Jewellery E-commerce Web Application

Software Requirements Specification

INT-252

Piyush Vishwakarma
12216662

Prepared for
Continuous Assessment 3
Autumn 2024

Jewellery E-commerce Web Application

Table of Contents

1. Introduction

1.1 Purpose

The purpose of the Jewellry E-Commerce Website is to create an accessible, engaging, and user-friendly platform for showcasing and selling jewellry collections online. This application aims to:

- Provide an attractive and intuitive interface for browsing various jewelry products, allowing users to easily navigate through collections such as rings, necklaces, bracelets, and more.
- Simplify the shopping experience with organized categories, detailed product descriptions, and high-quality images, enabling users to make informed purchasing decisions.
- Facilitate a seamless purchasing process by integrating a shopping cart, secure checkout, and various payment options to enhance user convenience.
- Enhance customer engagement through sections like "About Us," which share the brand's story, values, and unique craftsmanship, fostering a deeper connection with the audience.
- Support store management by allowing for updates to product listings, inventory tracking, and price adjustments, helping the business stay current and efficient.
- Improve customer satisfaction with efficient order tracking, email notifications, and an overall responsive platform that adapts well across different devices and screen sizes

1.2 Scope

Scope of the Jewellry E-Commerce Website
The scope of the Jewellry E-Commerce Website outlines its main functionalities and limitations:
Functionalities:

- ***User Registration and Secure Login: Allows customers to create accounts for a personalized shopping experience and securely log in to access their profile, wish list, and order history.***
- ***Product Browsing and Filtering: Users can explore jewellry collections, view high-quality images, read detailed descriptions, and filter products by category, price, material, or style.***
- ***Shopping Cart and Checkout: Customers can add items to their cart, review their selections, and proceed to checkout with integrated payment options for secure transactions.***
- ***Order Tracking: Provides customers with order status updates and tracking information for a transparent post-purchase experience.***
- ***Content Management: Admins can manage product listings, update inventory, adjust prices, and add or remove items as needed.***

- ***Promotions and Discounts: Enables the display of promotional banners, discount codes, or seasonal sales to enhance user engagement and incentivize purchases.***

1.3 Definitions, Acronyms, and Abbreviations

A list of technical terms, acronyms, and abbreviations used throughout this document:

API (Application Programming Interface): A set of functions and protocols that allow different software applications to communicate with each other.

CRUD (Create, Read, Update, Delete): Basic operations for managing data in databases and applications.

DFD (Data Flow Diagram): A graphical representation showing how data flows through a system, including processes, data stores, and interactions.

EMR (Electronic Medical Records): Digital version of a patient's paper chart, used by healthcare providers to store patient data.

JWT (JSON Web Token): A secure method for transmitting information between parties, commonly used for authentication and session management.

UI (User Interface): The visual elements through which users interact with the application.

UX (User Experience): The overall experience and satisfaction a user has while interacting with the application.

HTTPS (Hypertext Transfer Protocol Secure): A protocol for secure communication over the internet, ensuring data encryption.

MERN Stack: A technology stack including MongoDB, Express.js, React, and Node.js, used for building web applications.

REST (Representational State Transfer): A set of architectural principles for designing networked applications, often used for web APIs.

SSL (Secure Sockets Layer): A security protocol for establishing encrypted links between web servers and browsers.

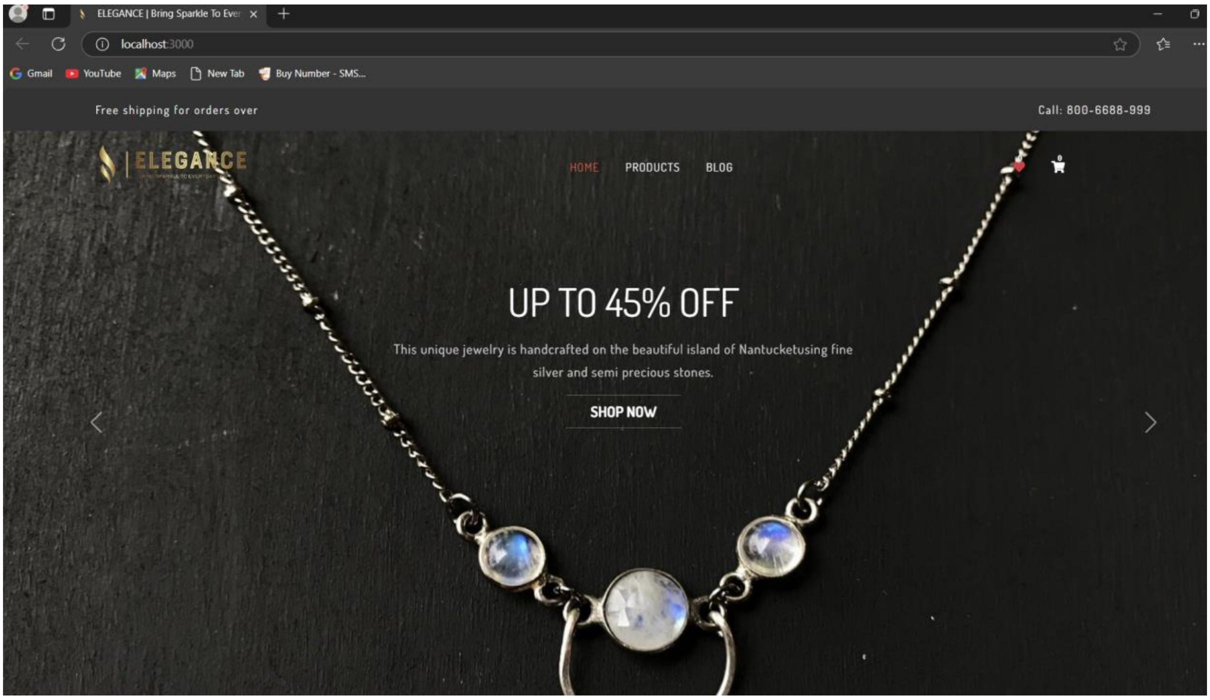
CRUD Operations: Basic actions in data management (Create, Read, Update, Delete) fundamental to web applications.

