipping); ?&gt;&lt;/strong&gt;
        &lt;/div&gt;

        &lt;hr&gt;

        &lt;div class="d-flex justify-content-between mb-3"&gt;
            &lt;strong&gt;Total Paid:&lt;/strong&gt;
            &lt;h4 class="text-hollywood mb-0"&gt;
                &lt;?php echo formatCurrency($order['total_amount']); ?&gt;
            &lt;/h4&gt;
        &lt;/div&gt;

        &lt;div class="alert alert-info small mb-0"&gt;
            &lt;i class="fas fa-file-invoice"&gt;&lt;/i&gt; Payment Method: Invoice
        &lt;/div&gt;
    &lt;/div&gt;
&lt;/div&gt;
</pre>

```

```

<!-- Tracking Info -->
<?php if ($order['status'] === 'Shipped' || $order['status'] === 'Delivered'): ?>
    <div class="card card-hollywood">
        <div class="card-header card-header-hollywood">
            <i class="fas fa-truck"></i> Shipping Info
        </div>
        <div class="card-body">
            <p class="mb-1"><strong>Status:</strong> <?php echo h($order['status']); ?></p>
            <?php if ($order['fulfilled_at']): ?>
                <p class="mb-0">
                    <small class="text-muted">
                        <?php echo $order['status'] === 'Shipped' ? 'Shipped' : 'Delivered'; ?> on:
                        <?php echo formatDateTime($order['fulfilled_at']); ?>
                    </small>
                </p>
            <?php endif; ?>
        </div>
    </div>
<?php endif; ?>
</div>
</div>

<div class="mt-4 text-center">
    <a href="shop.php" class="btn btn-hollywood">
        <i class="fas fa-shopping-bag"></i> Continue Shopping
    </a>
</div>
</div>

<style>
    @media print {
        .btn, nav, footer, .navbar { display: none !important; }
        .container { max-width: 100% !important; }
    }
</style>

<?php include '../includes/footer.php'; ?>

```

Orders.php
<?php

```

require_once '../config/config.php';
require_once '../includes/session.php';

```

```

requireLoginForAction('view your orders');

$page_title = 'Order History';

$database = new Database();
$db = $database->getConnection();

require_once MODEL_PATH . 'Order.php';

$order_model = new Order($db);
$customer_id = $_SESSION['customer_id'];

$orders = $order_model->getCustomerOrders($customer_id);

include '../includes/header.php';
?>

<div class="container my-4">
    <h1 class="text-hollywood mb-4">Order History</h1>

    <?php if (empty($orders)): ?>
    <div class="row">
        <div class="col-md-8 mx-auto">
            <div class="card card-hollywood text-center py-5">
                <div class="card-body">
                    <i class="fas fa-receipt fa-4x text-muted mb-3"></i>
                    <h3>No orders yet</h3>
                    <p class="text-muted">Start shopping to create your first order!</p>
                    <a href="shop.php" class="btn btn-hollywood btn-lg mt-3">
                        <i class="fas fa-shopping-bag"></i> Start Shopping
                    </a>
                </div>
            </div>
        </div>
    </?php else: ?>
    <div class="row">
        <div class="col-md-12">
            <?php foreach ($orders as $order): ?>
            <div class="card card-hollywood mb-3">
                <div class="card-body">
                    <div class="row align-items-center">
                        <div class="col-md-2">
                            <h5 class="mb-1">Order #<?php echo $order['order_id']; ?></h5>

```

```

<small class="text-muted">
    <?php echo formatDate($order['order_date']); ?>
</small>
</div>
<div class="col-md-3">
<strong>Total:</strong><br>
<span class="text-hollywood h5">
    <?php echo formatCurrency($order['total_amount']); ?>
</span>
</div>
<div class="col-md-3">
<strong>Status:</strong><br>
<span class="badge status-<?php echo strtolower($order['status']); ?>">
    <?php echo h($order['status']); ?>
</span>
</div>
<div class="col-md-2">
<strong>Ship To:</strong><br>
<small class="text-muted">
    <?php echo h($order['shipping_city']); ?>, <?php echo h($order['shipping_state']);
?>
</small>
</div>
<div class="col-md-2 text-end">
<a href="order_detail.php?id=<?php echo $order['order_id']; ?>">
    class="btn btn-outline-hollywood">
        <i class="fas fa-eye"></i> View Details
    </a>
</div>
</div>
<?php endforeach; ?>
</div>
</div>

<div class="mt-4">
<a href="shop.php" class="btn btn-hollywood">
    <i class="fas fa-shopping-bag"></i> Continue Shopping
</a>
</div>
<?php endif; ?>
</div>

```

```

<?php include '../includes/footer.php'; ?>

Product_detail.php
<?php

require_once '../config/config.php';
require_once '../includes/session.php';

$database = new Database();
$db = $database->getConnection();

require_once MODEL_PATH . 'Product.php';
require_once MODEL_PATH . 'Cart.php';

$product_model = new Product($db);
$cart = new Cart($db);

$product_id = isset($_GET['id']) ? (int)$_GET['id'] : 0;

if (!$product_id) {
    setFlashMessage('error', 'Invalid product.');
    redirect('public/customer/shop.php');
}

if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['add_to_cart'])) {
    if (!isLoggedIn()) {
        $_SESSION['return_url'] = $_SERVER['REQUEST_URI'];
        setFlashMessage('error', 'Please login to add items to your cart.');
        redirect('public/login.php');
    }

    if (!verifyCSRFToken($_POST['csrf_token'] ?? '')) {
        setFlashMessage('error', 'Invalid security token.');
        redirect('public/customer/product_detail.php?id=' . $product_id);
    }
}

$quantity = isset($_POST['quantity']) ? (int)$_POST['quantity'] : 1;

if ($quantity < 1) {
    setFlashMessage('error', 'Invalid quantity.');
} else {
    $product = $product_model->getById($product_id);
}

```

```

if ($quantity > $product['quantity_available']) {
    setFlashMessage('error', 'Not enough stock available. Only ' . $product['quantity_available'] . ' items in
stock.');
} else {
    if ($cart->addItem($_SESSION['customer_id'], $product_id, $quantity)) {
        logAudit('ADD_TO_CART', 'shopping_cart', $product_id);
        setFlashMessage('success', 'Product added to cart!');
        redirect('public/customer/cart.php');
    } else {
        setFlashMessage('error', 'Failed to add product to cart.');
    }
}
}

$product = $product_model->getById($product_id);

if (!$product) {
    setFlashMessage('error', 'Product not found.');
    redirect('public/customer/shop.php');
}

$page_title = $product['product_name'];

include '../../../includes/header.php';
?>

<div class="container my-4">
    <nav aria-label="breadcrumb">
        <ol class="breadcrumb">
            <li class="breadcrumb-item"><a href="shop.php">Shop</a></li>
            <li class="breadcrumb-item"><a href="shop.php?category=<?php echo $product['category_id']; ?>">
                <?php echo h($product['category_name']); ?>
            </a></li>
            <li class="breadcrumb-item active"><?php echo h($product['product_name']); ?></li>
        </ol>
    </nav>

    <div class="row">
        <div class="col-md-5">
            <div class="card card-hollywood">
                <div class="product-image" style="height: 400px;">
                    <?php if ($product['image_url']): ?>
                    
                </div>
            </div>
        </div>
    </div>

```

```

        class="card-img-top" alt="<?php echo h($product['product_name']); ?>" style="height: 100%; object-fit: cover;">
<?php else: ?>
<div class="text-center d-flex align-items-center justify-content-center h-100">
    <span style="font-size: 8rem;">δŶŽ¬</span>
</div>
<?php endif; ?>
</div>
</div>
</div>

<div class="col-md-7">
<div class="card card-hollywood">
    <div class="card-body">
        <span class="badge badge-hollywood mb-3">
            <?php echo h($product['category_name']); ?>
        </span>

        <h1 class="h2 mb-3"><?php echo h($product['product_name']); ?></h1>

        <div class="mb-3">
            <span class="display-5 text-hollywood fw-bold">
                <?php echo formatCurrency($product['price']); ?>
            </span>
        </div>

        <div class="mb-3">
            <strong>SKU:</strong> <?php echo h($product['sku']); ?>
        </div>

        <div class="mb-3">
            <strong>Availability:</strong>
            <?php if ($product['quantity_available'] > 0): ?>
                <span class="badge badge-stock-ok">
                    <i class="fas fa-check"></i> In Stock (<?php echo $product['quantity_available']; ?>
available)
                </span>
            <?php else: ?>
                <span class="badge badge-stock-low">
                    <i class="fas fa-times"></i> Out of Stock
                </span>
            <?php endif; ?>
        </div>
    </div>
</div>

```

```

<hr>

<h5>Description</h5>
<p class="text-muted"><?php echo nl2br(h($product['description'])); ?></p>

<hr>

<?php if ($product['quantity_available'] > 0): ?>
<form method="POST" action="" class="mt-4">
    <input type="hidden" name="csrf_token" value="<?php echo generateCSRFToken(); ?>">
    required>
        <div class="row align-items-end">
            <div class="col-md-4 mb-3">
                <label for="quantity" class="form-label">Quantity</label>
                <input type="number" class="form-control" id="quantity" name="quantity"
                    value="1" min="1" max="<?php echo $product['quantity_available']; ?>">
            </div>
            <div class="col-md-8 mb-3">
                <button type="submit" name="add_to_cart" class="btn btn-hollywood btn-lg w-100">
                    <i class="fas fa-shopping-cart"></i> Add to Cart
                </button>
            </div>
        </div>
    </form>
    <?php else: ?>
        <div class="alert alert-warning">
            <i class="fas fa-exclamation-triangle"></i> This product is currently out of stock.
        </div>
    <?php endif; ?>

    <div class="mt-3">
        <a href="shop.php" class="btn btn-outline-secondary">
            <i class="fas fa-arrow-left"></i> Continue Shopping
        </a>
    </div>
</div>
</div>
</div>

<!-- Additional Product Info -->
<div class="row mt-4">

```

```

<div class="col-md-12">
  <div class="card card-hollywood">
    <div class="card-body">
      <h4 class="text-hollywood">Product Features</h4>
      <ul>
        <li>Authentic Hollywood-inspired design</li>
        <li>High-quality materials and craftsmanship</li>
        <li>Perfect for Hollywood enthusiasts and collectors</li>
        <li>Ships from Los Angeles, California</li>
        <li>Makes a great gift for film lovers</li>
      </ul>
    </div>
  </div>
</div>

```

```
<?php include '../includes/footer.php'; ?>
```

Profile.php

```
<?php
```

```
require_once '../../config/config.php';
require_once '../../includes/session.php';
```

```
requireLoginForAction('view your profile');
```

```
$page_title = 'My Profile';
```

```
$database = new Database();
$db = $database->getConnection();
```

```
require_once MODEL_PATH . 'Customer.php';
$customer_model = new Customer($db);
```

```
$customer_query = "SELECT customer_id FROM customers WHERE user_id = :user_id";
$customer_stmt = $db->prepare($customer_query);
$customer_stmt->execute([':user_id' => getCurrentUserId()]);
$customer_data = $customer_stmt->fetch();
$customer_id = $customer_data['customer_id'];
```

```
if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['update_profile'])) {
  if (!verifyCSRFToken($_POST['csrf_token'] ?? '')) {
    setFlashMessage('error', 'Invalid security token.');
  }
}
```

```

} else {
$first_name = trim($_POST['first_name']);
$last_name = trim($_POST['last_name']);
$phone = trim($_POST['phone']);
$address_line1 = trim($_POST['address_line1']);
$address_line2 = trim($_POST['address_line2']);
$city = trim($_POST['city']);
$state = trim($_POST['state']);
$zip_code = trim($_POST['zip_code']);

