**Online Shopping cart**

**CS23333-Object Oriented Programing using Java**

*Submitted by*

Mukesh.M-231001127

Mukesh.R-231001128

Mahtab Alam.A-231001510

*Of*

BACHELOR OF TECHNOLOGY

*In*

**INFORMATION TECHNOLOGY**

**RAJALAKSHMI ENGINEERING COLLEGE, THANDALAM (An**

**Autonomous Institution)**

**RAJALAKSHMI ENGINEERING COLLEGE**

**BONAFIDE CERTIFICATE**

Certified that this project titled “Online Shopping cart” is the bonafide work of “Mukesh.M-231001127, Mukesh.R-231001128,Mahtab Alam.A-231001510 ” who carried out the project work under my supervision.

SIGNATURE

Dr.Valarmathie

SIGNATURE

Mrs.Sangeetha Tupili

HEAD OF THE DEPARTMENT

Information Technology Rajalakshmi Engineering College, Rajalakshmi Nagar, Thandalam Chennai – 602105

This project is submitted for IT19341 –

Introduction to Oops and Java held on

COURSE INCHARGE

Information and Technology Rajalakshmi Engineering College Rajalakshmi Nagar, Thandalam Chennai – 602105

# INTERNAL EXAMINAR EXTERNAL EXAMINAR

**TABLE OF CONTENTS**

# Online Shopping cart:

* 1. [Abstract 5](#_TOC_250008)
  2. [Introduction 5](#_TOC_250007)
  3. [Purpose 5](#_TOC_250006)
  4. Scope of Project 6
  5. [Software Requirement Specification 6](#_TOC_250005)

1. [System Flow Diagrams 11](#_TOC_250004)
   1. Use Case Diagram 11
   2. Entity-relationship Diagrams 12
   3. [Data Flow Diagram 13](#_TOC_250003)
2. [Module Description 14](#_TOC_250002)
3. [Implementation](#_TOC_250001)
   1. [Design 15](#_TOC_250000)
   2. **Database Design 18**
   3. **Code 20**
4. **Conclusion 27**
5. **References 27**

# Abstract:

The Online Shopping cart is a comprehensive platform designed to streamline the purchase of electronic furnitureover-the-counter medications, and e-commerce devices. Leveraging technologies like Java for backend logic, MySQL for database management, and HTML/CSS/JavaScript for a user-friendly interface, The platform integrates secure authentication, prescription validation, real-time inventory updates, and seamless online payment gateways.the system accommodates e-commerce providers, pharmacies, and patients, ensuring a secure and efficient transaction experience.

# Introduc:tion

In today's digital age, e-commerce has revolutionized the e-commerce sector by enabling easy access to medical products. The Online Shopping cart automates traditional processes such as prescription validation, inventory management, and delivery tracking. With a focus on user-centric design, the system addresses challenges like maintaining data security and ensuring medication authenticity. By providing a convenient solution for patients to order medicines from the comfort of their homes, the system bridges the gap between pharmacies and consumers.

# Purpose:

The primary goal of this project is to create a robust, user-friendly platform that benefits both customers and pharmacy administrators. Objectives include:

**.** Simplifying the process of purchasing medical products online.

**.** Automating prescription validation to ensure legal compliance.

**.** Providing real-time inventory updates to customers.

**.** Enhancing business insights through data analytics.

# Scope of the Project:

The Online Shopping cart is intended for diverse users,including patients, pharmacy owners, and e-commerce providers.

**The system features:**

Secure user authentication for customers and administrators.Real-time tracking of orders and deliveries.Integration with third-party payment gateways for secure transactions.Compatibility with various devices, including desktops and mobile phones The project addresses the challenges of manual operations, such as stock discrepancies, data security risks, and slow order fulfillment, offering a modern, automated alternative.

# Software Requirement Specification:

**Introduction**:

This section outlines the essential hardware and software requirements needed to develop and deploy the Online Shopping cart, ensuring smooth operation, scalability, and user satisfaction. It defines the tools and platforms required for building a reliable and efficient solution.

# Document purpose:

The purpose of this document is to define the key functionalities,constraints, and technical requirements of the Online Shopping cart. It serves as a guideline for developers and stakeholders, ensuring clear understanding and effective implementation.

# Product Scope:

* + - Provide a seamless shopping experience for medical supplies.
    - Ensure legal compliance for prescription-based medicines.
    - Automate administrative tasks for pharmacies.

