

# Feethub Documentation

**Stylish Steps, Seamless Shopping**

**DEVELOPED BY**

Rishika V

## INTRODUCTION

FeetHub is a full-stack e-commerce website built using **PHP and MySQL** for the backend and **HTML, CSS, JavaScript, and jQuery** for the frontend. It focuses on implementing CRUD operations, dynamic data rendering, and a role-based management system for multiple users. The project offers a real-time shopping experience where products added by merchants appear dynamically on the website's product pages.

## PROJECT OVERVIEW

The FeetHub backend system supports three dashboards — **Admin, Sub-admin, and Merchant** — each with distinct functionalities.

- The **Admin** can manage all users and create sub-admin accounts.
- The **Sub-admin** can manage both customers and merchants, including all products.
- The **Merchant** can manage their own products and view their customers' orders.

Products created by merchants are stored in the MySQL database and displayed dynamically on the website. This ensures seamless interaction between the backend and frontend, providing a complete shopping experience.

## TOOLS USED

- **Frontend:** HTML5, CSS3, JavaScript, jQuery
- **Backend:** PHP
- **Database:** MySQL
- **Editor:** Visual Studio Code
- **Server Environment:** XAMPP (Apache + MySQL)

## USER FLOW

### 1. Admin Dashboard

The Admin has full control of the application. They can view all users, create sub-admin accounts, and manage both merchants and customers. The Admin can also perform CRUD operations on all data, including product details.

## **2. Sub-admin Dashboard**

The Sub-admin acts as a support role to the Admin. They can manage customers, merchants, and all product-related information. The Sub-admin has permission to add, edit, or delete any record in the database.

## **3. Merchant Dashboard**

Each Merchant has their own dashboard. Merchants can add new products with details such as name, price, description, image, and category. They can also update or delete their products as needed. Products added by merchants are automatically displayed on the main website.

## **4. Frontend Integration**

The website dynamically fetches product data from the MySQL database using PHP.

When a customer clicks on a product, the site retrieves and displays its details dynamically using the product ID.

## **5. Add to Cart and Checkout Flow**

Clicking “Add to Cart” opens a modal popup showing product details, total amount, and quantity controls. The total updates automatically when the quantity changes, using jQuery logic. After clicking “Buy Now,” the user is redirected to the checkout form, where jQuery validation ensures all customer details are correctly filled before submission.

## **6. My Orders Page**

Customers can access their order history and tracking information by clicking the box icon in the navigation bar. This section displays the order’s progress (Placed, Packed, Shipped, Delivered) along with the product details and payment information.

## **7. Add to Cart Flow (Backend Integration)**

The Add to Cart feature is integrated with the backend using PHP and MySQL. When a user clicks the “Add to Cart” button, the selected product, along with the user ID and quantity, is stored in the cart table of the database.

Users can view, modify, or remove products from their cart. The total amount updates dynamically, ensuring accurate calculation before proceeding to checkout. Once the order is placed, the cart data is transferred to the orders table, and the user’s cart is cleared automatically.

## **8. Profile Edit Flow**

Each registered user has access to their profile section, where they can view and update personal details such as name, email, phone number, and address. The data is fetched dynamically from the database using the user ID. When the user submits updated information, it is validated and stored back into the user\_details table through PHP.

This flow enhances personalization and ensures users can manage their data securely without admin assistance.

## CRUD OPERATIONS

- **Create:** Merchants add new products into the system with complete details.
- **Read:** The website fetches all product data from the database and displays it dynamically.
- **Update:** Merchants or Admins can modify product or user data.
- **Delete:** Unnecessary or outdated products can be removed from the database.

## FEATURES

- Three-level user system (Admin, Sub-admin, Merchant)
- Session-based authentication for each user
- CRUD functionality for products and users
- Real-time data display from MySQL to frontend
- Product details loaded dynamically using product\_id
- Add to Cart backend integration with real-time updates
- Profile edit feature for user personalization
- Interactive cart modal with dynamic total calculation
- jQuery validation on checkout form
- Order tracking and management system
- Responsive, clean, and scalable architecture

## PAGE URLs

### ADMIN

- **Login Page:** <http://localhost/feethub-git/Feet-hub/admin/login.php>
- **Dashboard:** <http://localhost/feethub-git/Feet-hub/admin/admin-dashboard.php>

### SUB-ADMIN

- **Login Page:** <http://localhost/feethub-git/Feet-hub/admin/login.php>
- **Dashboard:** <http://localhost/feethub-git/Feet-hub/sub-admin/sub-admin.php>

### MERCHANT

- **Register Page:** <http://localhost/feethub-git/Feet-hub/ register.php>
- **Login Page:** <http://localhost/feethub-git/Feet-hub/ login.php>
- **Dashboard:** <http://localhost/feethub-git/Feet-hub/merchant-dashboard /merchant-dashboard.php>

### CUSTOMER

- **Home Page:** <http://localhost/feethub-git/Feet-hub/index.php>
- **Register Page:** <http://localhost/feethub-git/Feet-hub/ register.php>
- **Login Page:** <http://localhost/feethub-git/Feet-hub/login.php>

## CREDENTIALS

### ADMIN

- **User ID:** admin
- **Password:** admin@123

### SUB-ADMIN

- **Mail ID:** rishikav30@gmail.com
- **Password:** Rishi@303

### MERCHANT

- **Mail ID:** navi05@gmail.com
- **Password:** Navivaish@505

### CUSTOMER

- **Mail ID:** rishikav@gmail.com
- **Password:** asdfASDF@123

## CONCLUSION

FeetHub's backend version demonstrates a complete integration of frontend and backend technologies. Using PHP and MySQL, the system performs all CRUD operations while maintaining a structured, multi-user management flow.

The addition of **Add to Cart Flow** and **Profile Edit Flow** further enhances user engagement, offering personalized experiences and seamless purchasing. The dynamic connection between merchant dashboards and the customer-facing website ensures real-time updates and a smooth user experience.

This project showcases a strong foundation for a professional-level e-commerce platform with both design and development perspectives.

## PROJECT LINKS

Design Link: [FeetHub](#)

Git Repository Link: [FeetHub](#)