$data = [
'first_name' => $first_name,
'last_name' => $last_name,
'phone' => $phone,
'address_line1' => $address_line1,
'address_line2' => $address_line2,
'city' => $city,
'state' => $state,
'zip_code' => $zip_code,
'country' => 'USA'
];
}

if ($customer_model->update($customer_id, $data)) {
logAudit('UPDATE_PROFILE', 'customers', $customer_id, null, "$first_name $last_name");
setFlashMessage('success', 'Profile updated successfully!');
} else {
setFlashMessage('error', 'Failed to update profile.');
}
redirect('public/customer/profile.php');
}

if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['change_password'])) {
if (!verifyCSRFToken($_POST['csrf_token'] ?? '')) {
setFlashMessage('error', 'Invalid security token.');
} else {
$current_password = $_POST['current_password'];
$new_password = $_POST['new_password'];
$confirm_password = $_POST['confirm_password'];

$user_query = "SELECT password_hash FROM users WHERE user_id = :user_id";
$user_stmt = $db->prepare($user_query);
$user_stmt->execute([':user_id' => getCurrentUserId()]);
$user = $user_stmt->fetch();
}
}

```

```

if (!password_verify($current_password, $user['password_hash'])) {
    setFlashMessage('error', 'Current password is incorrect.');
} elseif ($new_password !== $confirm_password) {
    setFlashMessage('error', 'New passwords do not match.');
} elseif (strlen($new_password) < 6) {
    setFlashMessage('error', 'Password must be at least 6 characters.');
} else {
    $new_hash = password_hash($new_password, PASSWORD_BCRYPT, ['cost' => 10]);
    $update_query = "UPDATE users SET password_hash = :hash WHERE user_id = :user_id";
    $update_stmt = $db->prepare($update_query);

    if ($update_stmt->execute([':hash' => $new_hash, ':user_id' => getUserId()])) {
        logAudit('CHANGE_PASSWORD', 'users', getUserId());
        setFlashMessage('success', 'Password changed successfully!');
    } else {
        setFlashMessage('error', 'Failed to change password.');
    }
}
}

redirect('public/customer/profile.php');
}

}

$profile = $customer_model->getById($customer_id);

include '../includes/header.php';
?>

<div class="container my-4">
    <h1 class="text-hollywood mb-4">¤ My Profile</h1>

    <div class="row">
        <!-- Profile Information -->
        <div class="col-md-8 mb-4">
            <div class="card card-hollywood">
                <div class="card-header card-header-hollywood">
                    <i class="fas fa-user"></i> Profile Information
                </div>
                <div class="card-body">
                    <form method="POST" action="">
                        <input type="hidden" name="csrf_token" value="<?php echo generateCSRFToken(); ?>">

                        <div class="row">
                            <div class="col-md-6 mb-3">

```

```

<label for="first_name" class="form-label">First Name *</label>
<input type="text" class="form-control" name="first_name" id="first_name"
value="<?php echo h($profile['first_name']); ?>" required>
</div>
<div class="col-md-6 mb-3">
<label for="last_name" class="form-label">Last Name *</label>
<input type="text" class="form-control" name="last_name" id="last_name"
value="<?php echo h($profile['last_name']); ?>" required>
</div>
</div>

<div class="mb-3">
<label for="email" class="form-label">Email</label>
<input type="email" class="form-control" id="email"
value="<?php echo h($profile['email']); ?>" disabled>
<small class="text-muted">Email cannot be changed</small>
</div>

<div class="mb-3">
<label for="phone" class="form-label">Phone *</label>
<input type="tel" class="form-control" name="phone" id="phone"
value="<?php echo h($profile['phone']); ?>" required>
</div>

<h6 class="text-hollywood mb-3 mt-4">Default Shipping Address</h6>

<div class="mb-3">
<label for="address_line1" class="form-label">Address Line 1 *</label>
<input type="text" class="form-control" name="address_line1" id="address_line1"
value="<?php echo h($profile['address_line1']); ?>" required>
</div>

<div class="mb-3">
<label for="address_line2" class="form-label">Address Line 2</label>
<input type="text" class="form-control" name="address_line2" id="address_line2"
value="<?php echo h($profile['address_line2']); ?>">
</div>

<div class="row">
<div class="col-md-6 mb-3">
<label for="city" class="form-label">City *</label>
<input type="text" class="form-control" name="city" id="city"
value="<?php echo h($profile['city']); ?>" required>
</div>

```

```

<div class="col-md-3 mb-3">
    <label for="state" class="form-label">State *</label>
    <input type="text" class="form-control" name="state" id="state"
        value="<?php echo h($profile['state']); ?>" maxlength="2" required>
</div>
<div class="col-md-3 mb-3">
    <label for="zip_code" class="form-label">ZIP Code *</label>
    <input type="text" class="form-control" name="zip_code" id="zip_code"
        value="<?php echo h($profile['zip_code']); ?>" required>
</div>
</div>

<button type="submit" name="update_profile" class="btn btn-hollywood">
    <i class="fas fa-save"></i> Save Changes
</button>
</form>
</div>
</div>

<!-- Account Actions -->
<div class="col-md-4 mb-4">
<div class="card card-hollywood mb-3">
    <div class="card-header card-header-hollywood">
        <i class="fas fa-lock"></i> Security
    </div>
    <div class="card-body">
        <button class="btn btn-outline-hollywood w-100" data-bs-toggle="modal"
            data-bs-target="#changePasswordModal">
            <i class="fas fa-key"></i> Change Password
        </button>
    </div>
</div>

<div class="card card-hollywood mb-3">
    <div class="card-header card-header-hollywood">
        <i class="fas fa-chart-bar"></i> Account Stats
    </div>
    <div class="card-body">
        <?php
            $orders_query = "SELECT COUNT(*) as count FROM orders WHERE customer_id = :cid";
            $orders_stmt = $db->prepare($orders_query);
            $orders_stmt->execute([':cid' => $customer_id]);
            $order_count = $orders_stmt->fetch()['count'];
    
```

```

$spent_query = "SELECT SUM(total_amount) as total FROM orders
                WHERE customer_id = :cid AND status != 'Cancelled'";
$spent_stmt = $db->prepare($spent_query);
$spent_stmt->execute([':cid' => $customer_id]);
$total_spent = $spent_stmt->fetch()['total'] ?? 0;
?>
<p><strong>Total Orders:</strong> <?php echo number_format($order_count); ?></p>
<p><strong>Total Spent:</strong> <?php echo formatCurrency($total_spent); ?></p>
<p><strong>Member Since:</strong> <?php echo formatDate($profile['created_at']); ?></p>
</div>
</div>

<div class="card card-hollywood">
    <div class="card-header card-header-hollywood">
        <i class="fas fa-link"></i> Quick Links
    </div>
    <div class="card-body">
        <a href="orders.php" class="btn btn-sm btn-outline-hollywood w-100 mb-2">
            <i class="fas fa-shopping-bag"></i> My Orders
        </a>
        <a href="shop.php" class="btn btn-sm btn-outline-hollywood w-100 mb-2">
            <i class="fas fa-store"></i> Continue Shopping
        </a>
        <a href="hollywood101.php" class="btn btn-sm btn-outline-hollywood w-100">
            <i class="fas fa-book"></i> Hollywood 101
        </a>
    </div>
</div>
</div>
</div>
</div>

<!-- Change Password Modal -->
<div class="modal fade" id="changePasswordModal" tabindex="-1">
    <div class="modal-dialog">
        <div class="modal-content">
            <form method="POST" action="">
                <input type="hidden" name="csrf_token" value="<?php echo generateCSRFToken(); ?>">

                <div class="modal-header bg-hollywood text-white">
                    <h5 class="modal-title">Change Password</h5>
                    <button type="button" class="btn-close btn-close-white" data-bs-dismiss="modal"></button>
                </div>

```

```

<div class="modal-body">
<div class="mb-3">
<label for="current_password" class="form-label">Current Password *</label>
<input type="password" class="form-control" name="current_password"
       id="current_password" required>
</div>
<div class="mb-3">
<label for="new_password" class="form-label">New Password *</label>
<input type="password" class="form-control" name="new_password"
       id="new_password" minlength="6" required>
<small class="text-muted">Minimum 6 characters</small>
</div>
<div class="mb-3">
<label for="confirm_password" class="form-label">Confirm New Password *</label>
<input type="password" class="form-control" name="confirm_password"
       id="confirm_password" minlength="6" required>
</div>
</div>
<div class="modal-footer">
<button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Cancel</button>
<button type="submit" name="change_password" class="btn btn-hollywood">
<i class="fas fa-key"></i> Change Password
</button>
</div>
</form>
</div>
</div>
</div>

<?php include '../includes/footer.php'; ?>

```

Shop.php

```

<?php

require_once '../config/config.php';
require_once '../includes/session.php';

$page_title = 'Shop Hollywood Merchandise';

$database = new Database();
$db = $database->getConnection();

require_once MODEL_PATH . 'Product.php';
require_once MODEL_PATH . 'Category.php';

```

```

require_once MODEL_PATH . 'Customer.php';

$product = new Product($db);
$category = new Category($db);

if (isLoggedIn()) {
    require_once MODEL_PATH . 'Customer.php';
    $customer_model = new Customer($db);
    $customer = $customer_model->getById(getCurrentUserId());
    if ($customer) {
        $_SESSION['customer_id'] = $customer['customer_id'];
    }
}

$search = $_GET['search'] ?? '';
$category_id = $_GET['category'] ?? null;
$page = isset($_GET['page']) ? (int)$_GET['page'] : 1;
$limit = PRODUCTS_PER_PAGE;
$offset = ($page - 1) * $limit;

if (!empty($search)) {
    $products = $product->search($search);
    $total_products = count($products);
    $products = array_slice($products, $offset, $limit);
} else {
    $products = $product->getAll($category_id, $limit, $offset);
    $total_products = $product->getCount($category_id);
}

$categories = $category->getAll();

$total_pages = ceil($total_products / $limit);

include '../includes/header.php';
?>

<style>
.product-image {
    height: 250px;
    overflow: hidden;
    display: flex;
    align-items: center;
    justify-content: center;
    background: #f8f9fa;
}

```

```
}
```

```
.product-image img {
    width: 100%;
    height: 100%;
    object-fit: cover;
}

</style>

<div class="container my-4">
    <div class="row mb-4">
        <div class="col-md-12">
            <h1 class="text-hollywood">🎬 Shop Hollywood Merchandise</h1>
            <p class="lead">Discover exclusive Hollywood-inspired fashion, decor, and collectibles</p>
        </div>
    </div>

    <div class="row mb-4">
        <!-- Search Bar -->
        <div class="col-md-8">
            <form method="GET" action="" class="search-bar">
                <div class="input-group">
                    <span class="input-group-text bg-white">
                        <i class="fas fa-search text-muted"></i>
                    </span>
                    <input type="text" class="form-control" name="search"
                        placeholder="Search for Hollywood products..." value="<?php echo h($search); ?>">
                    <button type="submit" class="btn btn-hollywood">Search</button>
                </div>
            </form>
        </div>
        <div class="col-md-4 text-end">
            <?php if ($search || $category_id): ?>
                <a href="shop.php" class="btn btn-outline-hollywood">
                    <i class="fas fa-times"></i> Clear Filters
                </a>
            <?php endif; ?>
        </div>
    </div>

    <div class="row">
        <!-- Sidebar - Categories -->
        <div class="col-md-3">
```