# Definitions, Acronyms, and Abbreviations:

OSC -Online Shopping cart.

# OverallDescription:

A medical e-commerce website using Java, JSP, and MySQL is an online platform where users can browse, purchase, and manage medical products such as medicines, medical equipment. The website would have features like user authentication, product listings and payment integration. Java is used for server-side logic, JSP for dynamic web pages, and MySQL for database management to handle products, customer data, and transactions.

# Product Perspective:

The medical e-commerce website is a user-friendly platform that provides access to a wide range of medical products, including prescription medications, over-the-counter drugs, and medical devices. The site supports personalized user accounts, order tracking, and product recommendations. It uses Java and JSP for backend processing and dynamic content generation. MySQL handles data storage for products, users, orders, and transactions efficiently.

# Product Functionality:

* **User Registration & Authentication:** Users can create accounts, log in, and manage profiles.
* **Product Search & Filtering:** Allows users to search and filter products based on categories (e.g., medications, medical equipment), brand, price, and health conditions.
* **Product Details & Reviews**: Provides detailed product information, including descriptions, usage instructions, and customer reviews/ratings.
* **Shopping Cart:** Users can add products to their cart, modify quantities, and view a summary of

selected items.

* **Order Placement & Payment Integration:** Users can proceed to checkout, select payment methods, and make secure online payments (credit card, e-wallets, etc.).
* **Admin Panel for Product Management:** Admins can add, update, or delete products, manage stock, and view sales data.

# User and Characteristics:

**Qualification:** Users should have at least basic education (e.g., matriculation) and be comfortable using the internet and English.

**Experience:** Familiarity with online shopping or e-commerce platforms is beneficial, though not mandatory.

**Technical Experience:** Users should possess basic computer skills, such as navigating websites, using a shopping cart, and making online payments.

## Operating Environment:

*Hardware Requirements:*

* Processor: Any Processor over i3
* Operating System: Windows 8, 10, 11
* Processor Speed: 2.0 GHz
* RAM: 4GB
* Hard Disk: 500GB

*Software Requirements:*

* Database: MySQL
* Frontend: JSP
* Technology: Java (JDBC)

*Constraints:*

* System access limited to administrators.
* Delete operation restricted to administrators without additional checks for simplicity.
* Administrators must exercise caution during deletion to maintain data consistency.

*Assumptions and Dependencies:*

* System administrators create and confidentially communicate login IDs and passwords to users.

# User Interface:

The Medical E-Commerce Store provides user-friendly, menu-driven interfaces for:

1. **Register:** Registering new user.
2. **Login:** Logging in existing user.
3. **Product pages:** with detailed information, images, prices.
4. **Shopping cart:** with visible item summaries and easy quantity adjustments.
5. **Add Quantity:** increase the quantity of the product.

g) **Order:** Viewing and Place order.

# Hardware Interface:

* Screen resolution of at least 640 x 480 or above.
* Compatible with any version of Windows 8, 10, 11.

## Software Interface:

* 1. MS-Windows Operating System .
  2. Jsp for designing the front end .
  3. java for the backend & MYSQL as database.
  4. Platform: Java Language.

# FunctionalRequirements:

## User Registration and Authentication:

Allow users to register with their details (name, email, password, address, etc.).

Provide secure login/logout functionality using encrypted passwords.

## Product Catalog Management:

Display a categorized list of medical products with filtering options (e.g., by category, price, or brand).

Show detailed product descriptions, images, prices, and stock availability.

## Shopping Cart and Checkout:

Allow users to add, update, or remove products in their shopping cart.

## Payment Gateway Integration:

Support multiple payment option (credit/debit cards).

Ensure secure transactions through encryption and fraud detection mechanisms.

## Admin Functions:

Allow admins to manage product inventory (add, update, delete products). Provide order management features, including viewing and updating order status.

## Reports and Analytics:

Generate sales and user activity reports for business insights. Provide analytics on product popularity, order trends, and revenue.

# Non-functional Requirements:

## Performance:

The website should load quickly (within 2-3 seconds) and handle a high volume of traffic without delays or crashes.

## Scalability:

It should support increasing numbers of users, transactions, and product listings without performance degradation.

## Security:

Ensure data protection through SSL encryption, secure payment processing, and compliance with privacy regulations (e.g., HIPAA).

## Availability:

The website should be available 99.9% of the time, with minimal downtime for maintenance or upgrades.

## Usability:

The site should be user-friendly, with an intuitive interface and smooth navigation for all user groups (patients, e-commerce providers, admins).