OAuth: An open standard for access delegation commonly used for token-based authorization.

JSON (JavaScript Object Notation): A lightweight format for data interchange between servers and web applications.

UI/UX Design Standards: Guidelines ensuring the application is user-friendly and visually consistent.

This glossary provides clarity on key technical terms, enhancing understanding of the document’s contents.



Free shipping for orders over

Call: 800-6688-999

UP TO 45% OFF

This unique jewelry is handcrafted on the beautiful island of Nantucketusing fine silver and semi precious stones.

1.4 References

References for Standards, Frameworks, and Libraries in the Development of the Jewelry E-Commerce Website

- React: A JavaScript library for building user interfaces, especially suitable for single-page applications. React allows for efficient component-based UI development, enabling dynamic product displays, shopping cart management, and an overall seamless user experience.
- Redux : A predictable state container for JavaScript applications, commonly used with React to manage complex state logic for functions like user authentication, cart management, and product filtering.
- Node.js and Express.js: A JavaScript runtime (Node.js) and a web application framework (Express.js) used to create a RESTful API for backend operations. These enable secure and efficient server-side processing, handle user authentication, and facilitate data retrieval for product information, order history, and user profiles.
- MongoDB (or SQL-based databases): A database solution for storing application data. MongoDB, a NoSQL database, can store user profiles, product details, and orders in a flexible, document-based format. Alternatively, SQL databases can be used if relational data management is preferred.

- *PCI-DSS Compliance (Payment Card Industry Data Security Standard): An information security standard for businesses that handle credit card information, providing best practices for secure payment processing and safeguarding user payment details.*
- *ISO/IEC 27001: An international standard for information security management, establishing protocols for data security, confidentiality, and integrity in user data handling, relevant for e-commerce platforms handling sensitive customer data.*
- *GDPR (General Data Protection Regulation): A regulation in EU law on data protection and privacy, ensuring strict privacy controls for user data. Adherence to GDPR ensures compliance with privacy laws, especially for EU-based customers.*
- *SSL/TLS Encryption: Security protocols for ensuring secure connections between the user's browser and the server, protecting sensitive information like user credentials, payment details, and order information during transmission.*

1.5 Overview

Jewellry E-Commerce Website Documentation Outline

Introduction

The Introduction section provides essential background information for understanding the jewellry e-commerce website. It describes the purpose of the website, which aims to offer a seamless online shopping experience for customers looking to browse and purchase jewellry. The scope of the website is covered, detailing key functionalities (e.g., product browsing, secure checkout, order tracking) and limitations (e.g., internet dependency, no offline access). This section also includes definitions, acronyms, and abbreviations for technical terms used throughout the document, along with references to relevant libraries, frameworks, and compliance standards (e.g., React, MongoDB, PCI-DSS, GDPR).

General Description

This section provides a high-level overview of the jewellry e-commerce website.

- *Product Perspective describes how the website fits into the broader e-commerce ecosystem by providing an online marketplace for jewellry.*
- *Product Functions outlines the main features such as user registration, secure login, product browsing, shopping cart, checkout, and order tracking.*
- *User Characteristics defines the intended users (e.g., customers, administrators, store managers) and their needs (e.g., easy navigation, secure transactions, product management).*
- *General Constraints highlight challenges such as security concerns, internet dependency, and mobile responsiveness.*
- *Assumptions and Dependencies mention third-party services or technologies, such as payment gateways (Stripe, PayPal) and shipping integration, that are integral to the website’s operation.*

Specific Requirements

This core section specifies the technical and functional requirements of the jewellry e-commerce website.

- *External Interface Requirements cover the details about the user interface (UI), hardware, software, and communication interfaces required to make the system function properly.*

- ***Functional Requirements explain core features, such as browsing jewellry categories, adding products to the cart, managing orders, and user account management in detail.***
- ***Non-Functional Requirements outline attributes like performance (fast loading times), reliability (consistent uptime), and security (payment encryption and user data protection).***
- ***Design Constraints address any limitations or standards in UI/UX design, such as responsive layouts and accessibility for all users.***
- ***Other Unique Requirements specific to the jewellry business, such as inventory management and customer engagement features, will be outlined here.***