```

<div class="card card-hollywood mb-4">
    <div class="card-header card-header-hollywood">
        <i class="fas fa-list"></i> Categories
    </div>
    <div class="list-group list-group-flush">
        <a href="shop.php" class="list-group-item list-group-item-action <?php echo !$category_id ? 'active' : ''; ?>">
            All Products (<?php echo $product->getCount(); ?>)
        </a>
        <?php foreach ($categories as $cat): ?>
            <a href="shop.php?category=<?php echo $cat['category_id']; ?>" class="list-group-item list-group-item-action <?php echo $category_id == $cat['category_id'] ? 'active' : ''; ?>">
                <?php echo h($cat['category_name']); ?> (<?php echo $cat['product_count']; ?>)
            </a>
        <?php endforeach; ?>
    </div>
</div>

<!-- Featured Info -->
<div class="card card-hollywood">
    <div class="card-body">
        <h6 class="text-hollywood">★ Why Shop With Us?</h6>
        <ul class="list-unstyled small">
            <li>✓ Authentic Hollywood merchandise</li>
            <li>✓ Exclusive limited editions</li>
            <li>✓ Fast shipping from LA</li>
            <li>✓ Secure checkout</li>
        </ul>
    </div>
</div>

<!-- Products Grid -->
<div class="col-md-9">
    <?php if ($search): ?>
        <div class="alert alert-info">
            <i class="fas fa-info-circle"></i> Showing results for: <strong>"<?php echo h($search); ?>"</strong>
            (<?php echo $total_products; ?> products found)
        </div>
    <?php endif; ?>
<?php if (empty($products)): ?>

```

```

<div class="alert alert-warning">
<i class="fas fa-exclamation-triangle"></i> No products found. Try adjusting your search or
filters.
</div>
<?php else: ?>
<div class="row row-cols-1 row-cols-md-2 row-cols-lg-3 g-4">
<?php foreach ($products as $prod): ?>
<div class="col">
    <div class="card product-card card-hollywood h-100">
        <div class="product-image">
            <?php if ($prod['image_url']): ?>
                
                class="card-img-top" alt="<?php echo h($prod['product_name']); ?>">
            <?php else: ?>
                <div class="text-center py-5"><img alt="camera icon" /></div>
            <?php endif; ?>
        </div>
        <div class="card-body d-flex flex-column">
            <span class="badge badge-hollywood mb-2 align-self-start">
                <?php echo h($prod['category_name']); ?>
            </span>
            <h5 class="card-title product-name">
                <?php echo h($prod['product_name']); ?>
            </h5>
            <p class="card-text text-muted small flex-grow-1">
                <?php echo h(substr($prod['description'], 0, 80)); ?>...
            </p>
            <div class="d-flex justify-content-between align-items-center mt-2">
                <span class="product-price">
                    <?php echo formatCurrency($prod['price']); ?>
                </span>
                <?php if ($prod['quantity_available'] > 0): ?>
                    <span class="badge bg-success">In Stock</span>
                <?php else: ?>
                    <span class="badge bg-danger">Out of Stock</span>
                <?php endif; ?>
            </div>
            <div class="mt-3">
                <a href="product_detail.php?id=<?php echo $prod['product_id']; ?>">
                    class="btn btn-hollywood w-100">
                    <i class="fas fa-eye"></i> View Details
                </a>
            </div>
        </div>
    </div>

```

```

        </div>
    </div>
<?php endforeach; ?>
</div>

<!-- Pagination --&gt;
&lt;?php if ($total_pages &gt; 1): ?&gt;
&lt;nav aria-label="Product pagination" class="mt-4"&gt;
&lt;ul class="pagination justify-content-center"&gt;
    &lt;?php if ($page &gt; 1): ?&gt;
        &lt;li class="page-item"&gt;
            &lt;a class="page-link" href=?&gt;&lt;?php echo $page-1; ?&gt;&lt;?php echo $category_id ? '&amp;category='.$category_id : ''; ?&gt;&lt;?php echo $search ? '&amp;search=' . urlencode($search) : ''; ?&gt;"&gt;
                Previous
            &lt;/a&gt;
        &lt;/li&gt;
    &lt;?php endif; ?&gt;

    &lt;?php for ($i = 1; $i &lt;= $total_pages; $i++): ?&gt;
        &lt;li class="page-item &lt;?php echo $i == $page ? 'active' : '' ?&gt;"&gt;
            &lt;a class="page-link" href=?&gt;&lt;?php echo $i; ?&gt;&lt;?php echo $category_id ? '&amp;category='.$category_id : ''; ?&gt;&lt;?php echo $search ? '&amp;search=' . urlencode($search) : ''; ?&gt;"&gt;
                &lt;?php echo $i; ?&gt;
            &lt;/a&gt;
        &lt;/li&gt;
    &lt;?php endfor; ?&gt;

    &lt;?php if ($page &lt; $total_pages): ?&gt;
        &lt;li class="page-item"&gt;
            &lt;a class="page-link" href=?&gt;&lt;?php echo $page+1; ?&gt;&lt;?php echo $category_id ? '&amp;category='.$category_id : ''; ?&gt;&lt;?php echo $search ? '&amp;search=' . urlencode($search) : ''; ?&gt;"&gt;
                Next
            &lt;/a&gt;
        &lt;/li&gt;
    &lt;?php endif; ?&gt;
&lt;/ul&gt;
&lt;/nav&gt;
&lt;?php endif; ?&gt;
&lt;?php endif; ?&gt;
&lt;/div&gt;
&lt;/div&gt;
&lt;/div&gt;

&lt;?php include '../includes/footer.php'; ?&gt;</pre>

```

```

Articles.php
<?php

require_once '../config/config.php';
require_once '../includes/session.php';

requireRole(ROLE_CONTENT_EDITOR);

$page_title = 'Article Management';

$database = new Database();
$db = $database->getConnection();

require_once MODEL_PATH . 'Article.php';
$article_model = new Article($db);

if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['create_article'])) {
    if (!verifyCSRFToken($_POST['csrf_token'] ?? '')) {
        setFlashMessage('error', 'Invalid security token.');
    } else {
        $title = trim($_POST['title']);
        $content = trim($_POST['content']);
        $category = trim($_POST['category']);
        $summary = trim($_POST['summary'] ?? '');
        $is_published = isset($_POST['is_published']) ? 1 : 0;

        $emp_query = "SELECT employee_id FROM employees WHERE user_id = :user_id";
        $emp_stmt = $db->prepare($emp_query);
        $emp_stmt->execute([':user_id' => getCurrentUserId()]);
        $employee = $emp_stmt->fetch();

        if (!$employee) {
            setFlashMessage('error', 'Employee record not found.');
            redirect('public/employee/articles.php');
        }

        $slug = strtolower(trim(preg_replace('/[^A-Za-z0-9-]+/', '-', $title)));
        $data = [
            'title' => $title,
            'slug' => $slug,
            'content' => $content,
            'summary' => $summary ?: substr(strip_tags($content), 0, 200),
        ];
    }
}

```

```

'author_id' => $employee['employee_id'],
'category' => $category,
'featured_image' => null,
'is_published' => $is_published
];

if ($article_model->create($data)) {
logAudit('CREATE_ARTICLE', 'articles', null, null, $title);
setFlashMessage('success', 'Article created successfully!');
} else {
setFlashMessage('error', 'Failed to create article.');
}
redirect('public/employee/articles.php');
}

if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['update_article'])) {
if (!verifyCSRFToken($_POST['csrf_token'] ?? '')) {
setFlashMessage('error', 'Invalid security token.');
} else {
$article_id = (int)$_POST['article_id'];
$title = trim($_POST['title']);
$content = trim($_POST['content']);
$category = trim($_POST['category']);
$is_published = isset($_POST['is_published']) ? 1 : 0;

if ($article_model->update($article_id, $title, $content, $category, $is_published)) {
logAudit('UPDATE_ARTICLE', 'articles', $article_id, null, $title);
setFlashMessage('success', 'Article updated successfully!');
} else {
setFlashMessage('error', 'Failed to update article.');
}
redirect('public/employee/articles.php');
}
}

if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['delete_article'])) {
if (!verifyCSRFToken($_POST['csrf_token'] ?? '')) {
setFlashMessage('error', 'Invalid security token.');
} else {
$article_id = (int)$_POST['article_id'];

if ($article_model->delete($article_id)) {
logAudit('DELETE_ARTICLE', 'articles', $article_id);
}
}
}

```

```

        setFlashMessage('success', 'Article deleted successfully!');
    } else {
        setFlashMessage('error', 'Failed to delete article.');
    }
    redirect('public/employee/articles.php');
}
}

$articles = $article_model->getAll(false); // Include unpublished

$published_count = 0;
$draft_count = 0;
$total_comments = 0;

foreach ($articles as $article) {
    if ($article['is_published']) {
        $published_count++;
    } else {
        $draft_count++;
    }
    $total_comments += $article['comment_count'] ?? 0;
}

include '../includes/header.php';
?>

<div class="container-fluid my-4">
    <div class="d-flex justify-content-between align-items-center mb-4">
        <h1 class="text-hollywood"> Article Management</h1>
        <button class="btn btn-hollywood" data-bs-toggle="modal" data-bs-target="#createArticleModal">
            <i class="fas fa-plus"></i> New Article
        </button>
    </div>

    <!-- Statistics -->
    <div class="row mb-4">
        <div class="col-md-4">
            <div class="dashboard-card">
                <h3><?php echo number_format(count($articles)); ?></h3>
                <p>Total Articles</p>
            </div>
        </div>
        <div class="col-md-4">
            <div class="dashboard-card" style="border-left-color: #28a745;">

```

```

<h3><?php echo number_format($published_count); ?></h3>
<p>Published</p>
</div>
</div>
<div class="col-md-4">
<div class="dashboard-card" style="border-left-color: #ffc107;">
    <h3><?php echo number_format($draft_count); ?></h3>
    <p>Drafts</p>
</div>
</div>
</div>

<!-- Articles Table -->
<div class="card card-hollywood">
<div class="card-header card-header-hollywood">
    <i class="fas fa-newspaper"></i> Articles
</div>
<div class="card-body">
<?php if (empty($articles)): ?>
    <div class="text-center py-5 text-muted">
        <i class="fas fa-inbox fa-3x mb-3"></i>
        <p>No articles found. Create your first article!</p>
    </div>
<?php else: ?>
    <div class="table-responsive">
        <table class="table table-hover">
            <thead class="table-light">
                <tr>
                    <th>Title</th>
                    <th>Category</th>
                    <th>Author</th>
                    <th>Published Date</th>
                    <th>Status</th>
                    <th>Comments</th>
                    <th>Actions</th>
                </tr>
            </thead>
            <tbody>
                <?php foreach ($articles as $article): ?>
                <tr>
                    <td>
                        <strong><?php echo h($article['title']); ?></strong>
                    </td>
                    <td><?php echo h($article['category']); ?></td>

```

```

<td>
    <?php echo h(($article['first_name'] ?? '') . ' ' . ($article['last_name'] ?? '')); ?>
</td>
<td>
    <?php if (!empty($article['published_at'])): ?>
    <?php echo formatDate($article['published_at']); ?>
    <?php else: ?>
        <span class="text-muted">Not published</span>
    <?php endif; ?>
</td>
<td>
    <?php if ($article['is_published']): ?>
        <span class="badge bg-success">Published</span>
    <?php else: ?>
        <span class="badge bg-warning">Draft</span>
    <?php endif; ?>
</td>
<td class="text-center">
    <?php echo $article['comment_count'] ?? 0; ?>
</td>
<td>
    <button class="btn btn-sm btn-outline-hollywood"
    onclick="editArticle(<?php echo htmlspecialchars(json_encode($article)); ?>)">
        <i class="fas fa-edit"></i>
    </button>
    <button class="btn btn-sm btn-outline-danger"
    onclick="deleteArticle(<?php echo $article['article_id']; ?>, '<?php echo
h($article['title']); ?>')">
        <i class="fas fa-trash"></i>
    </button>
    <?php if ($article['is_published']): ?>
        <a href="../customer/article.php?slug=<?php echo h($article['slug']); ?>">
            <span class="btn btn-sm btn-outline-info" target="_blank">
                <i class="fas fa-eye"></i>
            </span>
        <?php endif; ?>
    </td>
</tr>
<?php endforeach; ?>
</tbody>
</table>
</div>
<?php endif; ?>
</div>

```