## Reliability:

The system must be stable and consistently deliver expected functionality, with minimal errors or disruptions.

## Compatibility:

The website should be compatible with major browsers (Chrome, Firefox, Safari, etc.) and devices (desktops, tablets, smartphones).

# System Flow Diagrams:

* 1. **Use Case Diagrams :**



Data Base

Login

Product

User

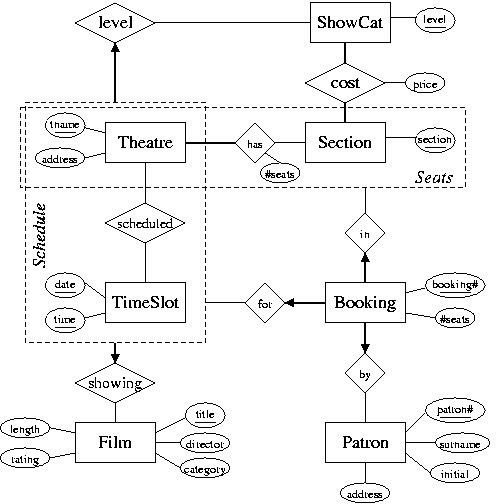
Cart

Order

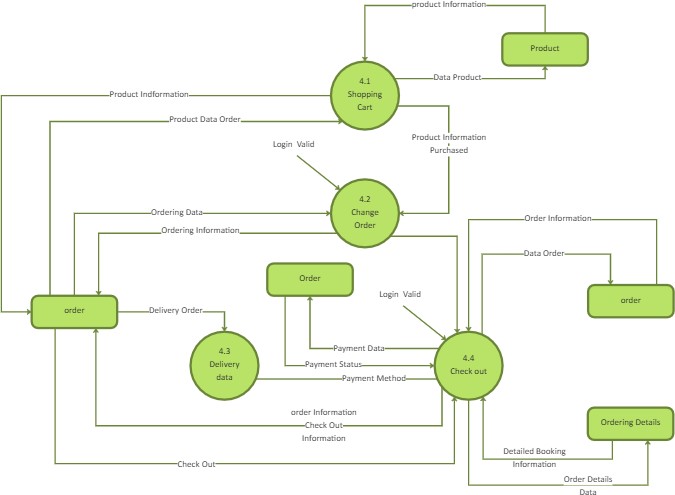
Payment

# Entity-relationship diagram:

E-R (Entity-Relationship) Diagram is used to represents the relationship between entities in the table.

**Entity -relationship diagram:**

# Data-flow diagram:



# Module description:

## .User Management :

Handles user registration, login, profile management, and authentication for customers and admins.

## Product Management :

Allows admins to add, update, delete, and categorize medical products; manage stock levels

and product details.

## Shopping Cart :

Manages the user's selected products, including quantity adjustments, view items, and proceed to checkout.

## Order Management :

Handles order placement,order history for users and admins.

## Payment & Checkout :

Integrates secure payment gateways for processing transactions, including credit cards, digital wallets, and other payment methods.

## Admin Dashboard Module:

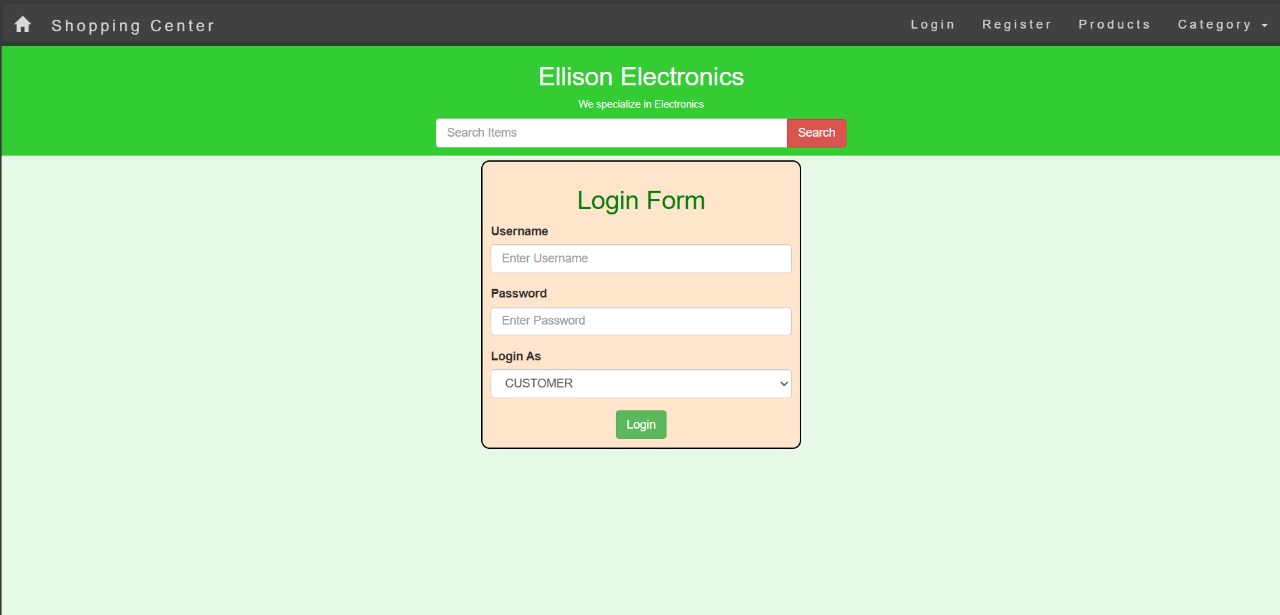
Provides an interface for admins to monitor sales, manage users, track inventory, and generate reports.