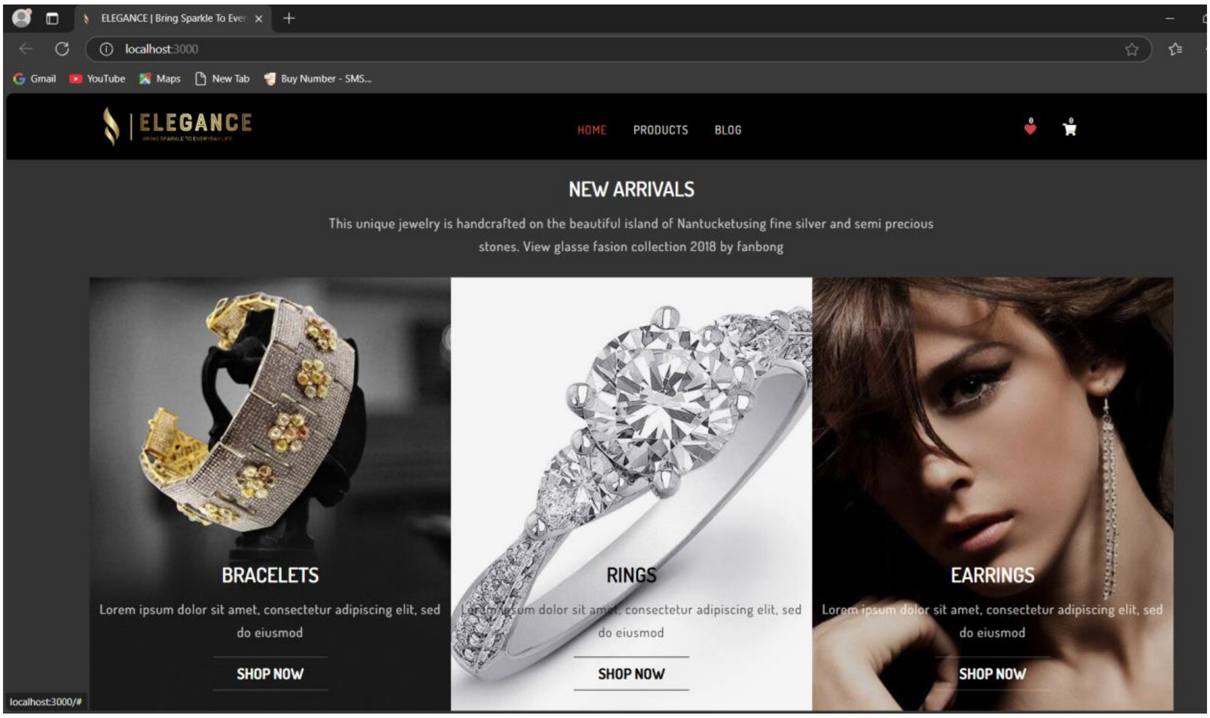
Analysis Models

This section includes visual representations, such as Data Flow Diagrams (DFDs), to explain how data moves through the system. These diagrams illustrate user interactions, product data management, and transaction processing, helping stakeholders understand the data structure and flow throughout the website's operations.

GitHub and Deployed Links

This section provides access to the GitHub repository and the live deployed application:

- ***The GitHub repository link allows developers and stakeholders to access the project's source code, track development progress, and contribute to the project.***
- ***The deployed application link offers a hands-on experience for users to test the website's features, including browsing products, adding to the cart, and checking out.***



NEW ARRIVALS

This unique jewelry is handcrafted on the beautiful island of Nantucketusing fine silver and semi precious stones. View glasse fasion collection 2018 by fanbong

BRACELETS

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod

RINGS

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod

EARRINGS

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod

2. General Description

2.1 Product Perspective

The Jewellry E-Commerce Website functions as a comprehensive online shopping platform for purchasing jewellry, providing a seamless and convenient experience for customers. This website aims to enhance the shopping process by allowing users to explore various jewellry collections, make secure purchases, and track orders with ease.
From a business perspective, the website offers a streamlined approach to managing product listings, inventory, and customer orders. Store managers and administrators can update product details, prices, and manage stock in real time, improving operational efficiency. It also allows businesses to reach a broader audience by providing an online store that operates 24/7, enabling customers to browse and purchase jewellry at their convenience.
For customers, the website offers an intuitive, user-friendly interface that makes the online shopping experience enjoyable. Customers can browse through detailed product categories (e.g., rings, necklaces, bracelets), read descriptions, view high-quality images, and make purchases with secure payment options. They can also track their orders, receive notifications for updates, and manage their profiles, including order history and saved preferences.
This e-commerce platform integrates smoothly with payment gateways (e.g., Stripe, PayPal) and shipping services, ensuring secure transactions and reliable order fulfillment. It can operate as a standalone platform or be integrated into a broader retail ecosystem that may include brick-and-mortar stores, marketing systems, or customer loyalty programs.

2.2 Product Functions

User Registration and Login:
Provides secure access for customers, store managers, and administrators through a simple registration and login process. User roles (customer, admin, store manager) are assigned based on credentials to ensure appropriate access to features, such as browsing products, managing orders, and updating product listings.
Product Browsing and Filtering:
Allows customers to explore jewelry collections by categories such as rings, necklaces, bracelets, etc. The browsing experience is enhanced with advanced filtering options (e.g., material, price range, style) and high-quality images to help users easily find the jewelry they want to purchase.
Shopping Cart Management:
Enables customers to add items to their shopping cart, review selected products, adjust quantities, and proceed to checkout. This feature ensures that users can manage their shopping experience seamlessly before completing their purchase.

Secure Checkout and Payment Processing:
Facilitates a secure and straightforward checkout process for customers, with integrated payment gateways (e.g., Stripe, PayPal) to process transactions safely. Customers can choose from multiple payment methods (credit/debit cards, digital wallets) to complete their purchase.
Order Tracking and Notifications:
Sends automated notifications to customers regarding order status (e.g., order confirmation, shipping updates). Customers can track their orders through the website, ensuring they stay informed about delivery timelines.
Admin Product Management:
Provides administrators and store managers with the ability to manage product listings, update product details (e.g., descriptions, images, prices), and monitor inventory. This feature ensures that the website stays up-to-date with new products and changes to existing ones.
Customer Profile Management:
Allows customers to create and manage their profiles, including saving favorite products, viewing order history, and updating their contact and payment information. This function enhances user convenience and fosters customer loyalty.

2.3 User Characteristics

 Customers:

- **Primary Role**: Customers are the primary users of the jewelry website, browsing and purchasing products.
- **Needs**:
    - Easy-to-navigate product catalog, with clear categories for different types of jewelry (e.g., rings, necklaces, bracelets).
    - High-quality images and detailed descriptions to make informed purchase decisions.
    - A secure, convenient shopping experience, including features like shopping cart management, secure checkout, and multiple payment options.
    - Access to order tracking and notifications regarding order status (e.g., shipped, delivered).
    - User-friendly account management to save preferences, payment information, and order history.
- **Technical Proficiency**: Varies from casual shoppers to tech-savvy users. The website should be intuitive and responsive across devices, making it easy for both experienced and new users to navigate.
- **Demographics**: Customers can range from various age groups and backgrounds, seeking quality jewelry at various price points, from everyday wear to special occasions.