```

    </div>
</div>

<!-- Create Article Modal --&gt;
&lt;div class="modal fade" id="createArticleModal" tabindex="-1"&gt;
    &lt;div class="modal-dialog modal-lg"&gt;
        &lt;div class="modal-content"&gt;
            &lt;form method="POST" action=""&gt;
                &lt;input type="hidden" name="csrf_token" value="&lt;?php echo generateCSRFToken(); ?&gt;"&gt;

                &lt;div class="modal-header bg-hollywood text-white"&gt;
                    &lt;h5 class="modal-title"&gt;Create New Article&lt;/h5&gt;
                    &lt;button type="button" class="btn-close btn-close-white" data-bs-dismiss="modal"&gt;&lt;/button&gt;
                &lt;/div&gt;
                &lt;div class="modal-body"&gt;
                    &lt;div class="mb-3"&gt;
                        &lt;label for="create_title" class="form-label"&gt;Title *&lt;/label&gt;
                        &lt;input type="text" class="form-control" name="title" id="create_title" required&gt;
                    &lt;/div&gt;
                    &lt;div class="mb-3"&gt;
                        &lt;label for="create_category" class="form-label"&gt;Category *&lt;/label&gt;
                        &lt;select class="form-select" name="category" id="create_category" required&gt;
                            &lt;option value=""&gt;Select Category&lt;/option&gt;
                            &lt;option value="History"&gt;History&lt;/option&gt;
                            &lt;option value="Stars"&gt;Stars&lt;/option&gt;
                            &lt;option value="Movies"&gt;Movies&lt;/option&gt;
                            &lt;option value="Behind the Scenes"&gt;Behind the Scenes&lt;/option&gt;
                            &lt;option value="Industry News"&gt;Industry News&lt;/option&gt;
                            &lt;option value="Trivia"&gt;Trivia&lt;/option&gt;
                        &lt;/select&gt;
                    &lt;/div&gt;
                    &lt;div class="mb-3"&gt;
                        &lt;label for="create_content" class="form-label"&gt;Content *&lt;/label&gt;
                        &lt;textarea class="form-control" name="content" id="create_content" rows="10" required&gt;&lt;/textarea&gt;
                        &lt;small class="text-muted"&gt;You can use basic HTML tags for formatting&lt;/small&gt;
                    &lt;/div&gt;
                    &lt;div class="mb-3 form-check"&gt;
                        &lt;input type="checkbox" class="form-check-input" name="is_published" id="create_is_published" value="1"&gt;
                        &lt;label class="form-check-label" for="create_is_published"&gt;
                            Publish immediately
                        &lt;/label&gt;
                    &lt;/div&gt;
                &lt;/div&gt;
            &lt;/form&gt;
        &lt;/div&gt;
    &lt;/div&gt;
&lt;/div&gt;
</pre>

```

```

</div>
<div class="modal-footer">
<button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Cancel</button>
<button type="submit" name="create_article" class="btn btn-hollywood">
<i class="fas fa-save"></i> Create Article
</button>
</div>
</div>
</form>
</div>
</div>
</div>
</div>

<!-- Edit Article Modal --&gt;
&lt;div class="modal fade" id="editArticleModal" tabindex="-1"&gt;
&lt;div class="modal-dialog modal-lg"&gt;
&lt;div class="modal-content"&gt;
&lt;form method="POST" action=""&gt;
&lt;input type="hidden" name="csrf_token" value="&lt;?php echo generateCSRFToken(); ?&gt;"&gt;
&lt;input type="hidden" name="article_id" id="edit_article_id"&gt;

&lt;div class="modal-header bg-hollywood text-white"&gt;
&lt;h5 class="modal-title"&gt;Edit Article&lt;/h5&gt;
&lt;button type="button" class="btn-close btn-close-white" data-bs-dismiss="modal"&gt;&lt;/button&gt;
&lt;/div&gt;
&lt;div class="modal-body"&gt;
&lt;div class="mb-3"&gt;
&lt;label for="edit_title" class="form-label"&gt;Title *&lt;/label&gt;
&lt;input type="text" class="form-control" name="title" id="edit_title" required&gt;
&lt;/div&gt;
&lt;div class="mb-3"&gt;
&lt;label for="edit_category" class="form-label"&gt;Category *&lt;/label&gt;
&lt;select class="form-select" name="category" id="edit_category" required&gt;
&lt;option value=""&gt;Select Category&lt;/option&gt;
&lt;option value="History"&gt;History&lt;/option&gt;
&lt;option value="Stars"&gt;Stars&lt;/option&gt;
&lt;option value="Movies"&gt;Movies&lt;/option&gt;
&lt;option value="Behind the Scenes"&gt;Behind the Scenes&lt;/option&gt;
&lt;option value="Industry News"&gt;Industry News&lt;/option&gt;
&lt;option value="Trivia"&gt;Trivia&lt;/option&gt;
&lt;/select&gt;
&lt;/div&gt;
&lt;div class="mb-3"&gt;
&lt;label for="edit_content" class="form-label"&gt;Content *&lt;/label&gt;
&lt;textarea class="form-control" name="content" id="edit_content"&gt;
</pre>

```

```

        rows="10" required></textarea>
<small class="text-muted">You can use basic HTML tags for formatting</small>
</div>
<div class="mb-3 form-check">
<input type="checkbox" class="form-check-input" name="is_published"
       id="edit_is_published" value="1">
<label class="form-check-label" for="edit_is_published">
    Published
</label>
</div>
</div>
<div class="modal-footer">
<button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Cancel</button>
<button type="submit" name="update_article" class="btn btn-hollywood">
<i class="fas fa-save"></i> Update Article
</button>
</div>
</form>
</div>
</div>
</div>
</div>

<!-- Delete Confirmation Modal -->
<div class="modal fade" id="deleteArticleModal" tabindex="-1">
    <div class="modal-dialog">
        <div class="modal-content">
            <form method="POST" action="">
                <input type="hidden" name="csrf_token" value="<?php echo generateCSRFToken(); ?>">
                <input type="hidden" name="article_id" id="delete_article_id">

                <div class="modal-header bg-danger text-white">
                    <h5 class="modal-title">Confirm Deletion</h5>
                    <button type="button" class="btn-close btn-close-white" data-bs-dismiss="modal"></button>
                </div>
                <div class="modal-body">
                    <p>Are you sure you want to delete this article?</p>
                    <p class="text-muted" id="delete_article_title"></p>
                    <p class="text-danger"><strong>This action cannot be undone!</strong></p>
                </div>
                <div class="modal-footer">
                    <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Cancel</button>
                    <button type="submit" name="delete_article" class="btn btn-danger">
<i class="fas fa-trash"></i> Delete Article
                    </button>
                </div>
            </form>
        </div>
    </div>
</div>

```

```

        </div>
    </form>
</div>
</div>
</div>

<script>
function editArticle(article) {
    document.getElementById('edit_article_id').value = article.article_id;
    document.getElementById('edit_title').value = article.title;
    document.getElementById('edit_category').value = article.category;
    document.getElementById('edit_content').value = article.content;
    document.getElementById('edit_is_published').checked = article.is_published == 1;

    const modal = new bootstrap.Modal(document.getElementById('editArticleModal'));
    modal.show();
}

function deleteArticle(articleId, title) {
    document.getElementById('delete_article_id').value = articleId;
    document.getElementById('delete_article_title').textContent = title;

    const modal = new bootstrap.Modal(document.getElementById('deleteArticleModal'));
    modal.show();
}
</script>

```

<?php include '../includes/footer.php'; ?>

Inventory.php

```

<?php

require_once '../../config/config.php';
require_once '../../includes/session.php';

requireRole(ROLE_INVENTORY_MANAGER);

$page_title = 'Inventory Management';

$database = new Database();
$db = $database->getConnection();

require_once MODEL_PATH . 'Inventory.php';
require_once MODEL_PATH . 'Product.php';

```

```

$inventory = new Inventory($db);
$product_model = new Product($db);

if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['update_inventory'])) {
    if (!verifyCSRFToken($_POST['csrf_token'] ?? '')) {
        setFlashMessage('error', 'Invalid security token.');
    } else {
        $product_id = (int)$_POST['product_id'];
        $new_quantity = (int)$_POST['quantity'];

        if ($new_quantity < 0) {
            setFlashMessage('error', 'Quantity cannot be negative.');
        } else {
            $old_inventory = $inventory->getById($product_id);

            if ($inventory->updateQuantity($product_id, $new_quantity, getCurrentUserId())) {
                logAudit('UPDATE_INVENTORY', 'inventory', $product_id,
                    $old_inventory['quantity_available'], $new_quantity);
                setFlashMessage('success', 'Inventory updated successfully!');
            } else {
                setFlashMessage('error', 'Failed to update inventory.');
            }
        }
        redirect('public/employee/inventory.php');
    }
}

$all_inventory = $inventory->getAll();
$low_stock = $inventory->getLowStock();
$stats = $inventory->getStatistics();

include '../includes/header.php';
?>

<div class="container-fluid my-4">
    <h1 class="text-hollywood mb-4">đÝ“| Inventory Management</h1>

    <!-- Statistics -->
    <div class="row mb-4">
        <div class="col-md-3">
            <div class="dashboard-card">
                <h3><?php echo number_format($stats['total_products']); ?></h3>
                <p>Total Products</p>

```

```

</div>
</div>
<div class="col-md-3">
<div class="dashboard-card" style="border-left-color: #28a745;">
    <h3><?php echo number_format($stats['total_stock']); ?></h3>
    <p>Total Items in Stock</p>
</div>
</div>
<div class="col-md-3">
<div class="dashboard-card" style="border-left-color: #ffc107;">
    <h3><?php echo number_format($stats['low_stock_count']); ?></h3>
    <p>Low Stock Alerts</p>
</div>
</div>
<div class="col-md-3">
<div class="dashboard-card" style="border-left-color: #dc3545;">
    <h3><?php echo number_format($stats['out_of_stock_count']); ?></h3>
    <p>Out of Stock</p>
</div>
</div>
</div>

<!-- Low Stock Alerts -->
<?php if (!empty($low_stock)): ?>
<div class="card card-hollywood mb-4">
<div class="card-header card-header-hollywood">
    <i class="fas fa-exclamation-triangle"></i> Low Stock Alerts
</div>
<div class="card-body">
    <div class="table-responsive">
        <table class="table table-hover">
            <thead>
                <tr>
                    <th>Product</th>
                    <th>Category</th>
                    <th>SKU</th>
                    <th>Current Stock</th>
                    <th>Reorder Level</th>
                    <th>Action</th>
                </tr>
            </thead>
            <tbody>
                <?php foreach ($low_stock as $item): ?>
                <tr>

```

```

<td><strong><?php echo h($item['product_name']); ?></strong></td>
<td><?php echo h($item['category_name']); ?></td>
<td><code><?php echo h($item['sku']); ?></code></td>
<td>
    <span class="badge badge-stock-low">
        <?php echo $item['quantity_available']; ?>
    </span>
</td>
<td><?php echo $item['reorder_level']; ?></td>
<td>
    <button class="btn btn-sm btn-hollywood"
        onclick="showUpdateModal(<?php echo $item['product_id']; ?>, '<?php echo
h($item['product_name']); ?>', <?php echo $item['quantity_available']; ?>)">
        <i class="fas fa-edit"></i> Update
    </button>
</td>
</tr>
<?php endforeach; ?>
</tbody>
</table>
</div>
</div>
<?php endif; ?>