# Implementation:

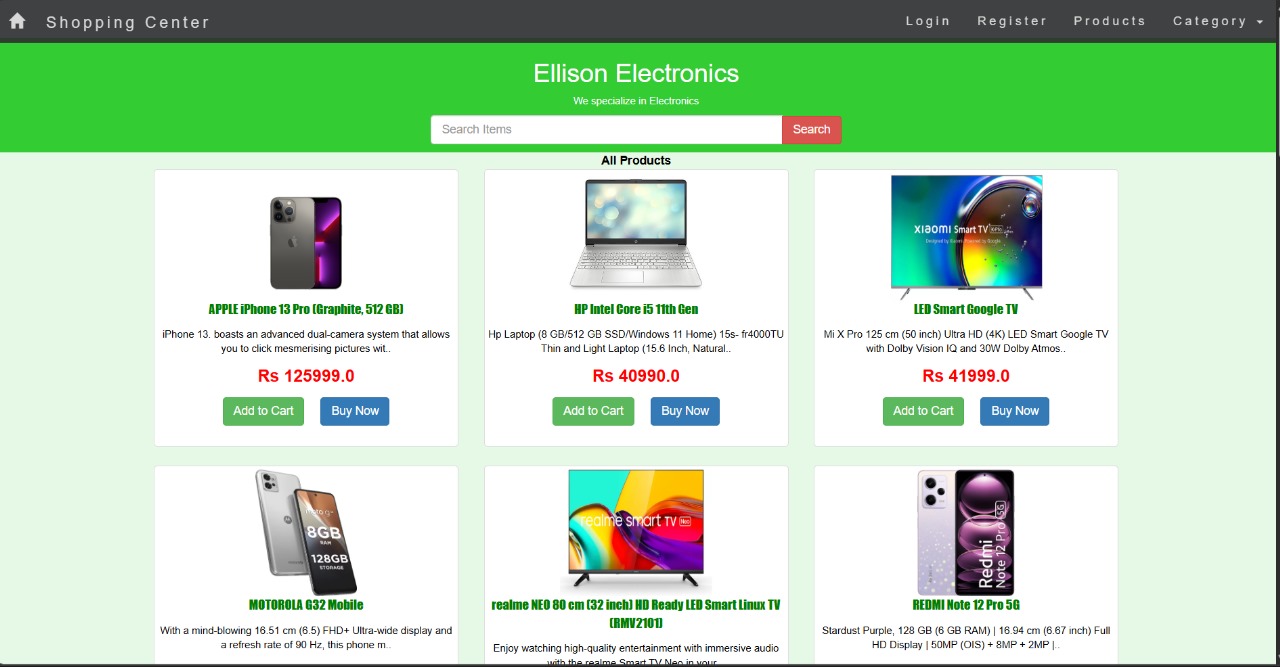
# Design:

**Login Page:**

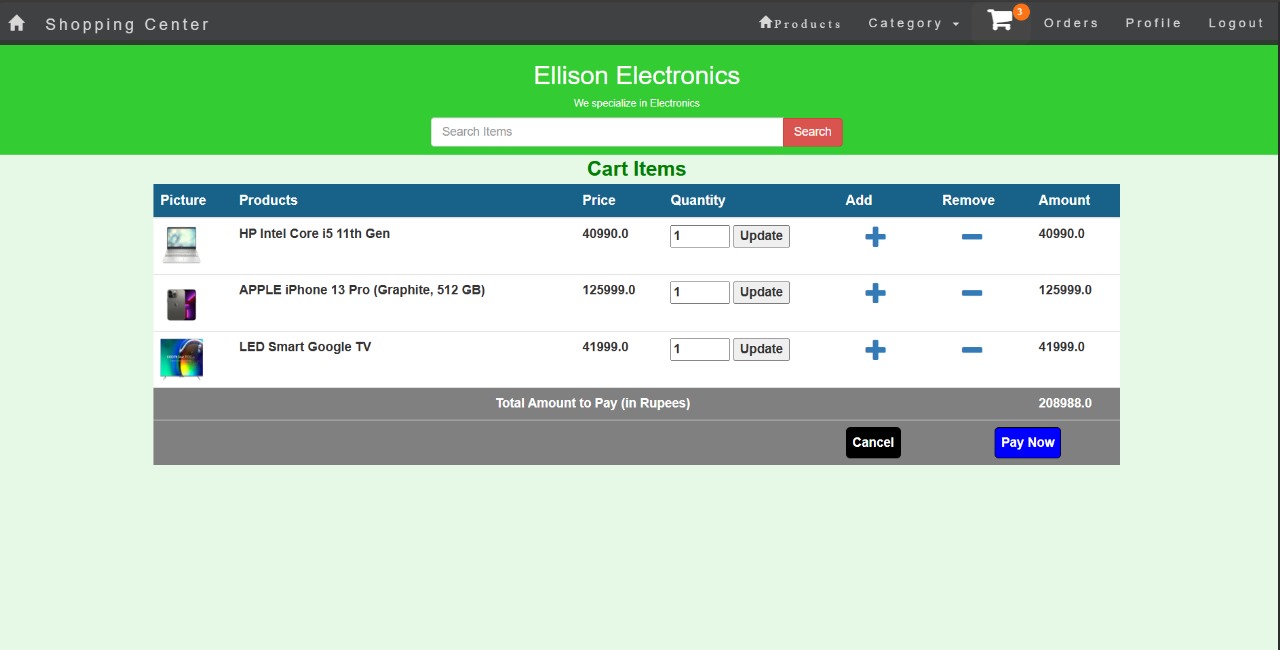
**Registration:**

****

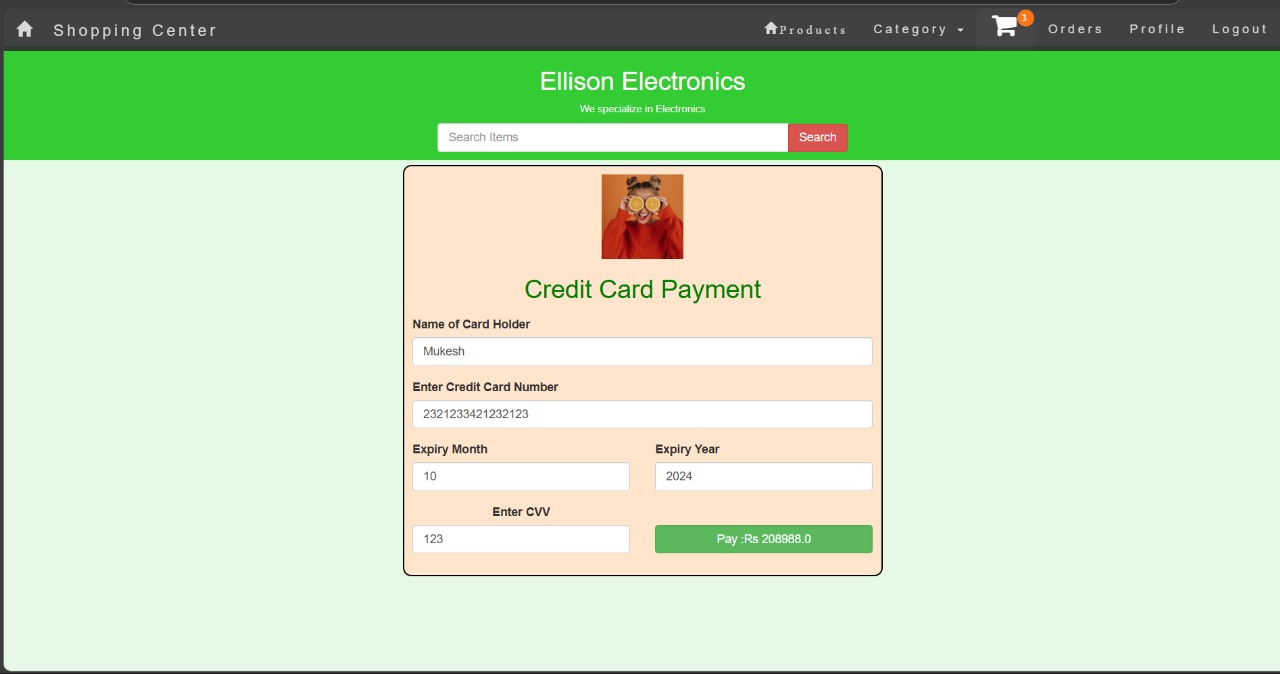
**Home page:**

****

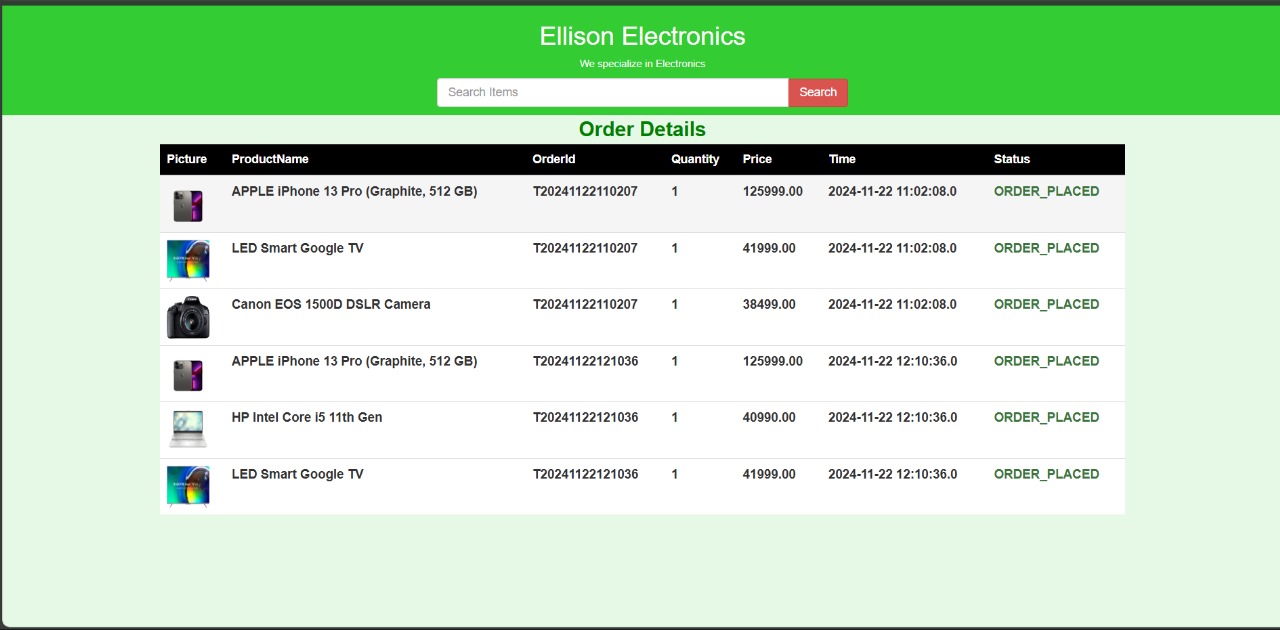
**Cart:**

****

**Payment:**

****

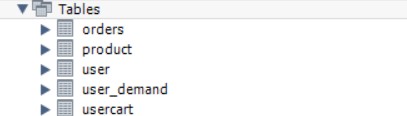
**Order message:**

****

* 1. **Database Design:**

The database design for a Medical E-Commerce Website includes multiple interconnected tables to manage different aspects of the system. The Users Table stores user data (e.g., ID, name, contact details), while the Products Table holds product information (e.g., name, description, price, stock). Orders and Order\_Items track customer orders and product quantities, and the Payments Table handles transaction details. The Reviews Table stores customer feedback, and the Cart Table manages items users add to their cart before purchase. Prescriptions Table is used for uploading and verifying prescriptions, while Categories Table organizes products. Admin Table manages admin access, and the Inventory Table monitors stock levels. Finally, the Shipping Table tracks delivery information. This design ensures efficient data storage and retrieval for all website functions, including user management, order processing, payments, and inventory control.

