K.RAMAKRISHNAN
COLLEGE OF TECHNOLOGY
An Autonomous Institution
Affiliated to Anna University Chennai, Approved by AICTE New Delhi,
ISO 9001:2015 & ISO 14001:2015 Certified Institution, Accredited with 'A+' grade by NAAC
Samayapuram, Tiruchirappalli – 621 112, Tamilnadu, India.

A Project Report

on

E COMMERCE PLATFORM

Submitted in partial fulfillment of requirements for the award of the course

of

CGB1201 - JAVA PROGRAMMING

Under the guidance of

Mrs. P. Jasmine Jose M.E.,

Assistant Professor / AI

Submitted By

RANJANI V (2303811714822037)

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY
(Autonomous)

SAMAYAPURAM 621 112

DECEMBER 2024

K.RAMAKRISHNAN
COLLEGE OF TECHNOLOGY
An Autonomous Institution
Affiliated to Anna University Chennai, Approved by AICTE New Delhi,
ISO 9001:2015 & ISO 14001:2015 Certified Institution, Accredited with 'A+' grade by NAAC
Samayapuram, Tiruchirappalli – 621 112, Tamilnadu, India.

K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY

SAMAYAPURAM 621 112

BONAFIDE CERTIFICATE

Certified that this project report on **"E COMMERCE PLATFORM IN JAVA"** is the Bonafide work of **RANJANI V (AM2337)** who carried out the project work during the academic year 2024 - 2025 under my supervision.

Signature	Signature
Mrs. P. JASMINE JOSE M.E.,	**Dr. T. AVUDAIAPPAN M.E.,Ph.D.,**
SUPERVISOR,	**HEAD OF THE DEPARTMENT,**
Department of Artificial Intelligence,	Department of Artificial Intelligence,
K. Ramakrishnan College of Technology,	K. Ramakrishnan College of Technology,
Samayapuram, Trichy -621 112.	Samayapuram, Trichy -621 112.

Submitted for the viva-voce examination held on 3.12.24

INTERNAL EXAMINER **EXTERNAL EXAMINER**

DECLARATION

I declare that the project report on "**E COMMERCE PLATFORM IN JAVA**" is the result of original work done by us and best of our knowledge, similar work has not been submitted to **"ANNA UNIVERSITY CHENNAI"** for the requirement of Degree of BACHELOR OF ENGINEERING. This project report is submitted on the partial fulfillment of the requirement of the award of the **CGB1201 – JAVA PROGRAMMING**.

Signature

(Ranjani V)

Place: Samayapuram

Date: 3/12/2024

ACKNOWLEDGEMENT

It is with great pride that I express our gratitude and indebtedness to our institution,**“K. Ramakrishnan College of Technology (Autonomous)”,** for providing us with the opportunity to do this project.

I extend our sincere acknowledgment and appreciation to the esteemed and honorable Chairman, **Dr. K. RAMAKRISHNAN, B.E.,** for having provided the facilities during the course of our study in college.

I would like to express our sincere thanks to our beloved Executive Director, **Dr. S. KUPPUSAMY, MBA, Ph.D.,** for forwarding our project and offering an adequate duration to complete it.

I would like to thank **Dr. N. VASUDEVAN, M.TECH., Ph.D.,** Principal,who gave the opportunity to frame the project to full satisfaction.

I thank **Dr. T. AVUDAIAPPAN, M.E., Ph.D.,** Head of the Department of **ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING**, for providing her encouragement in pursing this Project.

I wish to convey our profound and heartfelt gratitude to our esteemed project guide **Mrs. P. JASMINE JOSE M.E.,** Department of **ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING**, for her incalculable suggestions, creativity, assistance and patience, which motivated us to carry out this project.

I render our sincere thanks to the Course Coordinator and other staff members for providing valuable information during the course.

I wish to express our special thanks to the officials and Lab Technicians of our departments who rendered their help during the period of the work progress.

DEPARTMENT OF ARTIFICIAL INTELLIGENCE

VISION OF THE INSTITUTION

To serve the society by offering top-notch technical education on par with global standards.

MISSION OF THE INSTITUTION

- Be a centre of excellence for technical education in emerging technologies by exceeding the needs of industry and society.
- Be an institute with world class research facilities.
- Be an institute nurturing talent and enhancing competency of students to transform them as all- round personalities respecting moral and ethical values.

VISION AND MISSION OF THE DEPARTMENT

To excel in education, innovation and research in Artificial Intelligence and Machine Learning to fulfil industrial demands and societal expectations.

Mission 1: To educate future engineers with solid fundamentals, continually improving teaching methods using modern tools.

Mission 2: To collaborate with industry and offer top-notch facilities in a conductive learning environment.

Mission 3: To foster skilled engineers and ethical innovation in Artificial Intelligence and Machine Learning for global recognition and impactful research.

Mission 4: To tackle the societal challenge of producing capable professionals by instilling employability skills and human values.

PROGRAM EDUCATIONAL OBJECTIVES (PEOS)

PEO 1: Compete on a global scale for a professional career in Artificial Intelligence and Machine Learning

PEO 2: Provide industry-specific solutions for the society with effective communication and ethics.