 Store Managers/Administrators:

- **Primary Role**: Store managers or administrators are responsible for managing the backend of the jewelry website, handling product listings, inventory, customer orders, and promotions.
- **Needs**:
  - A robust dashboard for managing product inventory, updating descriptions, prices, and images.
  - Tools for tracking customer orders, monitoring sales trends, and processing returns or exchanges.
  - Ability to manage customer profiles and respond to queries or issues in a timely manner.
  - Support for running promotions, discount codes, and flash sales to engage customers and drive sales.
- **Technical Proficiency**: Moderately high, with a focus on using the admin interface to efficiently manage the website's operations.
- **Demographics**: Typically store owners, managers, or marketing teams responsible for the operational side of the e-commerce platform.

⍰ **Website Developers**:

- **Primary Role**: Developers are responsible for maintaining the technical aspects of the website, ensuring that the website is functional, secure, and optimized for performance.
- **Needs**:
  - Access to the codebase and ability to deploy updates or fixes to the site.
  - Implementation of security protocols (e.g., SSL encryption, PCI-DSS compliance) to ensure safe transactions.
  - Optimization for website performance, including fast loading times, responsive design, and mobile compatibility.
  - Integration with third-party services, such as payment gateways and shipping providers.
- **Technical Proficiency**: High, as developers need expertise in technologies such as React, Node.js, databases (MongoDB, SQL), and APIs.
- **Demographics**: Web developers, software engineers, or IT specialists working in the e-commerce or retail technology sector.

2.4 General Constraints

⍰ ***Requirement for Internet Connectivity:***

- ***The jewelry e-commerce website is web-based, meaning a continuous internet connection is required for users to browse products, make purchases, and manage orders. Without internet access, customers cannot complete transactions, and store administrators cannot update product listings or track inventory. This constraint highlights the necessity of stable internet access for the proper functioning of the website.***

⍰ ***Mobile Device Compatibility and Responsiveness:***

- ***The website must be optimized for mobile devices to ensure a seamless shopping experience on smartphones and tablets. If not fully optimized, users may face difficulties navigating the site, viewing high-quality images, or completing the checkout process on smaller screens. Ensuring the site is responsive and adapts well*** to different screen sizes is crucial for capturing mobile shoppers and providing an enjoyable user experience.

 Compliance with PCI-DSS for Payment Security:

- Given the e-commerce nature of the website, it must comply with PCI-DSS (Payment Card Industry Data Security Standard) to ensure secure handling of payment information. This includes using encryption protocols for transactions, storing customer payment data securely, and implementing safeguards to prevent fraud. Compliance with PCI-DSS is mandatory to protect customer financial information and maintain trust in the platform.

 Privacy and Data Protection:

- The jewelry website must adhere to privacy regulations such as GDPR (General Data Protection Regulation) for European customers and local data protection laws in other regions. The website must ensure that personal data, including customer profiles and order history, is handled securely and only used for intended purposes. Secure user authentication, data encryption, and transparent privacy policies are necessary to protect user privacy and comply with legal requirements.

 Inventory and Product Management:

- The website's inventory management system should reflect real-time product availability. There may be limitations in handling large inventories or multiple product variations (e.g., different sizes, colors, or materials), requiring efficient backend systems for product updates and stock tracking. Admins must ensure accurate stock levels and avoid selling out-of-stock items to customers.

 Shipping and Delivery Constraints:

- Shipping options may be limited by geographical location, shipping partners, and cost constraints. Certain regions may not be eligible for free shipping or fast delivery options, which may affect customer satisfaction. The website must clearly communicate shipping terms, delivery times, and associated costs to users during the checkout process.

 Scalability and Load Performance:

- As the website grows in terms of traffic and product listings, ensuring that the platform can scale efficiently is essential. The infrastructure must support a growing number of visitors, orders, and product listings without affecting performance. High load times or website downtime could negatively impact the shopping experience and sales.

2.5 Assumptions and Dependencies

Assumptions

1. User Device Compatibility:
   It is assumed that users will access the website through devices with internet connectivity, including desktops, laptops, smartphones, and tablets. The website is designed to be responsive, ensuring compatibility across different screen sizes. However, the performance may be optimized for modern browsers (e.g., Chrome, Firefox, Safari) and mobile devices, and users with older devices or outdated browsers may experience limited functionality or degraded performance.
2. User Access and Technical Proficiency:
   It is assumed that customers and administrators will have basic internet and

computer skills to navigate the website and use its features (browsing products, adding items to the cart, and completing the checkout process). Some users may need additional support or instructions, particularly with payment methods or account management.

3. *Secure Payment Systems and SSL Encryption:*
   It is assumed that customers will use secure and trusted payment methods for completing transactions, and that the website will utilize SSL encryption (Secure Socket Layer) for protecting customer data during online transactions. This helps build customer trust and ensures data security.
4. *Regulatory Compliance for Payment and Data Privacy:*
   It is assumed that the website will comply with relevant legal requirements, including PCI-DSS (for secure handling of credit card information) and GDPR (General Data Protection Regulation) for customers in the EU. This ensures that customer information is handled securely and in compliance with privacy regulations.
5. *Availability of Shipping Partners:*
   It is assumed that the website will be able to offer reliable and timely shipping options through third-party logistics providers. Customers will rely on these external services to deliver their purchases on time and within the promised timeframe.

Dependencies

1. *Software Stack Requirements:*
   The jewelry website relies on specific software technologies, including:
   - *React for the frontend user interface, ensuring a dynamic and responsive shopping experience.*
   - *Node.js and Express.js for backend operations, handling customer requests, processing orders, and interacting with the database.*
   - *MongoDB (or other database systems like MySQL) for storing customer data, product listings, and order information, allowing for quick retrieval and scalability.*
2. *Third-Party Libraries and APIs:*
   Key functionalities may depend on external services or libraries, including:
   - *Payment Gateways (e.g., Stripe, PayPal) for secure payment processing.*
   - *Shipping APIs for real-time shipping cost calculation and tracking (e.g., USPS, FedEx).*
   - *Email/Notification APIs for sending order confirmations, shipping updates, and promotional offers (e.g., SendGrid, Mailgun). Dependencies on these external services may impact the website's functionality if any of these services experience downtime, compatibility issues, or changes in their API structures.*
3. *Hosting Environment:*
   The website requires a reliable hosting environment with sufficient server capacity to handle user traffic, secure data storage, and support e-commerce operations. Providers like AWS, Heroku, or DigitalOcean may be used for hosting. The performance of the website is dependent on the reliability and scalability of the hosting infrastructure.

