**User Documentation**

**Table of Contents**

1. Overview
2. System Requirements
3. Installation
4. Classes and Their Workings
5. User Registration
6. User Authentication and Roles
7. Buyer Functions
8. Seller Functions
9. Admin Functions
10. Troubleshooting

**1. Overview**

The E-Commerce platform is a Java-based application that simulates an online shopping system. It allows users to register and log in as buyers, sellers, or admins, each with specific functionalities. The platform integrates with a PostgreSQL database to manage user and product data. This overview will explain the key classes and their workings within the system.

**2. System Requirements**

* Java Development Kit (JDK) 8 or higher
* PostgreSQL Database
* An IDE or text editor (e.g., IntelliJ IDEA or VS Code)

**3. Installation**

**3.1 Database Setup**

1. Install PostgreSQL and set up your database.
2. Create the necessary tables by executing the following SQL commands:

A computer screen with text on it

Description automatically generatedA computer code with text on it

Description automatically generated

**3.2 Project Setup**

1. Clone the project repository or download the project files.
2. Open the project in your preferred IDE.
3. Configure your database connection details in the “DatabaseConnection” class.
4. Build and run the project.

**4. Classes and Their Workings**

**4.1. User Class**

The User class represents a user in the system. It includes attributes like id, username, password, email, and role. The role can be either buyer, seller, or admin, determining the functionalities available to the user.

**4.1.1. Buyer Class**

The Buyer class inherits from the User class and represents a user with the role of a buyer. Buyers have the ability to browse and view products in the system. Since this class is primarily for identifying the role and organizing functionality, it doesn’t require additional attributes beyond those inherited from User.

**4.1.2. Seller Class**

The Seller class also inherits from the User class and represents a user with the role of a seller. Sellers can add, update, and delete their own products. This class, like the Buyer class, primarily serves to distinguish the user type and ensure appropriate functionality is applied to them.

**4.1.3 Admin Class**

The Admin class inherits from the User class and represents a user with administrative privileges. Admins can manage users and view all products, along with seller information.

**4.2. UserDAO Class**

The UserDAO (Data Access Object) class handles database operations related to users. It includes methods to add a new user and retrieve a user by username.

**4.3. UserService Class**

The UserService class provides higher-level functionalities for user-related operations, such as user registration and authentication. It uses BCrypt for secure password hashing.

**4.4. Product Class**

The Product class represents a product in the system. It includes attributes like id, name, price, quantity, and sellerId.

**4.5. ProductDAO Class**

The ProductDAO class handles database operations related to products. It includes methods to add a new product, retrieve products by seller, and retrieve all products.

**4.6. ProductService Class**

The ProductService class provides higher-level functionalities for product-related operations, such as adding a product and retrieving products. It leverages the ProductDAO for database operations.

**4.7. ConsoleUI Class**

The ConsoleUI class serves as the entry point of the application. It provides a console-based user interface for interacting with the system. Users can register, log in, and perform role-specific functions.

**4.8. Class Diagram**

**A diagram of a computer

Description automatically generated with medium confidence**

**5. User Registration**

1. Run the application.
2. On the main menu, select - “Register”
3. Enter the required details:
   * Username
   * Password
   * Email
   * Role (buyer, seller, or admin)
4. Your account will be created, and you can now log in.

**6. User Authentication and Roles**

1. Run the application.
2. On the main menu, select - “Login”
3. Enter your username and password.
4. Depending on your role, you will be directed to your correct menu (buyer, seller, or admin).

**7. Buyer Functions**

**7.1 Viewing Products**

1. After logging in as a buyer, select – “View Products”.
2. A list of available products will be displayed with details such as name, price, and quantity.

**7.2 Logging Out**

1. To log out, select – “Logout”.

**8. Seller Functions**

**8.1 Adding a Product**

1. After logging in as a seller, select – “Add Product”.
2. Enter the product details:
   * Name
   * Price
   * Quantity
3. The product will be added to your list of products.

**8.2 Updating a Product**

1. After logging in as a seller, select – “Update Product”.
2. Enter the changes you want to make to the selected product.

**8.3 Deleting a Product**

1. After logging in as a seller, select – “Delete Product”.
2. Select the product you want to delete from the system.

**8.4 Viewing Your Products**

1. After logging in as a seller, select – “View My Products
2. A list of your products will be displayed with details such as name, price, and quantity.

**8.5 Logging Out**

1. To log out, select – “Logout”

**9. Admin Functions**

**9.1 Viewing Users**

1. After logging in as an admin, select – “View Users”.
2. A list of all registered users will be displayed with their contact information.

**9.2 Viewing Products**

1. After logging in as an admin, select – “View Products”.
2. A list of all products will be displayed with details such as name, price, quantity, and seller information.

**9.3 Logging Out**

1. To log out, select – “Logout”.

**10. Troubleshooting**

* **Database Connection Issues:**
  + Ensure PostgreSQL is running.
  + Check your connection details in the “DatabaseConnection” class.
* **User Registration Issues:**
  + Ensure your username is unique.
  + Check your email format.
* **Login Issues:**
  + Verify your username and password.
  + Ensure your role is correctly assigned during registration.

Development Documentation

The source code directory structure for this project is as follows: The project starts in ConsoleUI.java, where it will call upon UserService and ProductService which in turn will connect to UserDAO and ProductDAO respectively to do CRUD operations on both the users and products database tables. To compile this project you need to create a database with the create statements given in the “sql” folder. To get the source code from the GitHub repository, go to <https://github.com/KylePike1/Sprint2Java.git> click “Code” and copy the URL, you can then paste this URL into your GitHub desktop and press “Open in Visual Studio Code” to see the project in your Visual Studio Code.

Deployment Documentation

Java Final Sprint 2 Reagan Adams

This E-Commerce platform is a console-based Java application that simulates an online marketplace. Users can register as buyers, sellers, or admins and interact with the system through a text-based menu.

**User Roles:**

* **Buyers:** Can browse and search for products, view product details, and make purchase decisions.
* **Sellers:** Can list new products, update existing product information, and manage their inventory.
* **Admins:** Can manage users, view user details, and oversee all products listed on the platform.

**How to Use:**

1. **Starting the Application:**
   * Launch the program, and you will be greeted with a login or registration prompt.
   * New users can register by providing their details, including choosing a role (buyer, seller, or admin).
   * Existing users can log in to access their role-specific features.
2. **Navigating the Menu:**
   * After logging in, the menu will present options based on your role:
     + **Buyers:** Browse products, search for specific items, and view product details.
     + **Sellers:** Add new products, edit or remove listings, and view their product catalog.
     + **Admins:** Manage user accounts, view all users and products, and remove users if necessary.
3. **Performing Actions:**
   * Select menu options by entering the corresponding number or command.
   * Follow the on-screen prompts to complete your desired actions.