PEO 3: Hone their professional skills through research and lifelong learning initiatives.

PROGRAM OUTCOMES

Engineering students will be able to:

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

11. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAM SPECIFIC OUTCOMES (PSOs)

- **PSO 1:** Capable of working on data-related methodologies and providing industry-focussed solutions.
- **PSO2:** Capable of analysing and providing a solution to a given real-world problem by designing an effective program.

ABSTRACT

Now a days the lifetime of the people is different. People feel painful and period uncontrollable for going crowded markets. So, E-Shopping could even be a boon because it saves lot of it slow. Online shopping could even be a process whereby consumers directly buy goods, services etc. from a seller without an intermediary service over the web. Shoppers can visit web stores from the comfort of their house and shop as by sitting before the pc. Online stores are usually available 24 hours day by day and will consumers have internet access both at work and reception. So, it's extremely convenient for them to buy for online. one in every of the foremost enticing factors about online shopping., particularly during season is, it alleviates the requirement to attend in long lines or search from a store for a selected item. style of goods are available in online. So, the researcher wants to understand the preference of the consumers. So, fifty defendants were met and data were together concerning their predilection towards supermarket run online.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 Objective

The primary objective of developing an E-Commerce Platform in Java is to provide a functional and extensible system that simulates the core features of an online shopping platform. The platform should cater to both customers and administrators by enabling.

- ✓ User Management
- ✓ Product Management
- ✓ Cart Management
- ✓ Order Processing
- ✓ Admin Features

1.2 Overview

Building an e-commerce platform involves creating a robust system to facilitate online buying and selling, with essential components like user management, product catalog, cart functionality, order processing, and payment integration. Java, known for its scalability, security, and strong community support, is an excellent choice for such a project.

- ➢ **User Management**
- ➢ **Product Catalog**
- ➢ **Shopping Cart and Wishlist**
- ➢ **Order Processing**
- ➢ **Payment Gateway Integration**
- ➢ **Admin Panel**

1.3 Java Programming Concepts

- ○ **Object-Oriented Programming (OOP)**

  - **Encapsulation**: Keep sensitive data secure and only accessible through methods (e.g., Product, User, Order classes).
  - **Inheritance**: Reuse code by creating base classes and extending them (e.g., User as a base class for Customer and Admin).

- **Design Patterns:**
  - **Factory Pattern**: Create objects like User or Order dynamically based on certain conditions.
  - **Builder Pattern**: Handle complex object creation, such as building an Order object with nested details.
- **Core Java Concepts:**
  - **Collections Framework**: Use List, Map, Set for managing product catalogs, shopping carts, and order history.
  - **Streams and Lambda Expressions**: Perform operations like filtering products, sorting by price, or aggregating cart totals efficiently.
- **Database Handling:**
  - **JDBC**: For basic database connectivity and SQL execution.
  - **Hibernate/JPA**: ORM tools for simplifying database interaction and mapping Java objects to relational tables.

CHAPTER 2

PROJECT METHODOLOGY

2.1Proposed Work

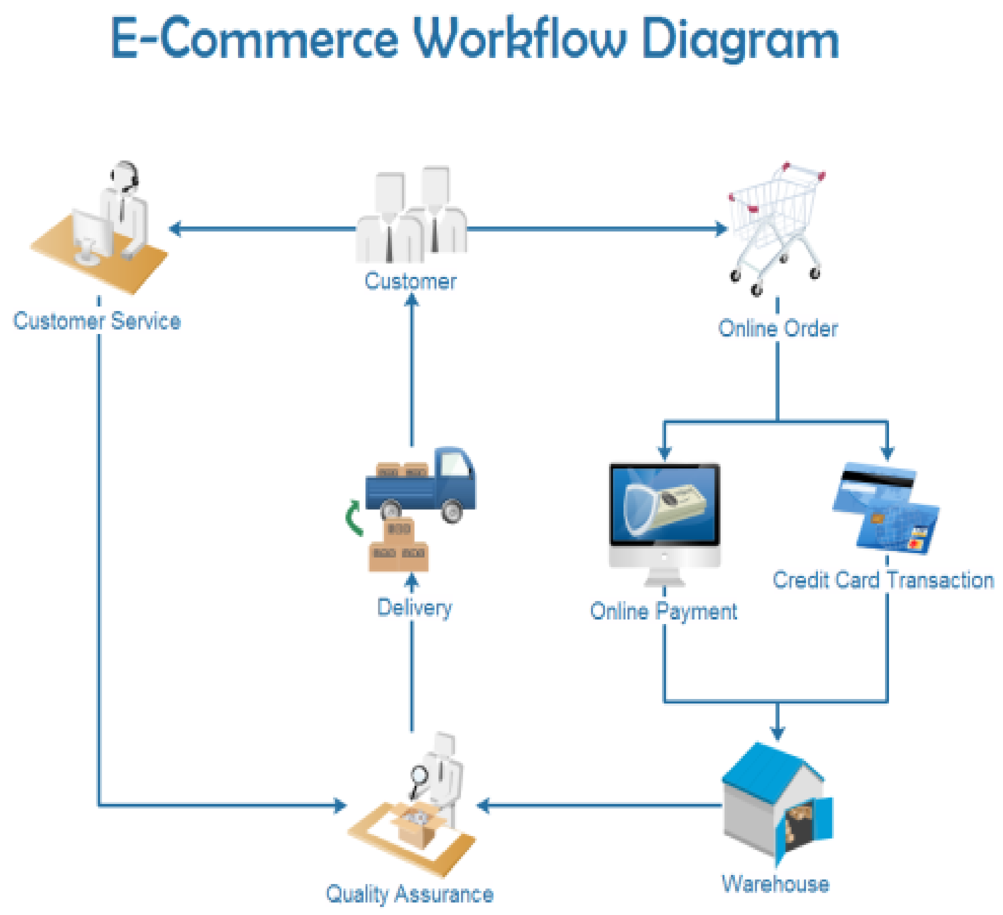
The proposed work for developing an e-commerce platform in Java will involve several stages, from planning and system design to implementation, testing, and deployment. Below is a detailed breakdown of the tasks and steps involved in this process:

1. Requirement Analysis and Planning

Objective: Define the functional and non-functional requirements of the e-commerce platform.

- **Functional Requirements**:
    - User Registration and Authentication (Admin and Customer roles).
    - Product Management (CRUD operations).
    - Shopping Cart Management.
    - Order Management and Checkout Process.
    - Payment Gateway Integration.
    - Admin Dashboard (for managing products, users, orders).
    - Customer Profile and Order History.
    - Email notifications for order updates.
- **Non-Functional Requirements**:
    - Scalability to handle increasing users and transactions.
    - Security (e.g., encryption, secure payment processing).
    - High Availability and Fault Tolerance.
    - Performance (fast loading times, efficient data retrieval).

2.2 Block Diagram



CHAPTER 3

MODULE DESCRIPTION

3.1 Module 1: User Management Module

The User Management Module handles all activities related to users on the platform. This includes:

1. **User Registration:** Allows users to create accounts with email verification for security.
2. **Authentication and Authorization:** Ensures secure login with features like multi-factor authentication and role-based access control (e.g., admin, customer).
3. **Profile Management:** Enables users to update their details such as name, email, **Profile Management:** phone number, and password.
4. **Password Recovery:** Includes a mechanism for resetting forgotten passwords through secure email or SMS-based verification.
5. **User Dashboard:** Offers personalized features like viewing order history, saving favourite products, and managing wish lists.

3.2 Module 2: Product Management Module

The Product Management Module facilitates the effective management of products displayed on the platform. This includes:

1. **Product Creation and Editing:** Enables admins to add new products, including attributes such as name, description, price, stock, and images.
2. **Category and Subcategory Management:** Organizes products into hierarchical categories for better browsing and filtering.
3. **Inventory Management:** Tracks product availability and alerts administrators about low-stock items.
4. **Product Review and Ratings:** Allows users to rate and review products, providing valuable feedback to administrators and other customers.
5. **Search and Filtering:** Implements advanced filtering options such as price range, brand, and features for easier product discovery.

3.3 Module 3: Shopping Cart Module

The Shopping Cart Module manages the items selected by the customer for purchase. This includes:

1. **Add to Cart:** Allows customers to add products to the cart with real-time price calculations, including discounts and taxes.
2. **Quantity Updates:** Users can update the quantity of items in the cart, which dynamically adjusts the total cost.
3. **Save for Later:** Provides an option for customers to save items in their cart for a future purchase.
4. **Cart Synchronization:** Synchronizes carts across devices, so users can access their cart from any logged-in device.
5. **Cart Summary**: Displays a detailed summary of all items, including subtotal, shipping costs, and taxes.

3.4 Module 4: Order Management Module

The Order Management Module tracks and processes all orders placed on the platform. This includes:

1. **Order Placement:** Captures customer orders with details like product information, delivery address, and payment method.
2. **Order Status Tracking:** Provides real-time updates on order status such as "Processing," "Shipped," "Out for Delivery," and "Delivered."
3. **Order Modification:** Allows users to cancel or modify orders (e.g., change delivery address) before shipping.
4. **Order History:** Maintains a record of all past orders for customer and admin reference.
5. **Return and Refund Management:** Handles customer requests for returns and refunds, ensuring seamless processing.

Module 5: Payment Integration Module

The Payment Integration Module ensures secure and efficient payment processing. This includes:

1. **Payment Gateway Integration:** Supports multiple payment options like credit/debit cards, net banking, UPI, wallets, and cash on delivery through trusted gateways (e.g., Stripe, PayPal).
2. **Secure Transactions:** Implements encryption and tokenization to protect sensitive payment data.
3. **Order Confirmation:** Provides instant payment confirmation to customers and updates the order status accordingly.
4. **Payment History:** Enables customers to view their payment history and download invoices.
5. **Fraud Prevention:** Uses advanced algorithms to detect and prevent fraudulent activities during payment.