4. ***Data Privacy and Security Compliance:***
   The website must ensure compliance with data privacy regulations such as GDPR for users in the EU and other relevant local data protection laws. It is dependent on strong security measures, including encryption of sensitive data (payment details, personal information), secure authentication, and restricted access based on user roles to ensure customer privacy.
5. ***Scalability and Performance Infrastructure:***
   As the website grows in terms of traffic and product listings, the hosting infrastructure must be able to scale efficiently to handle an increasing number of users and orders. A scalable cloud infrastructure (e.g., AWS EC2, Kubernetes) and database performance optimization techniques are necessary to maintain high performance and minimize downtime.

3. Specific Requirements

3.1 External Interface Requirements

The Jewellery E-commerce Web Application depends on specific external interfaces to facilitate smooth interaction between users and the system. These interfaces, including user, hardware, software, and communications, play an essential role in delivering a consistent, reliable, and accessible experience for all users.

3.1.1 User Interfaces

The **User Interface (UI)** for the jewelry e-commerce website is designed to be aesthetically pleasing, user-friendly, and intuitive for both customers and administrators. Each user type (customers and administrators) will have tailored dashboards and features to meet their specific needs, ensuring a smooth and engaging shopping experience. Key UI elements include:

Customer Dashboard:

- **Home Page**:
  A visually appealing homepage that showcases featured products, special promotions, and new arrivals. Customers can browse various product categories (e.g., rings, necklaces, bracelets) or search for specific items using filters.

Administrator Dashboard:

- **Product Management**:
  Administrators have an interface to add, update, or remove products from the website. They can manage product details like descriptions, pricing, images, and stock availability.
- **Order Management**:
  This dashboard allows administrators to monitor and manage customer orders. They can view all order details, update order statuses (e.g., processing, shipped, delivered), and track shipments.

Responsive and Intuitive Design:

- The website's UI is responsive and adapts seamlessly to all screen sizes, ensuring that customers can easily browse and shop on their desktops, tablets, and smartphones. Key design elements include large product images, clear calls-to-action (e.g., "Add to Cart," "Buy Now"), easy-to-read typography, and intuitive navigation menus.

3.1.2 Hardware Interfaces

While the application does not have specific hardware dependencies, some minimal hardware requirements are recommended:

Customer Devices:

- **Desktop and Laptop Computers**: Running modern web browsers such as **Google Chrome**, **Mozilla Firefox**, **Safari**, or **Microsoft Edge**. These devices provide the best experience for browsing and purchasing products, as they offer large screen sizes and more precise navigation (via mouse and keyboard).

Server Requirements:
Cloud Hosting (Recommended for scalability and reliability):

- **Provider**: Cloud platforms like **Amazon Web Services (AWS)**, **Google Cloud Platform (GCP)**, **Microsoft Azure**, or **DigitalOcean** are ideal. They offer scalable server options to handle high volumes of traffic and ensure smooth performance.
- **Web Server**: A **Nginx** or **Apache HTTP server** will be used to serve web pages and handle client requests.

3.1.3 Software Interfaces

The application requires integration with specific software tools, libraries, and platforms to provide its core functionalities:

Frontend Libraries:
React is used for building the user interface, allowing a dynamic, component-based UI that enhances user interactivity.

Backend Framework:
Express.js, built on Node.js, powers the server-side logic, managing data flow and supporting interactions between the frontend, database, and external services.

Database:
MongoDB is used to store patient records, appointments, user data, and system logs. This NoSQL database supports high scalability and efficient querying for rapid data retrieval.

Authentication Services:
JSON Web Tokens (JWT) enable secure, role-based access control, ensuring that patients, doctors, and administrators have appropriate access permissions.

Notification Services:
Integration with email or SMS services (such as Twilio or SendGrid) allows automated notifications and reminders, enhancing patient engagement and reducing missed appointments.

