**Documentation for Console-Based E-Commerce Platform**

**1. User Documentation**

**Overview**

This application is a console-based e-commerce platform built using Java and PostgreSQL. It allows users to register as buyers, sellers, or admins and provides different functionalities based on their roles. Buyers can browse, search, and view products. Sellers can add, update, and delete their products. Admins can manage users and view all products along with seller information.

**Classes and Their Working**

1. **User**: Abstract class that serves as a base class for Buyer, Seller and Admin
2. **Buyer**: Inherits from User. Can browse, search, and view products.
3. **Seller**: Inherits from User. Can add, update, delete, and list their products.
4. **Admin**: Inherits from User. Can view all users, delete users, and view all products with seller information.
5. **Product**: Represents a product with attributes like id, name, price, quantity, sellerId, sellerName, and sellerEmail.
6. **UserDAO**: Data Access Object for performing CRUD operations on users.
7. **ProductDAO**: Data Access Object for performing CRUD operations on products.
8. **UserService**: Provides services for user-related operations.
9. **ProductService**: Provides services for product-related operations.
10. **Main**: Entry point of the application with the main method. Provides the user interface and handles user inputs.

**How to Start/Access the Application**

1. Ensure PostgreSQL is installed and running.
2. Create the necessary tables in PostgreSQL:

Sql code

CREATE TABLE Users (

id SERIAL PRIMARY KEY,

username VARCHAR(50) UNIQUE NOT NULL,

password VARCHAR(255) NOT NULL,

email VARCHAR(100) NOT NULL,

role VARCHAR(10) NOT NULL

);

CREATE TABLE Products (

id SERIAL PRIMARY KEY,

name VARCHAR(100) NOT NULL,

price DECIMAL(10, 2) NOT NULL,

quantity INT NOT NULL,

seller\_id INT NOT NULL,

FOREIGN KEY (seller\_id) REFERENCES Users(id)

);

1. Clone the repository from GitHub.
2. Compile and run the Main class.

**Class Diagram**

Buyer

User Seller

Admin

Product

UserDAO

ProductDAO

UserService

ProductService

Main

**2. Development Documentation**

**Javadocs**

Ensure all classes and methods have Javadocs comments explaining their functionality. Here’s an example:

java

Copy code

/\*\*

\* Represents a product with details such as name, price, quantity, seller ID, seller name, and seller email.

\*/

public class Product {

private int id;

private String name;

private double price;

private int quantity;

private int sellerId;

private String sellerName;

private String sellerEmail;

// Getters and Setters

}

**Source Code Directory Structure**

src/

├── main/

│ ├── java/

│ │ ├── Admin.java

│ │ ├── Buyer.java

│ │ ├── Main.java

│ │ ├── Product.java

│ │ ├── ProductDAO.java

│ │ ├── ProductService.java

│ │ ├── Seller.java

│ │ ├── User.java

│ │ ├── UserDAO.java

│ │ └── UserService.java

└── resources/

**Build Process**

To compile the project, navigate to the src directory and run:

Copy code

javac main/java/\*.java

To run the application, navigate to the src directory and run:

Copy code

java main/java/Main

**Compiler Time Dependencies**

* **PostgreSQL JDBC Driver**: Ensure the PostgreSQL JDBC driver is included in the classpath.

**Development Standards**

* Follow Java naming conventions.
* Ensure code readability and maintainability.
* Use meaningful variable and method names.
* Properly handle exceptions.

**Setting Up the Database for Development**

1. Ensure PostgreSQL is installed and running.
2. Create a database named ecommerce:

sql

Copy code

CREATE DATABASE ecommerce.

1. Create the necessary tables as mentioned earlier.

**Getting the Source Code from the Repository**

1. Clone the repository:

sh

Copy code

git clone <repository\_url>

1. Navigate to the project directory.

**3. Deployment Documentation**

**Installation Manual**

1. **Install Java Development Kit (JDK)**: Ensure JDK is installed. You can download it from [Oracle](https://www.oracle.com/java/technologies/javase-downloads.html).
2. **Install PostgreSQL**: Ensure PostgreSQL is installed and running. You can download it from [PostgreSQL](https://www.postgresql.org/download/).
3. **Set Up Database**: Create the ecommerce database and the necessary tables as mentioned earlier.
4. **Clone the Repository**: Clone the project repository from GitHub.
5. **Compile the Code**: Navigate to the src directory and compile the code:

Copy code

javac main/java/\*.java

1. **Run the Application**: Navigate to the src directory and run the application:

Copy code

java main/java/Main