<!-- All Inventory -->
<div class="card card-hollywood">
<div class="card-header card-header-hollywood">
<i class="fas fa-boxes"></i> All Inventory
</div>
<div class="card-body">
<div class="table-responsive">
    <table class="table table-hover table-striped" id="inventoryTable">
        <thead class="table-dark">
            <tr>
                <th>Product</th>
                <th>Category</th>
                <th>SKU</th>
                <th>Price</th>
                <th>In Stock</th>
                <th>Reserved</th>
                <th>Reorder Level</th>
                <th>Status</th>
                <th>Action</th>
            

```

```

</tr>
</thead>
<tbody>
<?php foreach ($all_inventory as $item): ?>
    <tr>
        <td><strong><?php echo h($item['product_name']); ?></strong></td>
        <td><?php echo h($item['category_name']); ?></td>
        <td><code><?php echo h($item['sku']); ?></code></td>
        <td><?php echo formatCurrency($item['price']); ?></td>
        <td><strong><?php echo $item['quantity_available']; ?></strong></td>
        <td><?php echo $item['quantity_reserved']; ?></td>
        <td><?php echo $item['reorder_level']; ?></td>
        <td>
            <?php if ($item['stock_status'] === 'low'): ?>
                <span class="badge bg-danger">Low Stock</span>
            <?php elseif ($item['stock_status'] === 'medium'): ?>
                <span class="badge bg-warning text-dark">Medium</span>
            <?php else: ?>
                <span class="badge bg-success">Good</span>
            <?php endif; ?>
        </td>
        <td>
            <button class="btn btn-sm btn-outline-hollywood"
                onclick="showUpdateModal(<?php echo $item['product_id']; ?>, '<?php echo h($item['product_name']); ?>', <?php echo $item['quantity_available']; ?>)">
                <i class="fas fa-edit"></i>
            </button>
        </td>
    </tr>
<?php endforeach; ?>
</tbody>
</table>
</div>
</div>
</div>
</div>

<!-- Update Inventory Modal -->
<div class="modal fade" id="updateModal" tabindex="-1">
    <div class="modal-dialog">
        <div class="modal-content">
            <form method="POST" action="">
                <input type="hidden" name="csrf_token" value="<?php echo generateCSRFToken(); ?>">
                <input type="hidden" name="product_id" id="modal_product_id">

```

```

<div class="modal-header bg-hollywood text-white">
<h5 class="modal-title">Update Inventory</h5>
<button type="button" class="btn-close btn-close-white" data-bs-dismiss="modal"></button>
</div>
<div class="modal-body">
<h6 id="modal_product_name" class="mb-3"></h6>
<div class="mb-3">
<label for="current_quantity" class="form-label">Current Quantity</label>
<input type="number" class="form-control" id="current_quantity" readonly>
</div>
<div class="mb-3">
<label for="quantity" class="form-label">New Quantity *</label>
<input type="number" class="form-control" name="quantity" id="quantity"
min="0" required>
<small class="text-muted">Enter the new stock quantity</small>
</div>
</div>
<div class="modal-footer">
<button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Cancel</button>
<button type="submit" name="update_inventory" class="btn btn-hollywood">
<i class="fas fa-save"></i> Update Inventory
</button>
</div>
</form>
</div>
</div>
</div>

<script>
function showUpdateModal(productId, productName, currentQty) {
    document.getElementById('modal_product_id').value = productId;
    document.getElementById('modal_product_name').textContent = productName;
    document.getElementById('current_quantity').value = currentQty;
    document.getElementById('quantity').value = currentQty;

    const modal = new bootstrap.Modal(document.getElementById('updateModal'));
    modal.show();
}

</script>

<?php include '../includes/footer.php'; ?>

```

Order_view.php

```

<?php

require_once '../config/config.php';
require_once '../includes/session.php';

requireRole(ROLE_ORDER_FULFILLMENT);

$page_title = 'Order Details';

$order_id = (int)($_GET['id'] ?? 0);

if ($order_id === 0) {
    setFlashMessage('error', 'Invalid order ID.');
    redirect('public/employee/orders.php');
}

$database = new Database();
$db = $database->getConnection();

require_once MODEL_PATH . 'Order.php';
$order_model = new Order($db);

$order = $order_model->getById($order_id);

if (!$order) {
    setFlashMessage('error', 'Order not found.');
    redirect('public/employee/orders.php');
}

$items = $order_model->getOrderItems($order_id);

include '../includes/header.php';
?>

<div class="container my-4">
    <div class="d-flex justify-content-between align-items-center mb-4">
        <h1 class="text-hollywood">Order #<?php echo $order['order_id']; ?></h1>
        <a href="orders.php" class="btn btn-outline-hollywood">
            <i class="fas fa-arrow-left"></i> Back to Orders
        </a>
    </div>

    <div class="row">
        <!-- Order Information -->

```

```

<div class="col-md-6 mb-4">
<div class="card card-hollywood">
    <div class="card-header card-header-hollywood">
        <i class="fas fa-info-circle"></i> Order Information
    </div>
    <div class="card-body">
        <table class="table table-borderless">
            <tr>
                <td><strong>Order ID:</strong></td>
                <td>#<?php echo $order['order_id']; ?></td>
            </tr>
            <tr>
                <td><strong>Order Date:</strong></td>
                <td><?php echo formatDateTime($order['order_date']); ?></td>
            </tr>
            <tr>
                <td><strong>Status:</strong></td>
                <td>
                    <span class="badge status-<?php echo strtolower($order['status']); ?>">
                    <?php echo h($order['status']); ?>
                    </span>
                </td>
            </tr>
            <?php if (!empty($order['fulfilled_at'])): ?>
            <tr>
                <td><strong>Fulfilled Date:</strong></td>
                <td><?php echo formatDateTime($order['fulfilled_at']); ?></td>
            </tr>
            <?php endif; ?>
        </table>
    </div>
</div>
</div>

<!-- Customer Information -->
<div class="col-md-6 mb-4">
<div class="card card-hollywood">
    <div class="card-header card-header-hollywood">
        <i class="fas fa-user"></i> Customer Information
    </div>
    <div class="card-body">
        <table class="table table-borderless">
            <tr>
                <td><strong>Name:</strong></td>

```

```

<td><?php echo h($order['first_name'] . ' ' . $order['last_name']); ?></td>
</tr>
<tr>
    <td><strong>Email:</strong></td>
    <td><?php echo h($order['email']); ?></td>
</tr>
<tr>
    <td><strong>Phone:</strong></td>
    <td><?php echo h($order['phone']); ?></td>
</tr>
</table>
</div>

</div>
</div>
</div>

<!-- Shipping Address -->
<div class="card card-hollywood mb-4">
    <div class="card-header card-header-hollywood">
        <i class="fas fa-shipping-fast"></i> Shipping Address
    </div>
    <div class="card-body">
        <address>
            <?php echo h($order['shipping_address_line1']); ?><br>
            <?php if ($order['shipping_address_line2']): ?>
            <?php echo h($order['shipping_address_line2']); ?><br>
            <?php endif; ?>
            <?php echo h($order['shipping_city']); ?>,
            <?php echo h($order['shipping_state']); ?>
            <?php echo h($order['shipping_zip']); ?><br>
            United States
        </address>
    </div>
</div>

<!-- Order Items -->
<div class="card card-hollywood mb-4">
    <div class="card-header card-header-hollywood">
        <i class="fas fa-box"></i> Order Items
    </div>
    <div class="card-body">
        <div class="table-responsive">
            <table class="table table-hover">
                <thead class="table-light">

```

```

<tr>
    <th>Product</th>
    <th class="text-center">Quantity</th>
    <th class="text-end">Unit Price</th>
    <th class="text-end">Subtotal</th>
</tr>
</thead>
<tbody>
<?php foreach ($items as $item): ?>
    <tr>
        <td>
            <strong><?php echo h($item['product_name']); ?></strong><br>
            <small class="text-muted">SKU: <?php echo h($item['sku']); ?></small>
        </td>
        <td class="text-center"><?php echo $item['quantity']; ?></td>
        <td class="text-end"><?php echo formatCurrency($item['unit_price']); ?></td>
        <td class="text-end">
            <strong><?php echo formatCurrency($item['unit_price'] * $item['quantity']); ?></strong>
        </td>
    </tr>
<?php endforeach; ?>
</tbody>
<tfoot>
    <tr class="table-light">
        <td colspan="3" class="text-end"><h5>Total:</h5></td>
        <td class="text-end"><h5 class="text-hollywood"><?php echo
formatCurrency($order['total_amount']); ?></h5></td>
    </tr>
    </tfoot>
</table>
</div>
</div>
</div>

<!-- Action Buttons -->
<?php if ($order['status'] !== 'Delivered' && $order['status'] !== 'Cancelled'): ?>
<div class="d-flex gap-2 mb-4">
    <button class="btn btn-success" data-bs-toggle="modal" data-bs-target="#updateStatusModal">
        <i class="fas fa-edit"></i> Update Status
    </button>
    <button class="btn btn-info" onclick="window.print()">
        <i class="fas fa-print"></i> Print Order
    </button>
</div>

```

```

        <?php endif; ?>
</div>


<div class="modal fade" id="updateStatusModal" tabindex="-1">
    <div class="modal-dialog">
        <div class="modal-content">
            <form method="POST" action="orders.php">
                <input type="hidden" name="csrf_token" value="<?php echo generateCSRFToken(); ?>">
                <input type="hidden" name="order_id" value="<?php echo $order['order_id']; ?>">

                <div class="modal-header bg-hollywood text-white">
                    <h5 class="modal-title">Update Order Status</h5>
                    <button type="button" class="btn-close btn-close-white" data-bs-dismiss="modal"></button>
                </div>
                <div class="modal-body">
                    <div class="mb-3">
                        <label for="current_status" class="form-label">Current Status</label>
                        <input type="text" class="form-control" id="current_status"
                               value="<?php echo h($order['status']); ?>" readonly>
                    </div>
                    <div class="mb-3">
                        <label for="status" class="form-label">New Status *</label>
                        <select class="form-select" name="status" id="status" required>
                            <option value="">Select Status</option>
                            <option value="Pending">Pending</option>
                            <option value="Processing">Processing</option>
                            <option value="Shipped">Shipped</option>
                            <option value="Delivered">Delivered</option>
                            <option value="Cancelled">Cancelled</option>
                        </select>
                        <small class="text-muted">Select the new order status</small>
                    </div>
                </div>
                <div class="modal-footer">
                    <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Cancel</button>
                    <button type="submit" name="update_status" class="btn btn-hollywood">
                        <i class="fas fa-save"></i> Update Status
                    </button>
                </div>
            </form>
        </div>
    </div>
</div>

```

```

<?php include '../includes/footer.php'; ?>

Orders.php
<?php

require_once '../config/config.php';
require_once '../includes/session.php';

requireRole(ROLE_ORDER_FULFILLMENT);

$page_title = 'Order Fulfillment';

$database = new Database();
$db = $database->getConnection();

require_once MODEL_PATH . 'Order.php';
require_once MODEL_PATH . 'Customer.php';

$order_model = new Order($db);

if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['update_status'])) {
    if (!verifyCSRFToken($_POST['csrf_token'] ?? '')) {
        setFlashMessage('error', 'Invalid security token.');
    } else {
        $order_id = (int)$_POST['order_id'];
        $new_status = $_POST['status'];

        $emp_query = "SELECT employee_id FROM employees WHERE user_id = :user_id";
        $emp_stmt = $db->prepare($emp_query);
        $emp_stmt->execute([':user_id' => getCurrentUserId()]);
        $employee = $emp_stmt->fetch();