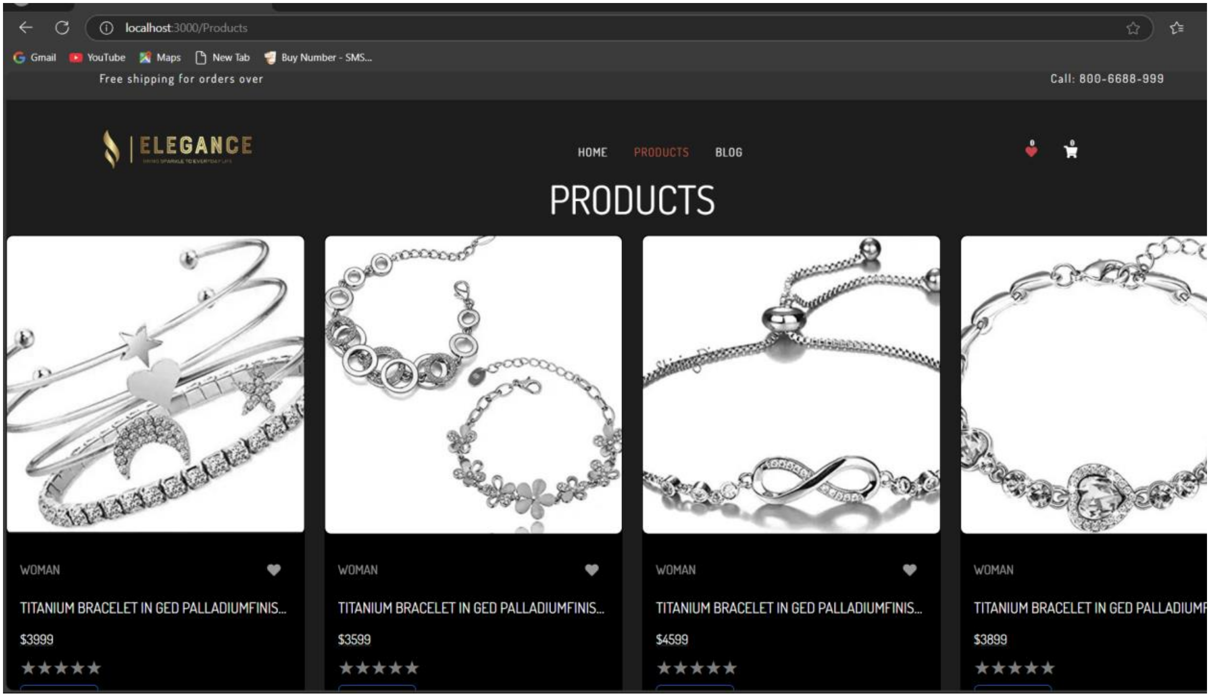
3.1.4 Communications Interfaces

The application's communications interfaces facilitate data exchange between different components and external systems. Key communication requirements include:

Network Requirements:
Reliable internet connectivity is essential for accessing and interacting with the application. HTTPS protocol is required to secure data transmission between the client and server, ensuring that sensitive patient data remains protected.

API Requirements:
The application may expose certain APIs for appointment scheduling, user management, and notifications, allowing interoperability with other systems or third-party applications. Internal APIs manage interactions between the frontend and backend, while external APIs, such as those for email/SMS notifications, enable communication with outside services.

These external interfaces ensure seamless integration, security, and efficiency, delivering a consistent experience for users across all functionalities within the application.



Free shipping for orders over

Call: 800-6688-999

PRODUCTS

WOMAN
TITANIUM BRACELET IN GED PALLADIUMFINIS...
$3999
★★★★★

WOMAN
TITANIUM BRACELET IN GED PALLADIUMFINIS...
$3599
★★★★★

WOMAN
TITANIUM BRACELET IN GED PALLADIUMFINIS...
$4599
★★★★★

WOMAN
TITANIUM BRACELET IN GED PALLADIUMF
$3899
★★★★★

3.2 Functional Requirements

Functional requirements describe the core capabilities that the Jewellery E-commerce Web Application must support. These functionalities enable users to perform essential tasks, ensuring smooth and efficient operation of the system.