CHAPTER 4

RESULTS AND DISCUSSION

Results

The proposed e-commerce platform is developed using a modular approach, where each module performs its specific functionality effectively. The results achieved from implementing the system include:

1. **Efficient User Management**:
   - The user registration and authentication system work seamlessly, providing a secure and user-friendly way for customers and administrators to access the platform.
   - Profile management and password recovery mechanisms function effectively, enhancing the overall user experience.
2. **Comprehensive Product Management**:
   - Admins can efficiently add, edit, and categorize products, while users can easily search and filter products based on categories, prices, and other attributes.
   - Real-time inventory updates ensure accurate stock tracking, reducing the risk of overselling.
3. **Streamlined Shopping Cart Functionality**:
   - Customers can add items to their cart, adjust quantities, and save items for later without any issues.
   - Cart synchronization across devices enables a seamless shopping experience, regardless of where the customer accesses their account.
4. **Robust Order Management**:
   - The order placement, modification, and tracking functionalities work as intended, providing customers with clear updates on their orders.
   - The return and refund processes are streamlined, ensuring customer satisfaction and reducing manual interventions for administrators.
5. **Secure Payment Integration**:
   - Payment gateway integration ensures fast and secure transactions, supporting a variety of payment methods.

- Fraud detection mechanisms provide additional security, while instant order confirmation builds customer trust in the platform.

Discussion

1. **System Efficiency**:

The modular design improves system efficiency and maintainability, as each module operates independently, allowing for easier updates and troubleshooting. This structure also facilitates scalability, enabling the platform to handle increasing traffic and larger product catalogs.

2. **User Experience**:

The implementation of personalized features like product recommendations, wishlists, and order history enhances the overall user experience. Features such as real-time inventory updates and cart synchronization further contribute to customer convenience.

3. **Security and Reliability**:

Using Spring Security for authentication and secure payment gateways ensures the platform is highly secure. Encrypted payment processing, coupled with fraud prevention algorithms, builds trust and minimizes risks for both customers and the business.

4. **Challenges Encountered**:

- **Performance Optimization**: Handling large-scale data in modules like product management and order management required optimization techniques to ensure fast response times.
- **Payment Gateway Integration**: Adhering to compliance standards such as PCI-DSS required additional development effort to ensure secure payment processing.
- **Scalability**: Ensuring the platform could handle a large number of simultaneous users required the implementation of caching and load-balancing techniques.

5. **Future Improvements**:

- Integration of advanced analytics tools for better insights into customer behavior and sales trends.
- Implementation of machine learning models for dynamic pricing and predictive inventory management.
- Development of a mobile application to complement the web platform and increase accessibility.

CHAPTER 5

CONCLUSION

The development of the proposed e-commerce platform demonstrates the effectiveness of a modular architecture in creating a robust, scalable, and user-friendly system. By dividing the platform into distinct modules User Management, Product Management, Shopping Cart, Order Management, and Payment Integration each component performs its functions efficiently while contributing to the overall system's reliability and performance.The platform successfully handles core functionalities, including user authentication, product organization, seamless shopping cart operations, order processing, and secure payment handling. The integration of real-time inventory updates, cart synchronization, and role-based access control enhances both the user and administrative experience. Furthermore, the adoption of security best practices, such as encrypted payment processing and fraud prevention mechanisms, ensures customer trust and data safety.Challenges encountered during the development, such as performance optimization and compliance with payment standards, were resolved through innovative solutions like caching, load balancing, and adherence to PCI-DSS standards. These efforts resulted in a highly secure and responsive platform.

REFERENCES

JOURNAL REFERENCES:

1. Chaffey, Dave. “Architecture of Next-Generation E-commerce Platforms.” *IEEE Xplore Digital Library*. IEEE, 2018. Available at IEEE Xplore.

2. Tiwana, Amrit. “Understanding How Platform Modularity Enhances Network Effects.” *SpringerLink*, 2019. Available at Springer.

3. “Building a Scalable E-Commerce Architecture with Microservices.” *Fabric Inc.*, 2022. Available at Fabric.

4. Varian, Hal. “Composable Architecture in E-commerce Agility.” *The Future of Commerce*. SAP Insights, 2023. Available at The Future of Commerce.

5. “System Design for E-commerce Websites.” *GeeksforGeeks*. 2021. Available at GeeksforGeeks.

WEB REFERENCES:

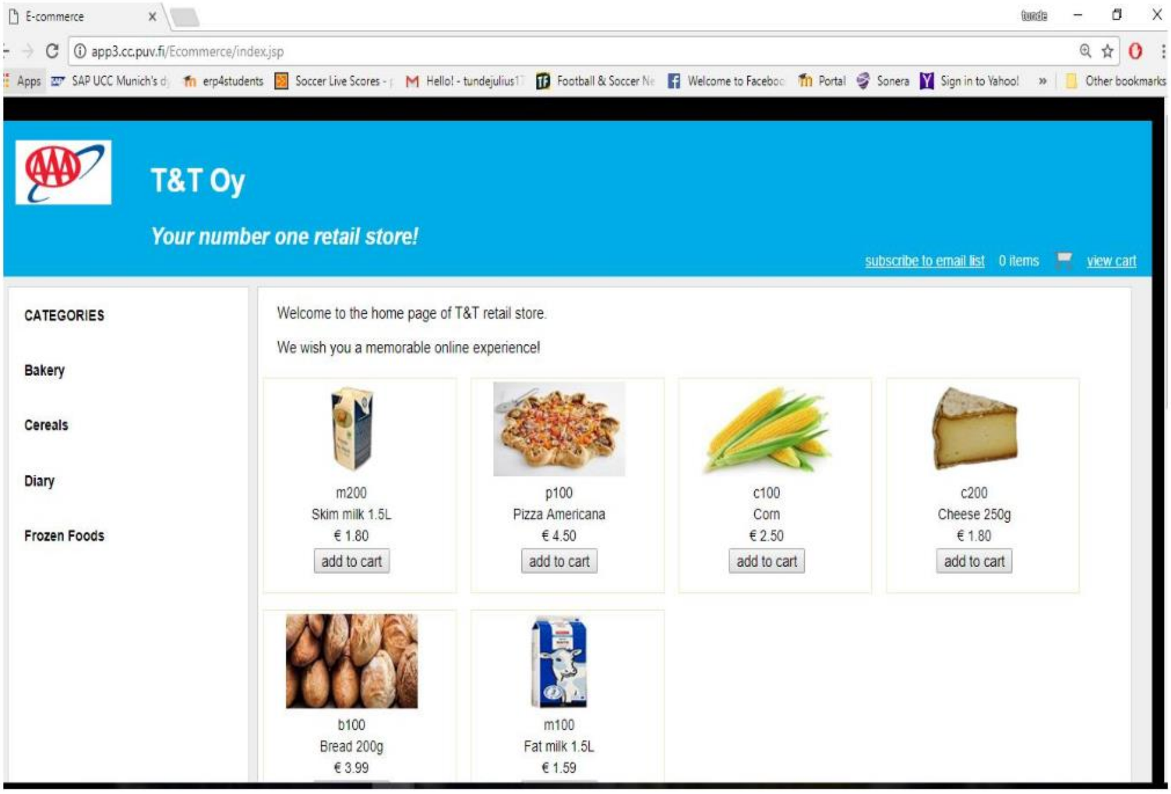
1. System Design for E-commerce Platforms - GeeksforGeeks
2. Building a Scalable E-Commerce Architecture with Microservices - Fabric Inc.
3. Composable E-commerce Explained - The Future of Commerce
4. SpringerLink: Modular Design and Platforms
5. Elastic Path: Composable Commerce Principles

APPENDIX

(SAMPLE CODING)