# Online Shopping cartWeb Store which contains 4 MySQL tables :



* 1. **CODE:**

package com.shashi.srv; import java.io.IOException;

import javax.servlet.RequestDispatcher; import javax.servlet.ServletException; import javax.servlet.annotation.WebServlet; import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest; import javax.servlet.http.HttpServletResponse; import javax.servlet.http.HttpSession;

import com.shashi.beans.UserBean;

import com.shashi.service.impl.UserServiceImpl;

/\*\*

\* Servlet implementation class LoginSrv

\*/ @WebServlet("/LoginSrv")

public class LoginSrv extends HttpServlet {

private static final long serialVersionUID = 1L;

public LoginSrv() {

super();

}

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException { String userName = request.getParameter("username"); String password = request.getParameter("password"); String userType = request.getParameter("usertype"); response.setContentType("text/html");

String status = "Login Denied! Invalid Username or password."; if (userType.equals("admin")) {

if (password.equals("admin") && userName.equals(["admin@](mailto:admin@gmail.com)g[mail.com](mailto:admin@gmail.com)"))

{

RequestDispatcher rd = request.getRequestDispatcher("adminViewProduct.jsp");

HttpSession session = request.getSession(); session.setAttribute("username", userName); session.setAttribute("password", password); session.setAttribute("usertype", userType); rd.forward(request, response);

} else {

RequestDispatcher rd = request.getRequestDispatcher("login.jsp?message=" + status);

rd.include(request, response);

} else {

}

UserServiceImpl udao = new UserServiceImpl(); status = udao.isValidCredential(userName, password); if (status.equalsIgnoreCase("valid")) {

resultPanel.setBackground(Color.WHITE); // Set background color

// Create the Reservation button panel with center alignment JPanel buttonPanel = new JPanel(new GridBagLayout()); GridBagConstraints gbc = new GridBagConstraints(); gbc.anchor = GridBagConstraints.CENTER; buttonPanel.add(createReservationButton(), gbc);

frame.add(mainPanel); frame.add(topPanel, BorderLayout.NORTH); frame.add(resultPanel, BorderLayout.CENTER); frame.add(buttonPanel, BorderLayout.SOUTH);

frame.pack(); frame.setVisible(true);

setupDatabaseConnection();

}

private JPanel createTopPanel() {

JPanel topPanel = new JPanel(new FlowLayout()); topPanel.setBackground(Color.LIGHT\_GRAY); // Set background color

placeComboBox = new JComboBox<>(); placeComboBox.addItem("None"); placeComboBox.addItem("Chennai"); placeComboBox.addItem("Coimbatore"); placeComboBox.setSelectedItem("None");

theatreComboBox = new JComboBox<>

UserBean user = udao.getUserDetails(userName, password); HttpSession session = request.getSession(); Session.setAttribute("userdata", user); session.setAttribute("username", userName); session.setAttribute("password", password); session.setAttribute("usertype", userType);

RequestDispatcher rd = request.getRequestDispatcher("userHome.jsp"); rd.forward(request, response);

} else {

RequestDispatcher rd =request.getRequestDispatcher("login.jsp?message=" + status); rd.forward(request, response);

}

}

}

protected void doPost(HttpServletRequest request, HttpServletResponse response) hrows ServletException, IOException {

doGet(request, response);

}

}

import com.shashi.service.impl.OrderServiceImpl;

/\*\*

\* Servlet implementation class OrderServlet

\*/ @WebServlet("/OrderServlet")

public class OrderServlet extends HttpServlet { private static final long serialVersionUID = 1L;

protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

HttpSession session = request.getSession();

String userName = (String) session.getAttribute("username"); String password = (String) session.getAttribute("password"); if (userName == null || password == null) {

response.sendRedirect("login.jsp?message=Session Expired, Login Again!!");

}

double paidAmount = Double.parseDouble(request.getParameter("amount")); String status = new OrderServiceImpl().paymentSuccess(userName, paidAmount); PrintWriter pw = response.getWriter();

response.setContentType("text/html");

RequestDispatcher rd = request.getRequestDispatcher("orderDetails.jsp");

rd.include(request, response);