        $employee_id = $employee ? $employee['employee_id'] : null;

        if ($order_model->updateStatus($order_id, $new_status, $employee_id)) {
            logAudit('UPDATE_ORDER_STATUS', 'orders', $order_id, null, $new_status);
            setFlashMessage('success', 'Order status updated successfully!');
        } else {
            setFlashMessage('error', 'Failed to update order status.');
        }
        redirect('public/employee/orders.php');
    }
}

```

```

$filter_status = $_GET['status'] ?? null;

if ($filter_status) {
    $orders = $order_model->getAllOrders($filter_status);
} else {
    $orders = $order_model->getAllOrders();
}

$stats = $order_model->getStatistics();

include '../includes/header.php';
?>

<div class="container-fluid my-4">
    <h1 class="text-hollywood mb-4">Order Fulfillment</h1>

    <!-- Statistics -->
    <div class="row mb-4">
        <div class="col-md-3">
            <div class="dashboard-card">
                <h3><?php echo number_format($stats['total_orders']); ?></h3>
                <p>Total Orders</p>
            </div>
        </div>
        <div class="col-md-3">
            <div class="dashboard-card" style="border-left-color: #ffc107;">
                <h3><?php echo number_format($stats['pending_orders']); ?></h3>
                <p>Pending Orders</p>
            </div>
        </div>
        <div class="col-md-3">
            <div class="dashboard-card" style="border-left-color: #17a2b8;">
                <h3><?php echo number_format($stats['processing_orders']); ?></h3>
                <p>Processing</p>
            </div>
        </div>
        <div class="col-md-3">
            <div class="dashboard-card" style="border-left-color: #007bff;">
                <h3><?php echo number_format($stats['shipped_orders']); ?></h3>
                <p>Shipped</p>
            </div>
        </div>
    </div>

```

```

<!-- Filter Buttons -->
<div class="mb-4">
<a href="orders.php" class="btn <?php echo !$filter_status ? 'btn-hollywood' : 'btn-outline-hollywood'; ?>">
    All Orders
</a>
<a href="orders.php?status=Pending" class="btn <?php echo $filter_status === 'Pending' ? 'btn-hollywood' : 'btn-outline-hollywood'; ?>">
    Pending
</a>
<a href="orders.php?status=Processing" class="btn <?php echo $filter_status === 'Processing' ? 'btn-hollywood' : 'btn-outline-hollywood'; ?>">
    Processing
</a>
<a href="orders.php?status=Shipped" class="btn <?php echo $filter_status === 'Shipped' ? 'btn-hollywood' : 'btn-outline-hollywood'; ?>">
    Shipped
</a>
<a href="orders.php?status=Delivered" class="btn <?php echo $filter_status === 'Delivered' ? 'btn-hollywood' : 'btn-outline-hollywood'; ?>">
    Delivered
</a>
</div>

<!-- Orders Table -->
<div class="card card-hollywood">
<div class="card-header card-header-hollywood">
<i class="fas fa-shopping-cart"></i> Orders
<?php if ($filter_status): ?>
    - <?php echo h($filter_status); ?>
<?php endif; ?>
</div>
<div class="card-body">
<?php if (empty($orders)): ?>
    <div class="text-center py-5 text-muted">
        <i class="fas fa-inbox fa-3x mb-3"></i>
        <p>No orders found.</p>
    </div>
<?php else: ?>
    <div class="table-responsive">
        <table class="table table-hover">
            <thead class="table-light">
                <tr>

```

```

<th>Order #</th>
<th>Customer</th>
<th>Date</th>
<th>Total</th>
<th>Ship To</th>
<th>Status</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<?php foreach ($orders as $order): ?>
<tr>
<td><strong>#<?php echo $order['order_id']; ?></strong></td>
<td>
    <?php echo h($order['first_name'] . ' ' . $order['last_name']); ?><br>
    <small class="text-muted"><?php echo h($order['phone']); ?></small>
</td>
<td><?php echo formatDateTime($order['order_date']); ?></td>
<td><strong><?php echo formatCurrency($order['total_amount']); ?></strong></td>
<td>
    <?php echo h($order['shipping_city']); ?>,
    <?php echo h($order['shipping_state']); ?>
</td>
<td>
    <span class="badge status-<?php echo strtolower($order['status']); ?>">
        <?php echo h($order['status']); ?>
    </span>
</td>
<td>
    <button class="btn btn-sm btn-outline-hollywood"
        onclick="showOrderDetails(<?php echo $order['order_id']; ?>)">
        <i class="fas fa-eye"></i> View
    </button>
    <?php if ($order['status'] !== 'Delivered' && $order['status'] !== 'Cancelled'): ?>
        <button class="btn btn-sm btn-success"
            onclick="showUpdateModal(<?php echo $order['order_id']; ?>, '<?php
echo h($order['status']); ?>')">
            <i class="fas fa-edit"></i> Update
        </button>
        <?php endif; ?>
    </td>
</tr>
<?php endforeach; ?>
</tbody>

```

```

        </table>
        </div>
    <?php endif; ?>
</div>
</div>
</div>


<div class="modal fade" id="updateStatusModal" tabindex="-1">
    <div class="modal-dialog">
        <div class="modal-content">
            <form method="POST" action="">
                <input type="hidden" name="csrf_token" value="<?php echo generateCSRFToken(); ?>">
                <input type="hidden" name="order_id" id="modal_order_id">

                <div class="modal-header bg-hollywood text-white">
                    <h5 class="modal-title">Update Order Status</h5>
                    <button type="button" class="btn-close btn-close-white" data-bs-dismiss="modal"></button>
                </div>
                <div class="modal-body">
                    <div class="mb-3">
                        <label for="current_status" class="form-label">Current Status</label>
                        <input type="text" class="form-control" id="current_status" readonly>
                    </div>
                    <div class="mb-3">
                        <label for="status" class="form-label">New Status *</label>
                        <select class="form-select" name="status" id="status" required>
                            <option value="">Select Status</option>
                            <option value="Pending">Pending</option>
                            <option value="Processing">Processing</option>
                            <option value="Shipped">Shipped</option>
                            <option value="Delivered">Delivered</option>
                            <option value="Cancelled">Cancelled</option>
                        </select>
                    </div>
                </div>
                <div class="modal-footer">
                    <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Cancel</button>
                    <button type="submit" name="update_status" class="btn btn-hollywood">
                        <i class="fas fa-save"></i> Update Status
                    </button>
                </div>
            </form>
        </div>
    </div>

```

```

        </div>
    </div>

<!-- Order Details Modal (placeholder) --&gt;
&lt;div class="modal fade" id="orderDetailsModal" tabindex="-1"&gt;
    &lt;div class="modal-dialog modal-lg"&gt;
        &lt;div class="modal-content"&gt;
            &lt;div class="modal-header bg-hollywood text-white"&gt;
                &lt;h5 class="modal-title"&gt;Order Details&lt;/h5&gt;
                &lt;button type="button" class="btn-close btn-close-white" data-bs-dismiss="modal"&gt;&lt;/button&gt;
            &lt;/div&gt;
            &lt;div class="modal-body" id="orderDetailsContent"&gt;
                &lt;div class="text-center py-5"&gt;
                    &lt;div class="spinner-border text-hollywood" role="status"&gt;
                        &lt;span class="visually-hidden"&gt;Loading...&lt;/span&gt;
                    &lt;/div&gt;
                &lt;/div&gt;
            &lt;/div&gt;
            &lt;/div&gt;
            &lt;/div&gt;
        &lt;/div&gt;
    &lt;/div&gt;
&lt;/div&gt;

&lt;script&gt;
function showUpdateModal(orderId, currentStatus) {
    document.getElementById('modal_order_id').value = orderId;
    document.getElementById('current_status').value = currentStatus;
    document.getElementById('status').value = "";

    const modal = new bootstrap.Modal(document.getElementById('updateStatusModal'));
    modal.show();
}

function showOrderDetails(orderId) {
    window.location.href = 'order_view.php?id=' + orderId;
}
&lt;/script&gt;
</pre>

```

<?php include '../includes/footer.php'; ?>

Dashboard.php

<?php

```

require_once '../config/config.php';
require_once '../includes/session.php';

```

```

requireLogin();

$user_id = getCurrentUserId();
$role_id = getCurrentUserRole();

switch ($role_id) {
    case ROLE_CUSTOMER:
        redirect('public/customer/shop.php');
        break;

    case ROLE_INVENTORY_MANAGER:
        redirect('public/employee/inventory.php');
        break;

    case ROLE_ORDER_FULFILLMENT:
        redirect('public/employee/orders.php');
        break;

    case ROLE_CONTENT_EDITOR:
        redirect('public/employee/articles.php');
        break;

    case ROLE_BUSINESS_MANAGER:
    case ROLE_DB_ADMIN:
        redirect('public/admin/dashboard.php');
        break;

    default:
        logoutUser();
        redirect('public/login.php');
}

?>

```

Home.php

```

<?php

require_once './config/config.php';
require_once './includes/session.php';

$page_title = 'Welcome to StarNaVs LA';

$database = new Database();
$db = $database->getConnection();

```

```

require_once MODEL_PATH . 'Product.php';
require_once MODEL_PATH . 'Category.php';

$product_model = new Product($db);
$category_model = new Category($db);

$featured_products = $product_model->getAll(8);

$categories = $category_model->getAll();

include '../includes/header.php';
?>

<style>
.product-image-home {
    height: 200px;
    overflow: hidden;
    display: flex;
    align-items: center;
    justify-content: center;
    background: #f8f9fa;
}

.product-image-home img {
    width: 100%;
    height: 100%;
    object-fit: cover;
}
</style>

<!-- Hero Section -->
<div class="bg-dark text-white py-5" style="background: linear-gradient(135deg, #000000 0%, #1a1a1a 100%);">
    <div class="container">
        <div class="row align-items-center">
            <div class="col-lg-6">
                <h1 class="display-3 fw-bold mb-4">
                    <span class="text-hollywood">Shop the Charm.</span><br>
                    Live the Fame.
                </h1>
                <p class="lead mb-4">
                    Discover authentic Hollywood memorabilia, vintage collectibles, and exclusive merchandise
                    from the Golden Age of Cinema to modern blockbusters.
                </p>
            </div>
        </div>
    </div>

```

```

</p>
<div class="d-flex gap-3">
<a href="customer/shop.php" class="btn btn-hollywood btn-lg">
<i class="fas fa-store"></i> Shop Now
</a>
<a href="customer/hollywood101.php" class="btn btn-outline-light btn-lg">
<i class="fas fa-book"></i> Hollywood 101
</a>
</div>
</div>
<div class="col-lg-6 text-center">
<i class="fas fa-film" style="font-size: 15rem; color: #d4af37; opacity: 0.3;"></i>
</div>
</div>
</div>

<!-- Categories Section --&gt;
&lt;div class="container my-5"&gt;
&lt;h2 class="text-center text-hollywood mb-4"&gt;Browse by Category&lt;/h2&gt;
&lt;div class="row g-4"&gt;
&lt;?php foreach ($categories as $category): ?&gt;
&lt;div class="col-md-4 col-lg-2-4"&gt;
&lt;a href="customer/shop.php?category=&lt;?php echo $category['category_id']; ?&gt;"&gt;
&lt;div class="card card-hollywood text-center h-100 hover-card"&gt;
&lt;div class="card-body"&gt;
&lt;i class="fas fa-star fa-3x text-hollywood mb-3"&gt;&lt;/i&gt;
&lt;h5 class="card-title"&gt;&lt;?php echo h($category['category_name']); ?&gt;&lt;/h5&gt;
&lt;p class="card-text text-muted small"&gt;
&lt;?php echo h($category['description']); ?&gt;
&lt;/p&gt;
&lt;/div&gt;
&lt;/div&gt;
&lt;/a&gt;
&lt;/div&gt;
&lt;?php endforeach; ?&gt;
&lt;/div&gt;
&lt;/div&gt;