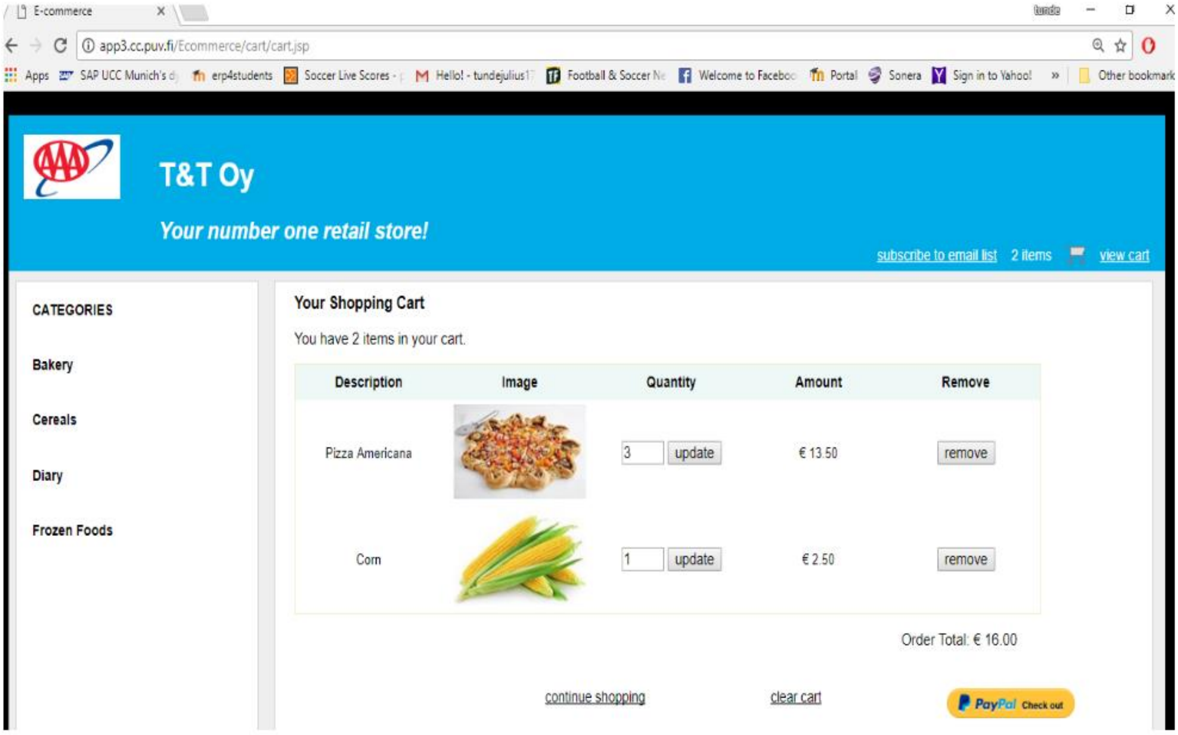
```
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

// Main Class
public class ECommercePlatform {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        UserManagement userManagement = new UserManagement();
        ProductManagement productManagement = new ProductManagement();
        OrderManagement orderManagement = new OrderManagement(productManagement);

        System.out.println("Welcome to the E-Commerce Platform!");

        boolean running = true;
        while (running) {
            System.out.println("\nMenu:");
            System.out.println("1. Register User");
            System.out.println("2. Login");
            System.out.println("3. Add Product (Admin)");
            System.out.println("4. View Products");
            System.out.println("5. Place Order");
            System.out.println("6. View Orders");
            System.out.println("7. Exit");

            System.out.print("Choose an option: ");
            int choice = scanner.nextInt();

            switch (choice) {
                case 1:
                    userManagement.registerUser();
                    break;
                case 2:
                    userManagement.login();
                    break;
                case 3:
                    productManagement.addProduct();
                    break;
                case 4:
                    productManagement.viewProducts();
                    break;
                case 5:
                    orderManagement.placeOrder(userManagement.getCurrentUser());
                    break;
                case 6:
                    orderManagement.viewOrders(userManagement.getCurrentUser());
                    break;
                case 7:
                    running = false;
                    System.out.println("Thank you for using the platform!");
                    break;
                default:
                    System.out.println("Invalid option. Try again.");
            }
        }

        scanner.close();
    }
}

// User Management Class
class UserManagement {
    private List<User> users = new ArrayList<>();
    private User currentUser;

    public void registerUser() {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter username: ");
        String username = scanner.nextLine();
        System.out.print("Enter password: ");
        String password = scanner.nextLine();

        users.add(new User(username, password));
        System.out.println("User registered successfully!");
    }

    public void login() {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter username: ");
        String username = scanner.nextLine();
        System.out.print("Enter password: ");
        String password = scanner.nextLine();

        for (User user : users) {
            if (user.getUsername().equals(username) && user.getPassword().equals(password)) {
                currentUser = user;
                System.out.println("Login successful!");
                return;
            }
        }
        System.out.println("Invalid credentials. Try again.");
    }

    public User getCurrentUser() {
        return currentUser;
    }
}

// User Class
class User {
    private String username;
    private String password;

    public User(String username, String password) {
        this.username = username;
        this.password = password;
    }

    public String getUsername() {
        return username;
    }

    public String getPassword() {
        return password;
    }
}

// Product Management Class
class ProductManagement {
    private List<Product> products = new ArrayList<>();

    public void addProduct() {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter product name: ");
        String name = scanner.nextLine();
        System.out.print("Enter product price: ");
        double price = scanner.nextDouble();

        products.add(new Product(name, price));
        System.out.println("Product added successfully!");
    }

    public void viewProducts() {
        System.out.println("\nAvailable Products:");
        for (int i = 0; i < products.size(); i++) {
            Product product = products.get(i);
            System.out.println((i + 1) + ". " + product.getName() + " - $" + product.getPrice());
        }
    }

    public List<Product> getProducts() {
        return products;
    }
}

// Product Class
class Product {
    private String name;
    private double price;

    public Product(String name, double price) {
        this.name = name;
        this.price = price;
    }

    public String getName() {
        return name;
    }
    public double getPrice() {
        return price;
    }
}

// Order Management Class
class OrderManagement {
    private List<Order> orders = new ArrayList<>();
    private ProductManagement productManagement;

    public OrderManagement(ProductManagement productManagement) {
        this.productManagement = productManagement;
    }

    public void placeOrder(User user) {
        if (user == null) {
            System.out.println("Please log in to place an order.");
            return;
        }

        productManagement.viewProducts();
        Scanner scanner = new Scanner(System.in);
        System.out.print("Select product number to order: ");
        int productIndex = scanner.nextInt() - 1;

        if (productIndex < 0 || productIndex >= productManagement.getProducts().size()) {
            System.out.println("Invalid product selection.");
            return;
        }

        Product selectedProduct = productManagement.getProducts().get(productIndex);
        orders.add(new Order(user, selectedProduct));
        System.out.println("Order placed successfully for " + selectedProduct.getName() + "!");
    }

    public void viewOrders(User user) {
        if (user == null) {
            System.out.println("Please log in to view orders.");
            return;
        }

        System.out.println("\nYour Orders:");
        for (Order order : orders) {
            if (order.getUser().equals(user)) {
                System.out.println("- " + order.getProduct().getName() + " - $" + order.getProduct().getPrice());
            }
        }
    }
}

// Order Class
class Order {
    private User user;
    private Product product;

    public Order(User user, Product product) {
        this.user = user;
        this.product = product;
    }

    public User getUser() {
        return user;
    }

    public Product getProduct() {
        return product;
    }
}
```

(SAMPLE OUTPUT)



Welcome to the home page of T&T retail store.

We wish you a memorable online experience!