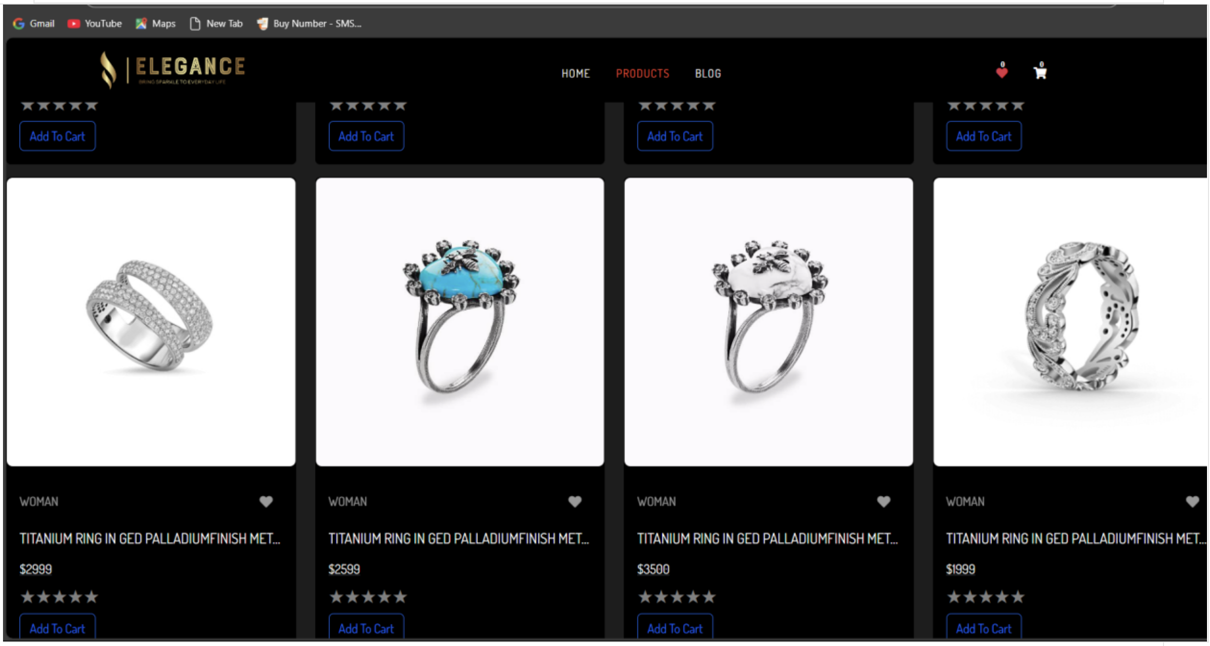
3.2.1 Product Catalog Management

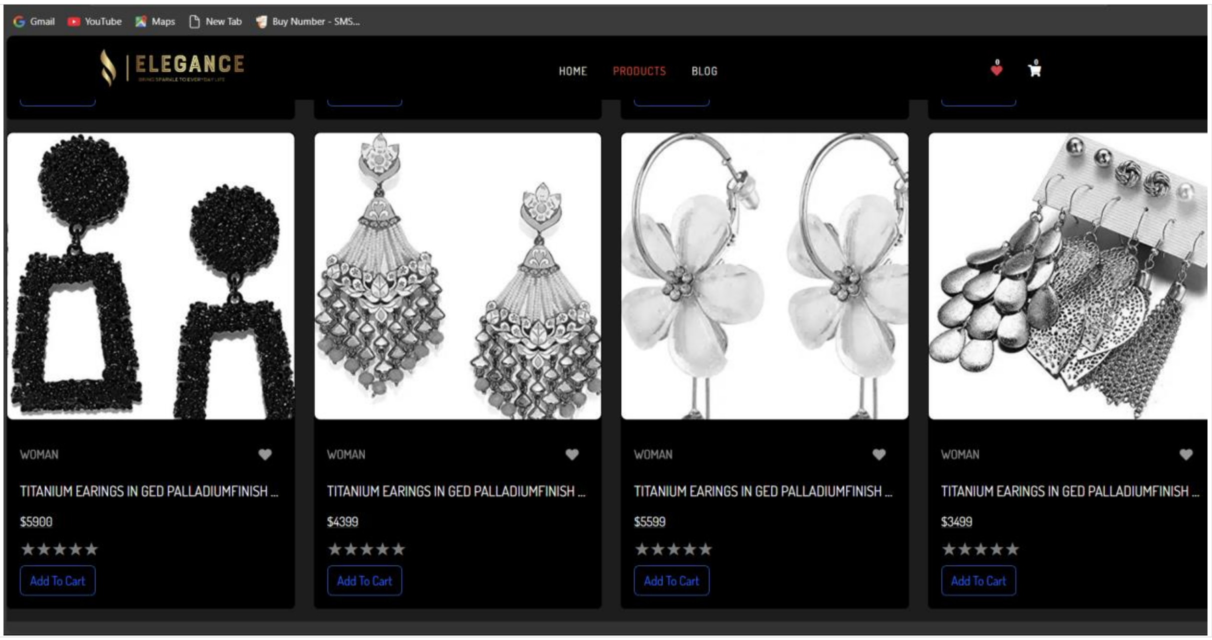
- □ **View Product Listings:** Customers can browse a catalog of available jewelry products. Products are categorized (e.g., rings, necklaces, earrings, etc.) and displayed with images, descriptions, pricing, and availability status.

- □ **Filter and Search:** Customers can filter products based on categories, materials, price range, or style. A search bar allows users to find specific items quickly by keywords (e.g., "gold ring" or "diamond necklace").
- □ **Product Details Page:** Clicking on a product brings up a detailed page with more information, including high-quality images, detailed descriptions, available sizes (if applicable), and price. Customers can also see product reviews and ratings.

3.2.2Shopping Cart and Checkout

- □ **Add to Cart: Customers can add items to their shopping cart by selecting the desired quantity and any applicable variations (e.g., size or color).**
- □ **View Cart: Customers can view and modify the contents of their shopping cart at any time, including adjusting quantities or removing items.**
- □ **Checkout Process: A multi-step process that includes entering shipping information, selecting payment methods, reviewing the order, and confirming the purchase. Payment options include credit cards, PayPal, and other integrated payment gateways.**
- □ **Order Confirmation and Notifications: After a successful transaction, the customer receives an order confirmation page along with an automated email or SMS notification. This includes the order details, shipping information, and estimated delivery date**





WOMAN

TITANIUM EARINGS IN GED PALLADIUMFINISH ...

$5900

★★★★★

Add To Cart

WOMAN

TITANIUM EARINGS IN GED PALLADIUMFINISH ...

$4399

★★★★★

Add To Cart

WOMAN

TITANIUM EARINGS IN GED PALLADIUMFINISH ...

$5599

★★★★★

Add To Cart

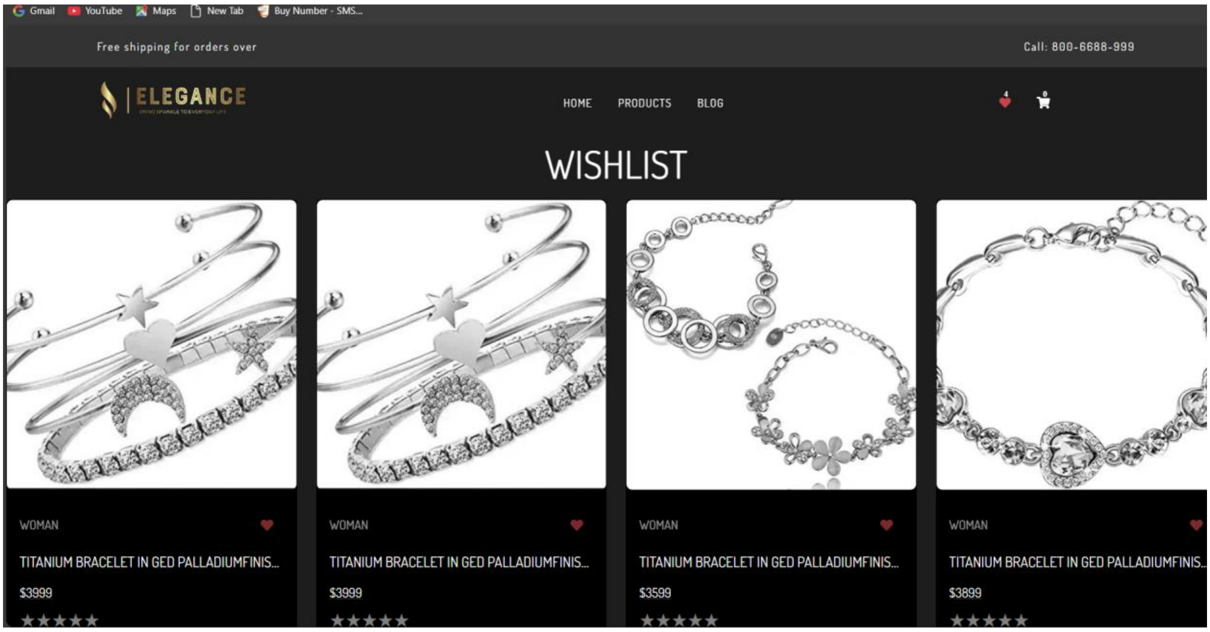
WOMAN

TITANIUM EARINGS IN GED PALLADIUMFINISH ...

