

E-Commerce Website for Gaming Products

Project Members

22k-4814: M Rafay Randhawa

22k-5196: Anas Bashir

Date of Submission

4-12-2024

Live Hosting URL

This project will be demonstrated locally.

Introduction

The E-Commerce Website for Gaming Products is an online platform dedicated to selling gaming-related products such as accessories, merchandise, and hardware. The project provides a seamless shopping experience for customers while ensuring comprehensive management tools for administrators.

This project incorporates a dual-layer functionality:

1. Customer Interface: Offers a user-friendly interface for browsing and purchasing products, leaving reviews, and tracking orders.
2. Admin Panel: A robust backend system for managing categories, products, users, reviews, payments, and orders.

The project aims to bridge the gap between gaming enthusiasts and quality products, offering an efficient solution for e-commerce operations in the gaming domain.

Detailed Project Description

The website is designed to cater to the needs of both customers and administrators by providing:

- A dynamic product catalog with detailed product pages.
- A secure and efficient checkout process integrated with payment management.
- Tools for administrators to monitor and control the platform's functionality.

The project is implemented using Spring Boot for the backend and JavaScript for frontend scripting. Its primary focus is on ease of use, security, and scalability.

Application Features

Customer Features:

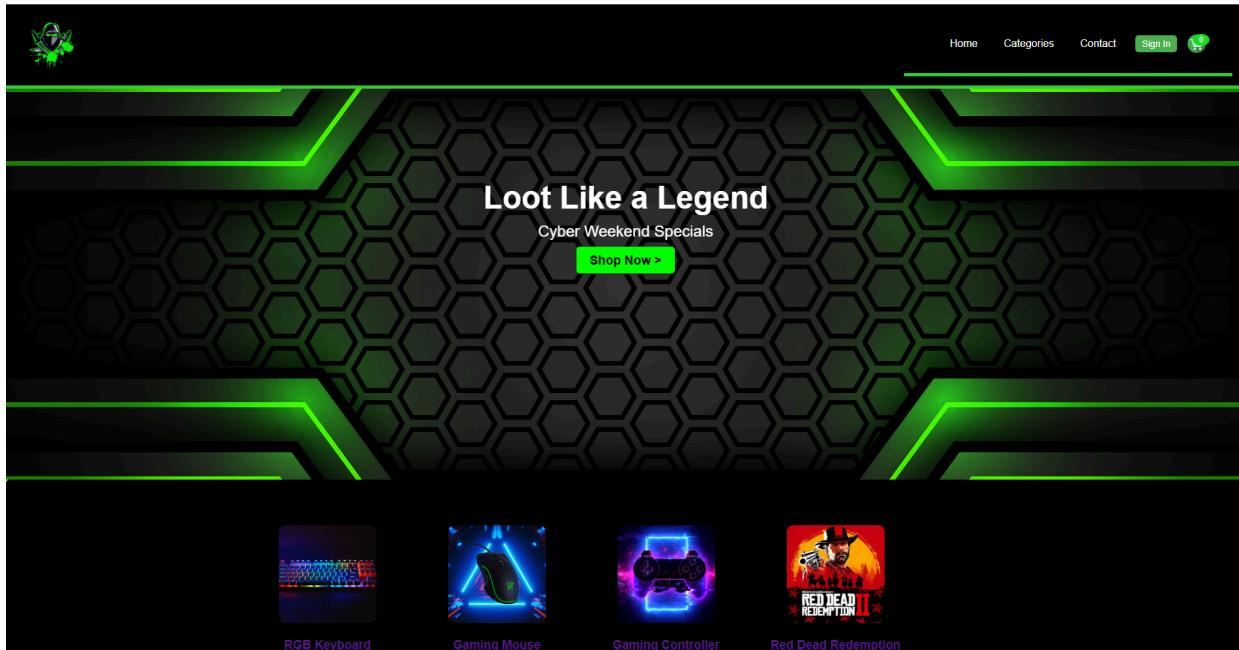
1. Product Browsing: View all gaming products with search and filter options.
2. Detailed Product Pages: Includes images, specifications, and reviews.
3. User Registration & Profile Management: Create and manage user accounts.
4. Add to Cart & Checkout: Seamlessly add items to the cart and complete purchases.
5. Order Tracking: View the status of placed orders.
6. Product Reviews: Customers can leave reviews on purchased products.

Admin Features:

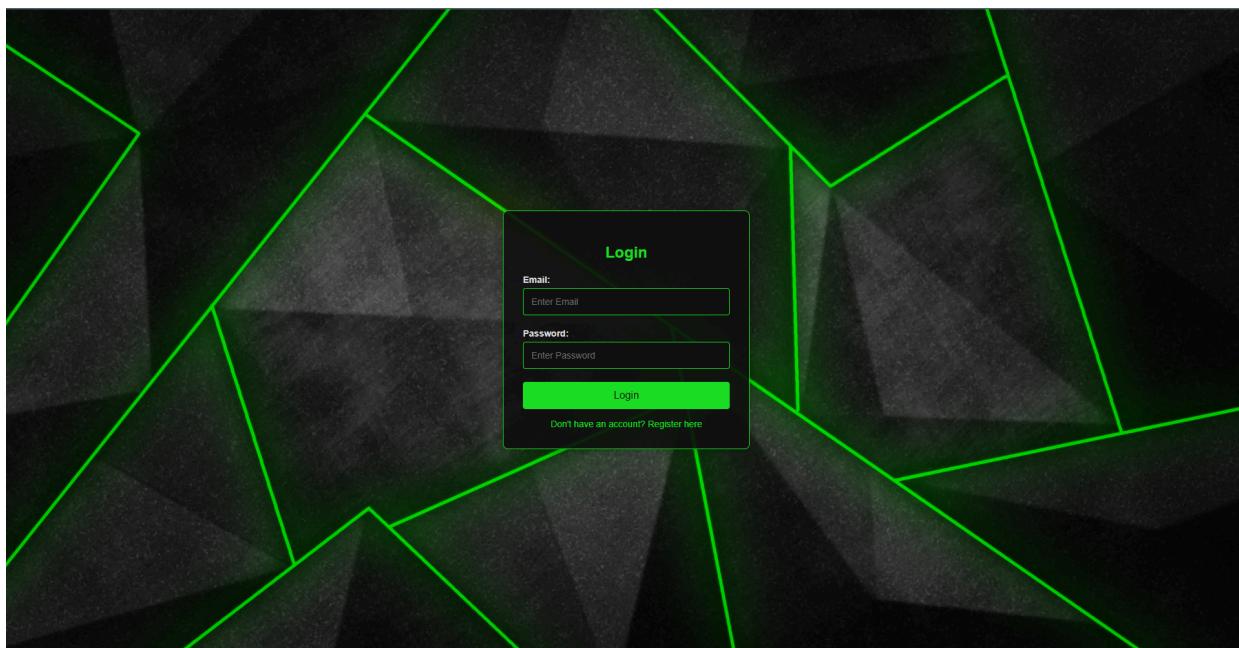
7. Manage Categories: Add, update, or delete product categories.
8. Manage Products: Create, update, or remove products from the inventory.
9. User Management: View and manage user accounts.
10. Review Moderation: Approve or delete inappropriate reviews.
11. Payment Management: Monitor and process payments securely.
12. Order Management: Oversee and update the status of customer orders.

Screenshots

Home:



Login:



Register:

Register New User

First Name:
Enter First Name

Last Name:
Enter Last Name

Email:
Enter Email

Password:
Enter Password

Confirm Password:
Confirm Password

Phone:
Enter Phone Number

Address:
Enter Address

Register

Already have an account? [Login here](#)

Dynamic navbar when user is logged in:



Categories Page:

Shopping Cart popup:

Shopping Cart

Product A

Quantity: 2

Price: 199.98

Delete

Checkout

Close

Profile page:

Menu

- Personal Information
- View Orders

Personal Information

First Name: John
Last Name: Doe

Contact Information

Email: doe@example.com
Phone: 038/654321

Address

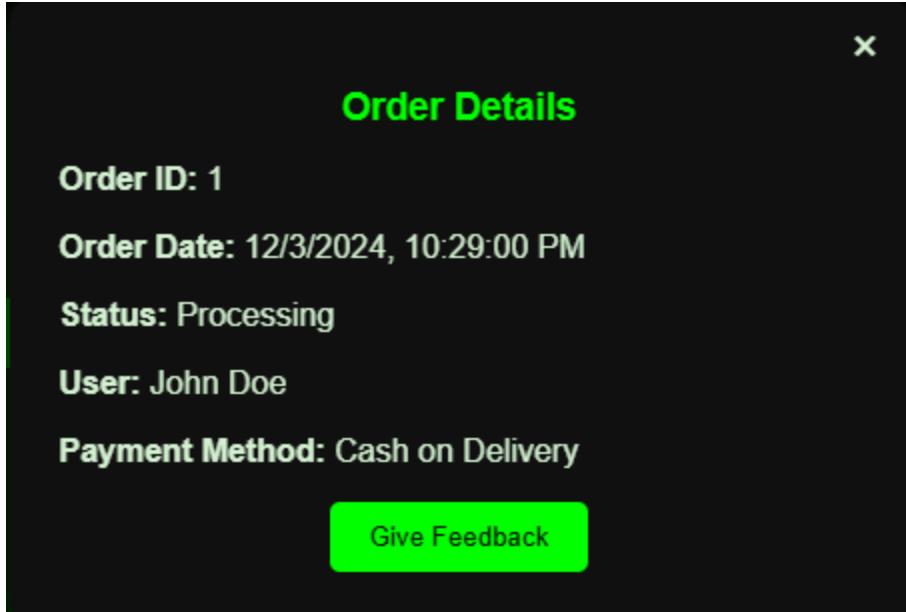
Address: xyz street

[Edit Profile](#)

View Orders

ORDER ID	DATE	STATUS	ACTION
1	2024-12-03T22:29:00Z	Processing	View Order
2	2024-12-04T00:06:53Z	Processing	View Order

Order details popup:



Feature Explanations

Explain each feature in detail using textual descriptions and diagrams/screenshots.

1. Product Browsing

- Description: Users can search for and filter products based on categories or keywords.

2. Manage Products (Admin Panel)

- Description: Admins can add new gaming products or edit/remove existing ones.

Setup Instructions

Prerequisites

- Software:

- Java (JDK 8 or higher)
- Spring Boot
- Oracle 11g

- Tools:

- Code editor (e.g., IntelliJ IDEA, Eclipse)
- Postman (optional for testing APIs)
- Oracle Sql Developer

Steps to Deploy

1. Clone/download the project repository.
2. Build the project using Gradle.
3. Set up the database:
 - The required tables will be created automatically when you run the project.
4. Configure environment variables for database connection:
 - Update `application.properties` file.
5. Start the Spring Boot application:

```
```bash
java -jar E_CommerceProject.jar
````
```
6. Access the application in your browser:
 - Customer Interface: '<http://localhost:8085>'
 - Home Page: '<http://localhost:8085/home>'
 - Login: '<http://localhost:8085/login>'