T&T Oy

Your number one retail store!

subscribe to email list 2 items view cart

CATEGORIES

Bakery

Cereals

Diary

Frozen Foods

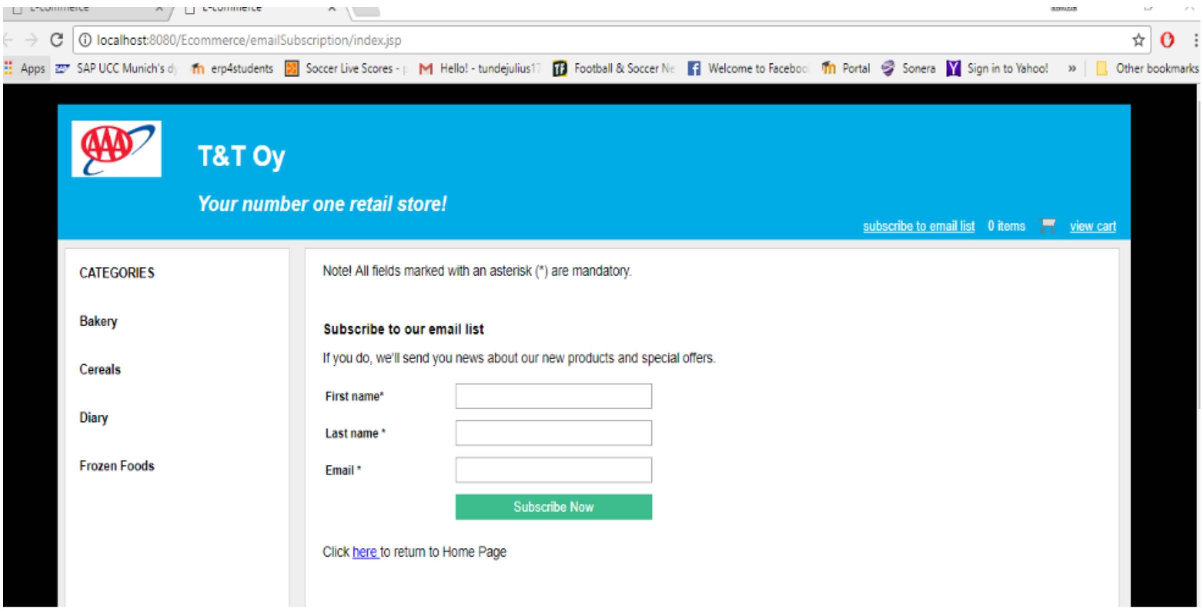
Your Shopping Cart

You have 2 items in your cart.

Description	Image	Quantity	Amount	Remove
Pizza Americana		3 update	€ 13.50	remove
Corn		1 update	€ 2.50	remove

Order Total: € 16.00

continue shopping clear cart



T&T Oy

Your number one retail store!

subscribe to email list 0 items view cart

CATEGORIES

Bakery

Cereals

Diary

Frozen Foods

Note! All fields marked with an asterisk (*) are mandatory.

Subscribe to our email list

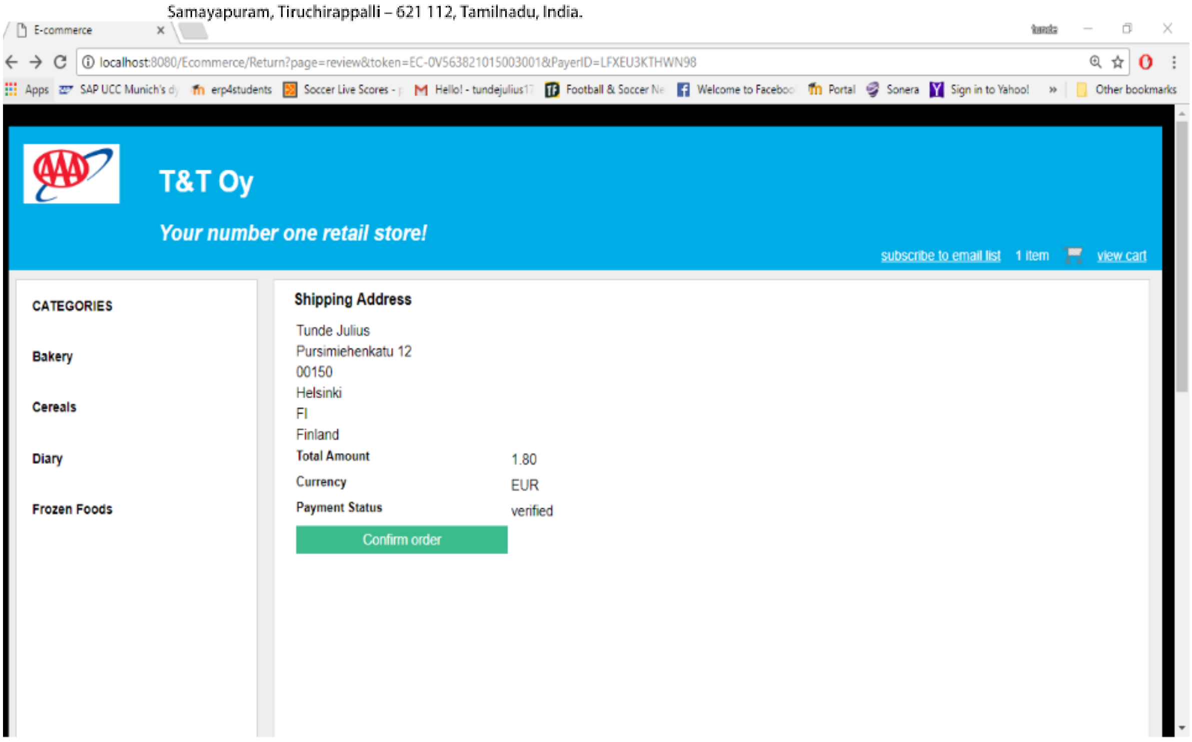
If you do, we'll send you news about our new products and special offers.