}

protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

doGet(request, response)}}

package com.shashi.srv; import java.io.IOException; import java.io.InputStream;

import javax.servlet.RequestDispatcher;

import javax.servlet.ServletException;

import javax.servlet.annotation.MultipartConfig; import javax.servlet.annotation.WebServlet; import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest; import javax.servlet.http.HttpServletResponse; import javax.servlet.http.HttpSession;

import javax.servlet.http.Part;

import com.shashi.service.impl.ProductServiceImpl;

/\*\*

\* Servlet implementation class AddProductSrv

\*/

@WebServlet("/AddProductSrv") @MultipartConfig(maxFileSize = 16177215) public class AddProductSrv extends HttpServlet {

private static final long serialVersionUID = 1L;protected void doGet(HttpServletRequest request, HttpServletResponse

response)

Continue!");

throws ServletException, IOException { HttpSession session = request.getSession();

String userType = (String) session.getAttribute("usertype"); String userName = (String) session.getAttribute("username"); String password = (String) session.getAttribute("password"); if (userType == null || !userType.equals("admin")) { response.sendRedirect("login.jsp?message=Access Denied!");

}

else if (userName == null || password == null) { response.sendRedirect("login.jsp?message=Session Expired, Login Again}

String status = "Product Registration Failed!"; String prodName = request.getParameter("name"); String prodType = request.getParameter("type"); String prodInfo = request.getParameter("info"); double prodPrice =

"));

Integer.parseInt(request.getParameter("quantity"));

Part part = request.getPart("image");

InputStream inputStreProductServiceImpl product = new ProductServiceImpl();

24 | Page

package com.shashi.srv; import java.io.IOException; import java.io.PrintWriter;

import javax.servlet.RequestDispatcher; import javax.servlet.ServletException; import javax.servlet.annotation.WebServlet; import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest; import javax.servlet.http.HttpServletResponse; import javax.servlet.http.HttpSession;

import com.shashi.service.impl.OrderServiceImpl;

/\*\*

\* Servlet implementation class OrderServlet

\*/ @WebServlet("/OrderServlet")

public class OrderServlet extends HttpServlet { private static final long serialVersionUID = 1L;

protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

HttpSession session = request.getSession();

String userName = (String) session.getAttribute("username"); String password = (String) session.getAttribute("password"); if (userName == null || password == null) {

response.sendRedirect("login.jsp?message=Session Expired, Login Again!!"); double paidAmount = Double.parseDouble(request.getParameter("amount")); String status = new OrderServiceImpl().paymentSuccess(userName, paidAmount); PrintWriter pw = response.getWriter();

response.setContentType("text/html");

RequestDispatcher rd = request.getRequestDispatcher("orderDetails.jsp"); rd.include(request, response);

}

protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

doGet(request, response);

}

}

25 | Page

package com.shashi.srv;

import java.awt.image.BufferedImage; import java.io.ByteArrayOutputStream; import java.io.File;

import java.io.IOException; import javax.imageio.ImageIO;

import javax.servlet.ServletException; import javax.servlet.ServletOutputStream; import javax.servlet.annotation.WebServlet; import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest; import javax.servlet.http.HttpServletResponse; import com.shashi.service.impl.ProductServiceImpl; @WebServlet("/ShowImage")

public class ShowImage extends HttpServlet {

private static final long serialVersionUID = 1L;

public ShowImage() {

super();

}

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException { String prodId = request.getParameter("pid");

ProductServiceImpl dao = new ProductServiceImpl();

byte[] image = dao.getImage(prodId); if (image == null) {

File fnew = new File(request.getServletContext().getRealPath("images/noimage.jpg "));

26 | Page

BufferedImage originalImage = ImageIO.read(fnew); ByteArrayOutputStream baos = new ByteArrayOutputStream(); ImageIO.write(originalImage, "jpg", baos);

image = baos.toByteArray();

}

# Conclusion:

In conclusion, a Online Shopping cart website plays a crucial role in enhancing accessibility to e-commerce products and services. By offering a convenient, secure platform for purchasing medical supplies, pharmaceuticals, and health-related products, it helps meet the growing demand for online e-commerce solutions. With user-friendly interfaces, reliable delivery systems, and strong data security measures, such websites can greatly improve the patient experience, foster trust, and contribute to better health outcomes. Ensuring compliance with regulations and maintaining high standards of product quality will be key to long-term success in the medical e-commerce space.

**Reference links:**

[1] <https://www.javatpoint.com/java-awt> [2] <https://www.javatpoint.com/java-swing>

27 | Page