Rio Fish Website Documentation

Introduction

The Rio Fish website is a comprehensive platform designed to connect customers, fish traders, and administrators in a dynamic marketplace. Built on a robust LAMP stack, the website facilitates the registration, ordering, and management of fish-related transactions with a focus on region-specific access and pricing.

Technology Stack

- **Platform:** Digital Ocean
- **IP Address:** 67.205.134.57
- **Technology Stack:** LAMP (Linux, Apache, MySQL, PHP)
- **Access Method:** SSH through Terminal / IDE like Visual Studio

Accessing the Codebase

To access the codebase, developers can SSH into the server using the provided IP address. The codebase resides at *'/var/www/147.182.209.71'* in the server. "*root*" is the username for the webserver access. By using reset root password in Digital Ocean site, the root password can be changed. For secure access, it's recommended to generate an SSH key pair and add the public key to the SSH Keys in the Digital Ocean account settings (Settings > Security). Detailed explanation of how to achieve it is present here: https://docs.digitalocean.com/products/droplets/howto/add-ssh-keys/.

Website Functionality

General

Home Page (index.html): Serves as the landing page, redirecting to *login.php*.

Login/Registration: Supports registration for customers or fish traders, with separate admin access.

Customer-Specific

Home Page: Brief introduction to Rio Fish and navigation to all customer functionalities.

My Account: View or edit personal information.

Make an Order: Access the available fish catalog, add selections to the cart, and proceed to checkout.

My Orders: Review current and past orders, with details on specific orders.

Fish Trader-Specific

Fish Requests: View, accept, and process orders from customers within their region.

Reports: Generate reports on processed or currently processing orders.

Admin-Specific

Region and Product Management: Add or edit regions and fish products.

User Management: Edit user roles, including upgrading users to admins.

Payment Settlement: Handle payments to traders, with Rio Fish managing direct customer payments.

MPESA Integration

Customer to RioFish (Order Payment – includes service fee): Utilizes the MPESA payBill API.

RioFish to Traders (Settle Payment – service fee excluded): Utilizes the MPESA B2C API.

Implementation: Found in *function_utils.php* and associated *'mpesa'* prefixed files.

Database Structure

The Rio Fish website's database is designed to support the complexities of an online marketplace for fish trading. It stores user information, product details, orders, and transaction records. The relational structure ensures data integrity and facilitates complex queries for reporting and management purposes.

Connecting to the Database

- **Host:** Access through SSH to the server (IP: 67.205.134.57) using a terminal (explained in the beginning on how to do this)
- **Database Access Command:** *mysql -u riofish -p*
- **Username:** riofish
- **Password:** RioFish@2023

After logging in with the above credentials, you can interact with the database named RioFish. Once connected, you can interact with the database using SQL commands. For example, to select the database, use terminal command:

USE RioFish;

To list all tables, execute command:

SHOW TABLES;

And to view the structure of a specific table, for instance, *Users_New* table, use:

DESC Users_New;

This connection method allows developers and administrators to access the database for queries, updates, or management tasks directly.

Tables and Structures

Users_New

Purpose: Stores detailed user information including role and associated region.

Columns: *user_id, phone_number, first_name, last_name, password, role, region_id*.

Relationships:

- *region_id* links to *Regions.region_id* to denote the user's region.
- Acts as a foreign key reference for *Orders_New.customer_id* and *Orders_New.trader_id*.

Regions

Purpose: Defines the geographical regions available on the platform.

Columns: *region_id, region_name.*

Relationships:

- *region_id* is used across *Users_New*, *Fish_Prices*, and *Orders_New* to ensure regional specificity.

Fish_Prices

Purpose: Contains fish product details and prices, which can vary by region.

Columns: *fish_id, fish_name, fish_desc, rate, region_id.*

Relationships:

- *region_id* links to *Regions.region_id* to apply regional pricing.
- *fish_id* is referenced by *Order_Products.fish_id* for order details.

Orders_New

Purpose: Manages orders placed on the platform, including status and associated trader/customer.

Columns: *order_id, customer_id, trader_id, region_id, status_code, address_id.*

Relationships:

- *customer_id* and *trader_id* link to *Users_New.user_id*.
- *region_id* connects to *Regions.region_id.*
- *status_code* links to *Order_Statuses.status_code.*
- *address_id* references *Address.address_id*.

Order_Products

Purpose: Tracks products within each order, including quantities and prices.

Columns: *order_id, fish_id, ordered_quantity, unit_price.*

Relationships:

- *order_id* links to *Orders_New.order_id*.
- *fish_id* references *Fish_Prices.fish_id*.

Address

Purpose: Stores customer address details for orders.

Columns: *address_id, user_id, name, address_1, address_2, city, state, country, zip, phone_number.*

Relationships:

- *user_id* references *Users_New.user_id.*

Order_Statuses

Purpose: Defines the statuses of orders for customers and traders.

Columns: *status_code, status_customer, status_trader.*

Transactions

Purpose: Logs MPESA transactions from customers to Rio Fish.

Columns: *merchant_request_id, checkout_request_id, result_code, amount, new_balance, mpesa_receipt_number, transaction_date, phone_number, order_id, additional_info, is_settled*.

Relationships:

- *order_id* references *Orders_New.order_id*.

Inter-Table Relationships and Key Points

Region Specificity: *region_id* in *Users_New*, *Fish_Prices*, and *Orders_New* tables ensures that users, orders, and fish prices are managed per region.

User Roles and Orders: The *Orders_New* table connects customers and traders to their orders, with *customer_id* and *trader_id* linking back to *Users_New*.

Order Processing: The *Order_Products* table details the fish included in orders, linking *fish_id* to *Fish_Prices* for price reference and *order_id* to *Orders_New* for order details.