First name*

Last name *

Email *

Click here to return to Home Page



localhost:8080/Ecommerce/Return?page=review&token=EC-0V563821015003001&PayerID=LFXEU3KTHWN98

T&T Oy

Your number one retail store!

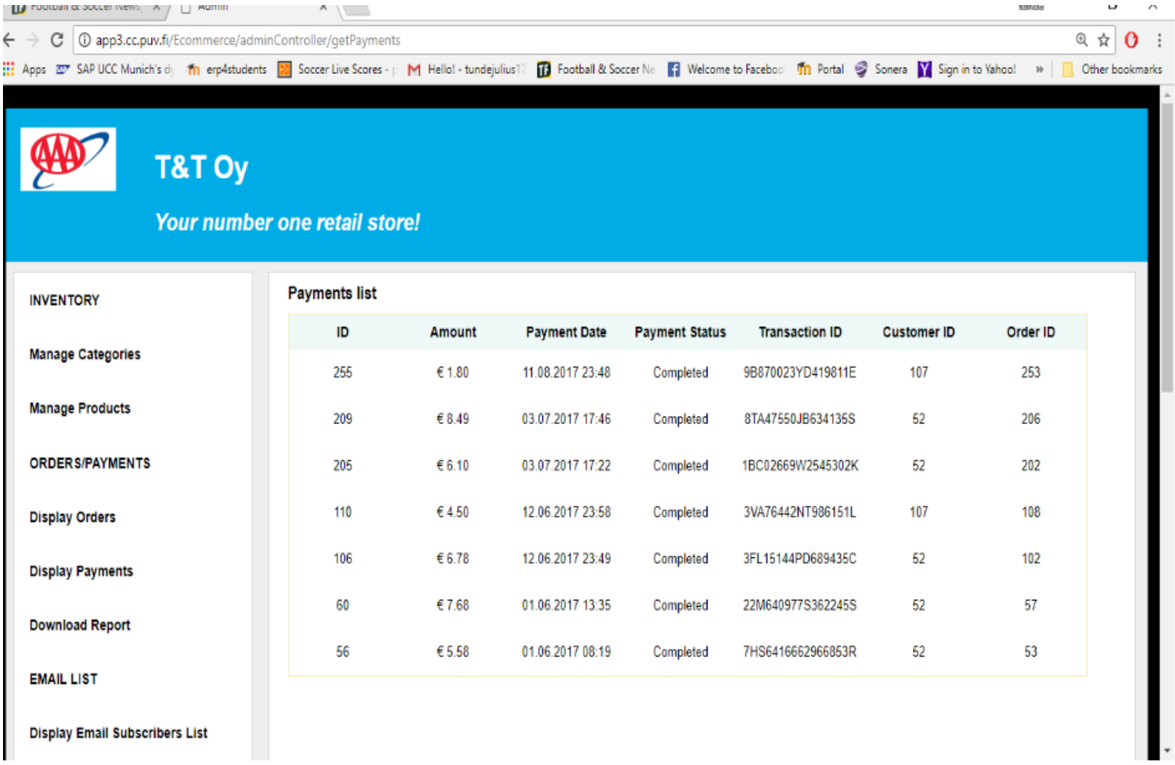
CATEGORIES

- Bakery
- Cereals
- Diary
- Frozen Foods

Shipping Address

Tunde Julius
Pursimiehenkatu 12
00150
Helsinki
FI
Finland

Total Amount	1.80
Currency	EUR
Payment Status	verified



app3.cc.puv.fi/Ecommerce/adminController/getPayments

T&T Oy

Your number one retail store!

INVENTORY

- Manage Categories
- Manage Products

ORDERS/PAYMENTS

- Display Orders
- Display Payments
- Download Report

EMAIL LIST

- Display Email Subscribers List

Payments list

ID	Amount	Payment Date	Payment Status	Transaction ID	Customer ID	Order ID
255	€ 1.80	11.08.2017 23:48	Completed	9B870023YD419811E	107	253
209	€ 8.49	03.07.2017 17:46	Completed	8TA47550JB634135S	52	206
205	€ 6.10	03.07.2017 17:22	Completed	1BC02669W2545302K	52	202
110	€ 4.50	12.06.2017 23:58	Completed	3VA76442NT986151L	107	108
106	€ 6.78	12.06.2017 23:49	Completed	3FL15144PD689435C	52	102
60	€ 7.68	01.06.2017 13:35	Completed	22M640977S362245S	52	57
56	€ 5.58	01.06.2017 08:19	Completed	7HS6416662966853R	52	53