<!-- Featured Products Section --&gt;
&lt;div class="bg-light py-5"&gt;
&lt;div class="container"&gt;
&lt;h2 class="text-center text-hollywood mb-4"&gt;Featured Products&lt;/h2&gt;
</pre>

```

```

<div class="row g-4">
<?php foreach ($featured_products as $product): ?>
    <div class="col-md-6 col-lg-3">
        <div class="card card-hollywood h-100">
            <div class="product-image-home">
                <?php if ($product['image_url']): ?>
                    ">
                <?php else: ?>
                    <div class="text-center">
                        <i class="fas fa-image fa-4x text-muted"></i>
                    </div>
                <?php endif; ?>
            </div>
            <div class="card-body">
                <h5 class="card-title"><?php echo h($product['product_name']); ?></h5>
                <p class="card-text text-muted small">
                    <?php echo h(substr($product['description'], 0, 80)) . '...'; ?>
                </p>
                <div class="d-flex justify-content-between align-items-center">
                    <span class="text-hollywood h4 mb-0">
                        <?php echo formatCurrency($product['price']); ?>
                    </span>
                    <a href="customer/product_detail.php?id=<?php echo $product['product_id']; ?>" class="btn btn-sm btn-outline-hollywood">
                        View Details
                    </a>
                </div>
            </div>
        </div>
    </div>
<?php endforeach; ?>
</div>
<div class="text-center mt-4">
    <a href="customer/shop.php" class="btn btn-hollywood btn-lg">
        View All Products <i class="fas fa-arrow-right"></i>
    </a>
</div>
</div>
</div>

<!-- Features Section -->
<div class="container my-5">
    <div class="row g-4">

```

```

<div class="col-md-4">
<div class="text-center">
    <i class="fas fa-certificate fa-3x text-hollywood mb-3"></i>
    <h4>Authentic Collectibles</h4>
    <p class="text-muted">
        Genuine Hollywood memorabilia verified for authenticity
    </p>
</div>
</div>
<div class="col-md-4">
<div class="text-center">
    <i class="fas fa-shipping-fast fa-3x text-hollywood mb-3"></i>
    <h4>Fast Shipping</h4>
    <p class="text-muted">
        Quick and secure delivery across the United States
    </p>
</div>
</div>
<div class="col-md-4">
<div class="text-center">
    <i class="fas fa-star fa-3x text-hollywood mb-3"></i>
    <h4>Premium Quality</h4>
    <p class="text-muted">
        Carefully curated collection from the Golden Age to modern classics
    </p>
</div>
</div>
</div>
</div>

<!-- Hollywood 101 CTA -->
<div class="bg-hollywood text-white py-5">
    <div class="container text-center">
        <h2 class="mb-3">Explore Hollywood History</h2>
        <p class="lead mb-4">
            Dive into the fascinating world of cinema with our Hollywood 101 educational section
        </p>
        <a href="customer/hollywood101.php" class="btn btn-light btn-lg">
            <i class="fas fa-book"></i> Start Learning
        </a>
    </div>
</div>

<style>
```

```

.hover-card {
    transition: transform 0.3s ease, box-shadow 0.3s ease;
}

.hover-card:hover {
    transform: translateY(-5px);
    box-shadow: 0 8px 16px rgba(212, 175, 55, 0.3);
}

.col-lg-2-4 {
    flex: 0 0 auto;
    width: 20%;
}

@media (max-width: 991px) {
    .col-lg-2-4 {
        width: 33.333333%;
    }
}

@media (max-width: 767px) {
    .col-lg-2-4 {
        width: 50%;
    }
}

```

</style>

<?php include '../includes/footer.php'; ?>

Login.php

```

<?php

require_once '../config/config.php';
require_once '../includes/session.php';

if (isLoggedIn()) {
    redirect('public/dashboard.php');
}

$error = "";
$success = "";

if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    $username = trim($_POST['username'] ?? "");

```

```

$password = $_POST['password'] ?? '';
if (empty($username) || empty($password)) {
    $error = 'Please enter both username and password.';
} else {
    try {
        $database = new Database();
        $db = $database->getConnection();

        $query = "SELECT u.*, r.role_name
                  FROM users u
                  JOIN roles r ON u.role_id = r.role_id
                 WHERE (u.username = :username OR u.email = :email)
                   AND u.is_active = 1";
        $stmt = $db->prepare($query);
        $stmt->execute([':username' => $username, ':email' => $username]);
        $user = $stmt->fetch();

        if ($user && password_verify($password, $user['password_hash'])) {
            loginUser($user['user_id'], $user['username'], $user['email'], $user['role_id']);

            if (isset($_SESSION['return_url'])) {
                $redirect_url = $_SESSION['return_url'];
                unset($_SESSION['return_url']);
                redirect($redirect_url);
            } else {
                redirect('public/dashboard.php');
            }
        } else {
            $error = 'Invalid username or password.';
            logAudit('FAILED_LOGIN', 'users', null);
        }
    } catch (Exception $e) {
        error_log("Login error: " . $e->getMessage());
        $error = 'Login error: ' . $e->getMessage();
    }
}

$flash = getFlashMessage();
?>
<!DOCTYPE html>
<html lang="en">

```

```

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Login - <?php echo APP_NAME; ?></title>
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">
    <link rel="stylesheet" href="<?php echo ASSETS_URL; ?>css/style.css">
    <style>
        body {
            background: linear-gradient(135deg, #1a1a2e 0%, #16213e 100%);
            min-height: 100vh;
            display: flex;
            align-items: center;
            justify-content: center;
            font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
        }
        .login-container {
            background: white;
            border-radius: 15px;
            box-shadow: 0 10px 40px rgba(0,0,0,0.3);
            overflow: hidden;
            max-width: 900px;
            width: 100%;
        }
        .login-header {
            background: linear-gradient(135deg, #d4af37 0%, #c7a229 100%);
            color: white;
            padding: 40px;
            text-align: center;
        }
        .login-header h1 {
            margin: 0;
            font-size: 2.5rem;
            font-weight: bold;
            text-shadow: 2px 2px 4px rgba(0,0,0,0.2);
        }
        .login-header p {
            margin: 10px 0 0 0;
            font-size: 1.1rem;
            font-style: italic;
        }
        .login-body {
            padding: 40px;
        }
        .form-control:focus {

```

```

border-color: #d4af37;
box-shadow: 0 0 0 0.2rem rgba(212, 175, 55, 0.25);
}
.btn-hollywood {
background: linear-gradient(135deg, #d4af37 0%, #c7a229 100%);
border: none;
color: white;
padding: 12px;
font-size: 1.1rem;
font-weight: bold;
transition: transform 0.2s;
}
.btn-hollywood:hover {
background: linear-gradient(135deg, #c7a229 0%, #b89420 100%);
transform: translateY(-2px);
color: white;
}
.demo-credentials {
background: #f8f9fa;
padding: 20px;
border-radius: 8px;
margin-top: 20px;
}
.demo-credentials h6 {
color: #d4af37;
margin-bottom: 10px;
}
.demo-credentials small {
display: block;
margin-bottom: 5px;
}
</style>
</head>
<body>
<div class="container">
<div class="login-container">
<div class="login-header">
<h1>THP <?php echo APP_NAME; ?> THP</h1>
<p><?php echo APP_TAGLINE; ?></p>
</div>
<div class="login-body">
<?php if ($flash): ?>
<div class="alert alert-<?php echo h($flash['type']); ?> alert-dismissible fade show">
<?php echo h($flash['message']); ?>

```

```

<button type="button" class="btn-close" data-bs-dismiss="alert"></button>
</div>
<?php endif; ?>

<?php if ($error): ?>
<div class="alert alert-danger alert-dismissible fade show">
<?php echo h($error); ?>
<button type="button" class="btn-close" data-bs-dismiss="alert"></button>
</div>
<?php endif; ?>

<form method="POST" action="">
<div class="mb-3">
<label for="username" class="form-label">Username or Email</label>
<input type="text" class="form-control" id="username" name="username"
       value="<?php echo h($username ?? ""); ?>" required autofocus>
</div>
<div class="mb-3">
<label for="password" class="form-label">Password</label>
<div class="input-group">
    <input type="password" class="form-control" id="password" name="password"
           required>
    <button class="btn btn-outline-secondary" type="button" id="togglePassword"
           aria-label="Toggle password visibility">
        <i class="fas fa-eye" id="togglePasswordIcon"></i>
    </button>
</div>
</div>
<button type="submit" class="btn btn-hollywood w-100">
    ЁЯОм Login to StarNaVs LA
</button>
</form>

<div class="text-center mt-3">
<p class="text-muted mb-0">
<small>ЁЯУХ Los Angeles, California | Hollywood & Entertainment</small>
</p>
</div>
</div>
</div>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.4.0/css/all.min.css">

```

```

<script>
document.addEventListener('DOMContentLoaded', function() {
const togglePassword = document.getElementById('togglePassword');
const password = document.getElementById('password');
const toggleIcon = document.getElementById('togglePasswordIcon');

if (togglePassword) {
    togglePassword.addEventListener('click', function() {
        const type = password.getAttribute('type') === 'password' ? 'text' : 'password';
        password.setAttribute('type', type);

        if (type === 'password') {
            toggleIcon.classList.remove('fa-eye-slash');
            toggleIcon.classList.add('fa-eye');
        } else {
            toggleIcon.classList.remove('fa-eye');
            toggleIcon.classList.add('fa-eye-slash');
        }
    });
}
});

</script>
</body>
</html>

```

Logout.php

```

<?php

require_once '../config/config.php';
require_once '../includes/session.php';

logoutUser();

setFlashMessage('success', 'You have been successfully logged out.');
redirect('public/login.php');
?>

```

Register.php

```

<?php

require_once '../config/config.php';
require_once '../includes/session.php';

$page_title = 'Register';

```

```

if (isLoggedIn()) {
    redirect('public/dashboard.php');
}

$database = new Database();
$db = $database->getConnection();

require_once MODEL_PATH . 'User.php';
$user_model = new User($db);

if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    if (!verifyCSRFToken($_POST['csrf_token'] ?? '')) {
        setFlashMessage('error', 'Invalid security token.');
    } else {
        $username = trim($_POST['username']);
        $email = trim($_POST['email']);
        $password = $_POST['password'];
        $confirm_password = $_POST['confirm_password'];
        $first_name = trim($_POST['first_name']);
        $last_name = trim($_POST['last_name']);
        $phone = trim($_POST['phone']);

        $errors = [];

        if (empty($username) || strlen($username) < 3) {
            $errors[] = 'Username must be at least 3 characters.';
        }

        if (empty($email) || !filter_var($email, FILTER_VALIDATE_EMAIL)) {
            $errors[] = 'Valid email is required.';
        }

        if (empty($password) || strlen($password) < 6) {
            $errors[] = 'Password must be at least 6 characters.';
        }

        if ($password !== $confirm_password) {
            $errors[] = 'Passwords do not match.';
        }

        if (empty($first_name) || empty($last_name)) {
            $errors[] = 'First and last name are required.';
        }
    }
}

```

```

if (empty($phone)) {
    $errors[] = 'Phone number is required.';
}

if (empty($errors) && $user_model->getByUsername($username)) {
    $errors[] = 'Username already exists.';
}

if (empty($errors) && $user_model->getEmail($email)) {
    $errors[] = 'Email already registered.';
}

if (empty($errors)) {
    try {
        $db->beginTransaction();

        $password_hash = password_hash($password, PASSWORD_BCRYPT, ['cost' => 10]);
        $user_query = "INSERT INTO users (username, email, password_hash, role_id)
                      VALUES (:username, :email, :password, :role_id)";
        $user_stmt = $db->prepare($user_query);
        $user_stmt->execute([
            ':username' => $username,
            ':email' => $email,
            ':password' => $password_hash,
            ':role_id' => 6
        ]);
        $user_id = $db->lastInsertId();

        $customer_query = "INSERT INTO customers (user_id, first_name, last_name, phone)
                          VALUES (:uid, :fn, :ln, :phone)";
        $customer_stmt = $db->prepare($customer_query);
        $customer_stmt->execute([
            ':uid' => $user_id,
            ':fn' => $first_name,
            ':ln' => $last_name,
            ':phone' => $phone
        ]);

        $db->commit();

        logAudit('REGISTER', 'users', $user_id, null, $username);

        setFlashMessage('success', 'Registration successful! Please login.');
    }
}

```

```

        redirect('public/login.php');

    } catch (Exception $e) {
        $db->rollBack();
        error_log("Registration error: " . $e->getMessage());
        $errors[] = 'Registration failed: ' . $e->getMessage();
    }
}

if (!empty($errors)) {
    foreach ($errors as $error) {
        setFlashMessage('error', $error);
    }
}
}

include '../includes/header.php';
?>

<div class="container my-5">
    <div class="row justify-content-center">
        <div class="col-md-6">
            <div class="card card-hollywood">
                <div class="card-header card-header-hollywood text-center">
                    <h3><i class="fas fa-user-plus"></i> Create Account</h3>
                    <p class="mb-0">Join StarNaVs LA today</p>
                </div>
                <div class="card-body p-4">
                    <form method="POST" action="">
                        <input type="hidden" name="csrf_token" value="<?php echo generateCSRFToken(); ?>">

                        <h6 class="text-hollywood mb-3">Account Information</h6>

                        <div class="mb-3">
                            <label for="username" class="form-label">Username *</label>
                            <input type="text" class="form-control" name="username" id="username"
                                value="<?php echo h($_POST['username']); ?>">
                                minlength="3" required>
                        </div>

                        <div class="mb-3">
                            <label for="email" class="form-label">Email *</label>
                            <input type="email" class="form-control" name="email" id="email">
                        </div>
                </div>

```

```

        value="<?php echo h($_POST['email']) ?? ""; ?>" required>
    </div>

<div class="row">
    <div class="col-md-6 mb-3">
        <label for="password" class="form-label">Password *</label>
        <div class="input-group">
            <input type="password" class="form-control" name="password"
                id="password" minlength="6" required>
            <button class="btn btn-outline-secondary" type="button" id="togglePassword"
                aria-label="Toggle password visibility">
                <i class="fas fa-eye" id="togglePasswordIcon"></i>
            </button>
        </div>
        <small class="text-muted">Min. 6 characters</small>
    </div>
    <div class="col-md-6 mb-3">
        <label for="confirm_password" class="form-label">Confirm Password *</label>
        <div class="input-group">
            <input type="password" class="form-control" name="confirm_password"
                id="confirm_password" minlength="6" required>
            <button class="btn btn-outline-secondary" type="button" id="toggleConfirmPassword"
                aria-label="Toggle password visibility">
                <i class="fas fa-eye" id="toggleConfirmPasswordIcon"></i>
            </button>
        </div>
    </div>
</div>

<h6 class="text-hollywood mb-3 mt-4">Personal Information</h6>

<div class="row">
    <div class="col-md-6 mb-3">
        <label for="first_name" class="form-label">First Name *</label>
        <input type="text" class="form-control" name="first_name"
            id="first_name"
            value="<?php echo h($_POST['first_name']) ?? ""; ?>" required>
    </div>
    <div class="col-md-6 mb-3">
        <label for="last_name" class="form-label">Last Name *</label>
        <input type="text" class="form-control" name="last_name"
            id="last_name"
            value="<?php echo h($_POST['last_name']) ?? ""; ?>" required>
    </div>

```

```

</div>

<div class="mb-3">
    <label for="phone" class="form-label">Phone *</label>
    <input type="tel" class="form-control" name="phone" id="phone"
    value=<?php echo h($_POST['phone']); ?>" required>
</div>

<button type="submit" class="btn btn-hollywood w-100 mb-3">
    <i class="fas fa-user-plus"></i> Create Account
</button>

<div class="text-center">
    <p class="mb-0">
        Already have an account?
        <a href="login.php" class="text-hollywood">Login here</a>
    </p>
</div>
</form>
</div>
</div>
</div>
</script>
document.addEventListener('DOMContentLoaded', function() {
    const togglePassword = document.getElementById('togglePassword');
    const password = document.getElementById('password');
    const toggleIcon = document.getElementById('togglePasswordIcon');

    if (togglePassword && password && toggleIcon) {
        togglePassword.addEventListener('click', function(e) {
            e.preventDefault();
            const type = password.getAttribute('type') === 'password' ? 'text' : 'password';
            password.setAttribute('type', type);

            if (type === 'password') {
                toggleIcon.classList.remove('fa-eye-slash');
                toggleIcon.classList.add('fa-eye');
            } else {
                toggleIcon.classList.remove('fa-eye');
                toggleIcon.classList.add('fa-eye-slash');
            }
        });
    }
})

```

```

    });
}

const toggleConfirmPassword = document.getElementById('toggleConfirmPassword');
const confirmPassword = document.getElementById('confirm_password');
const toggleConfirmIcon = document.getElementById('toggleConfirmPasswordIcon');

if (toggleConfirmPassword && confirmPassword && toggleConfirmIcon) {
    toggleConfirmPassword.addEventListener('click', function(e) {
        e.preventDefault();
        const type = confirmPassword.getAttribute('type') === 'password' ? 'text' : 'password';
        confirmPassword.setAttribute('type', type);

        if (type === 'password') {
            toggleConfirmIcon.classList.remove('fa-eye-slash');
            toggleConfirmIcon.classList.add('fa-eye');
        } else {
            toggleConfirmIcon.classList.remove('fa-eye');
            toggleConfirmIcon.classList.add('fa-eye-slash');
        }
    });
}
});

```

</script>

<?php include '../includes/footer.php'; ?>

Session.php

```

<?php

ini_set('session.use_strict_mode', 1);
ini_set('session.cookie_httponly', 1);
ini_set('session.cookie_secure', 0);
ini_set('session.cookie_samesite', 'Strict');
ini_set('session.use_only_cookies', 1);

```

session_name(SESSION_NAME);

```

ini_set('session.gc_maxlifetime', SESSION_LIFETIME);
ini_set('session.cookie_lifetime', SESSION_LIFETIME);

```

```

if (session_status() === PHP_SESSION_NONE) {
    session_start();
}

```

```

if (!isset($_SESSION['created'])) {
    $_SESSION['created'] = time();
} elseif (time() - $_SESSION['created'] > 1800) {
    session_regenerate_id(true);
    $_SESSION['created'] = time();
}

if (isset($_SESSION['last_activity']) && (time() - $_SESSION['last_activity'] > SESSION_LIFETIME)) {
    session_unset();
    session_destroy();
    session_start();
    if (isLoggedIn()) {
        setFlashMessage('warning', 'Your session has expired. Please log in again.');
        redirect('login.php');
    }
}

$_SESSION['last_activity'] = time();

if (isset($_SESSION['user_id'])) {
    if (!isset($_SESSION['ip_address'])) {
        $_SESSION['ip_address'] = $_SERVER['REMOTE_ADDR'];
    } elseif ($_SESSION['ip_address'] !== $_SERVER['REMOTE_ADDR']) {
        error_log("IP address changed for user " . $_SESSION['user_id']);
    }

    if (!isset($_SESSION['user_agent'])) {
        $_SESSION['user_agent'] = $_SERVER['HTTP_USER_AGENT'];
    } elseif ($_SESSION['user_agent'] !== $_SERVER['HTTP_USER_AGENT']) {
        error_log("User agent changed for user " . $_SESSION['user_id']);
    }
}

function loginUser($user_id, $username, $email, $role_id) {
    session_regenerate_id(true);

    $_SESSION['user_id'] = $user_id;
    $_SESSION['username'] = $username;
    $_SESSION['email'] = $email;
    $_SESSION['role_id'] = $role_id;
    $_SESSION['ip_address'] = $_SERVER['REMOTE_ADDR'];
    $_SESSION['user_agent'] = $_SERVER['HTTP_USER_AGENT'];
    $_SESSION['created'] = time();
}

```

```

$_SESSION['last_activity'] = time();

try {
$database = new Database();
$db = $database->getConnection();

$query = "UPDATE users SET last_login = NOW() WHERE user_id = :user_id";
$stmt = $db->prepare($query);
$stmt->execute([':user_id' => $user_id]);

logAudit('LOGIN', 'users', $user_id);
} catch (Exception $e) {
error_log("Login update error: " . $e->getMessage());
}

}

function logoutUser() {
if (isset($_SESSION['user_id'])) {
logAudit('LOGOUT', 'users', $_SESSION['user_id']);
}

$_SESSION = array();

if (isset($_COOKIE[SESSION_NAME])) {
setcookie(SESSION_NAME, "", time() - 3600, '/');
}

session_destroy();
}

function requireLogin() {
if (!isLoggedIn()) {
$_SESSION['redirect_after_login'] = $_SERVER['REQUEST_URI'];
setFlashMessage('warning', 'Please log in to access this page.');
redirect('public/login.php');
exit();
}
}

function requireLoginForAction($action = 'perform this action') {
if (!isLoggedIn()) {
$_SESSION['return_url'] = $_SERVER['REQUEST_URI'];
setFlashMessage('error', "Please login to $action.");
redirect('public/login.php');
}
}

```

```

        exit();
    }
}

function requireRole($required_role_id) {
    requireLogin();

    if (!hasRole($required_role_id)) {
        setFlashMessage('error', 'You do not have permission to access this page.');

        $current_role = getCurrentUserRole();
        if ($current_role == ROLE_CUSTOMER) {
            redirect('public/customer/shop.php');
        } else {
            redirect('public/login.php');
        }
        exit();
    }
}

function requireCustomer() {
    requireLogin();

    if (!isCustomer()) {
        setFlashMessage('error', 'This page is only accessible to customers.');
        redirect('public/dashboard.php');
        exit();
    }
}

function requireEmployee() {
    requireLogin();

    if (!isEmployee() && !isManager()) {
        setFlashMessage('error', 'This page is only accessible to employees.');
        redirect('public/dashboard.php');
        exit();
    }
}

function requireManager() {
    requireLogin();

    if (!isManager()) {

```

```
setFlashMessage('error', 'This page is only accessible to managers and administrators.');
redirect('public/dashboard.php');
exit();
}
?>
```

Citations:

Books & Academic Sources

Elmasri, R., & Navathe, S. B. (2016). Fundamentals of Database Systems (7th ed.).

Pearson.

Software, Tools & Development Platforms

cPanel, LLC. (2024). cPanel User Guide. <https://docs.cpanel.net/>

phpMyAdmin Team. (2024). phpMyAdmin Documentation. <https://www.phpmyadmin.net/>

Draw.io. (2024). Entity–Relationship Diagramming Tool. <https://app.diagrams.net/>

Microsoft. (2024). Microsoft Word – Office Suite Documentation. <https://www.microsoft.com>

Googledoc Google Docs :<https://docs.google.com/>

Frontend Development Resources

Bootstrap Authors. (2024). Bootstrap Documentation. <https://getbootstrap.com/>

MDN Web Docs. (2024). HTML & CSS Documentation. <https://developer.mozilla.org/>

W3Schools. (2024). HTML Tutorial – Frontend Development. <https://www.w3schools.com/html/>