$3499

★★★★★

Add To Cart



Free shipping for orders over

Call: 800-6688-999

WISHLIST

WOMAN

TITANIUM BRACELET IN GED PALLADIUMFINIS...

$3999

★★★★★

WOMAN

TITANIUM BRACELET IN GED PALLADIUMFINIS...

$3999

★★★★★

WOMAN

TITANIUM BRACELET IN GED PALLADIUMFINIS...

$3599

★★★★★

WOMAN

TITANIUM BRACELET IN GED PALLADIUMFINIS...

$3899

★★★★★

3.5 Non-Functional Requirements

Non-functional requirements describe the quality attributes that the system must possess to ensure that it performs effectively and securely. These requirements focus on areas such as system performance, reliability, security, and user experience. Below are the non-functional requirements for the Jewellery E-commerce Web Application.

3.5.1 Performance

The **Performance** requirements define the expected speed, responsiveness, and scalability of the application. These factors are crucial for ensuring that users have a seamless experience, even as the system grows in complexity and usage.

- **Response Time:**
  The application must respond to user requests within 2 seconds under normal load. This includes loading pages, submitting forms, and interacting with dynamic content such as appointment calendars. Any action that requires a server response (such as booking an appointment or updating the calendar) should not exceed this time limit.
- **Scalability:**
  The system must be able to scale horizontally, meaning it can handle an increase in the number of users or data without degradation in performance. This includes being able to support thousands of patients and doctors using the system concurrently.
- **Load Handling:**
  The application should be able to handle up to 10,000 concurrent users without significant performance issues. The backend should be optimized for high concurrency, ensuring that the system remains responsive even during peak usage times (such as early mornings when patients book appointments).
- **Availability:**
  The system should have a **99.9% uptime** to ensure that it is reliably available for users. This can be achieved through regular monitoring, backups, and the use of high-availability configurations (e.g., load balancing and failover systems).
- **Database Performance:**
  The MongoDB database should be optimized to perform quick queries and updates, particularly for frequently accessed data such as doctor schedules and patient records. Indexing and efficient query patterns will help improve the speed of data retrieval.
- **Mobile and Browser Optimization:**
  The application must be optimized for both desktop and mobile browsers, ensuring that it loads quickly and performs well on a range of devices. Mobile responsiveness should not compromise the application's performance, and the UI should adapt to smaller screens without causing delays.

3.5.4 Security

Security is a critical aspect of the Jewellery E-Commerce Web Application, given the various ornaments. The following security requirements are essential to protect both patient data and system integrity.

1. **Data Encryption:**
   - **SSL/TLS Encryption: All sensitive data (such as user login credentials, payment information, and personal details) should be transmitted over secure channels using SSL/TLS encryption. This ensures that the**

information cannot be intercepted by malicious actors during the transaction process.

- **End-to-End Encryption for Payment Processing: Payment information, including credit card details, should be encrypted and processed through secure payment gateways to ensure that user financial data is protected.**

2. **Authentication and Authorization:**
   - **Multi-Factor Authentication (MFA): The application should require MFA for both administrators and customers when accessing sensitive parts of the site (e.g., account management or order history).**
   - **Role-Based Access Control (RBAC): Users (customers, admins, support staff) should be assigned roles with specific access levels to ensure that only authorized users can modify products, process payments, or manage customer accounts.**
   - **Session Authentication: Secure session management should be implemented using JWT (JSON Web Tokens), which helps securely authenticate users during their session. Tokens should be set to expire after a certain period of inactivity to prevent unauthorized access.**
3. **Password Security:**
   - **Secure Password Storage: User passwords should be hashed using secure algorithms (such as bcrypt or Argon2) before being stored in the database. This prevents unauthorized access in case of data breaches.**
   - **Password Strength Enforcement: Customers and admins should be required to choose strong passwords (e.g., with a mix of letters, numbers, and special characters) and be provided with secure options for password reset (through email or SMS verification).**
4. **Data Privacy and Compliance:**
   - **GDPR Compliance: The website should comply with GDPR for users based in the EU, ensuring that all personal and payment data is handled with strict privacy measures. Users should be given clear consent options for data processing and should have the right to access, modify, or delete their data upon request.**
   - **PCI-DSS Compliance: For payment processing, the website should adhere to the Payment Card Industry Data Security Standard (PCI-DSS), which governs the secure handling of credit card information to prevent fraud and data breaches.**
5. **Session Management:**
   - **Automatic Session Expiry: User sessions should automatically expire after a set period of inactivity. This helps prevent unauthorized access in case users forget to log out.**
   - **Secure Cookies for Session Identification: Session identifiers should be stored in secure, HttpOnly cookies to prevent access to the cookies by client-side scripts.**
6. **Access Control:**

   - Granular Role Permissions: The application should define roles and permissions for different types of users. For instance, customers should only be able to view their orders, while admins should be able to manage products, process refunds, and view financial data.
   - Access Restrictions for Admins: Admins should be given only the necessary access required to manage the site, such as handling orders, managing the inventory, and viewing customer support queries, while preventing unnecessary access to sensitive customer data.
7. Audit Logs and Monitoring:
   - Activity Logging: The system should maintain detailed logs of all user activities, including login attempts, product changes, payment actions, and customer support interactions. Logs should be stored securely and be regularly reviewed for suspicious activity.
   - Monitoring for Suspicious Activity: Automated tools should be implemented to monitor for unusual patterns, such as multiple failed login attempts, unusual order activity, or irregular IP address access, which may indicate a potential security breach.
8. Backup and Data Recovery:
   - Regular Backups: Regular backups should be taken to ensure that user data, order details, and product catalog information are preserved in case of a server crash or cyberattack.
   - Encrypted Backups: Backup data should be encrypted to protect it from unauthorized access.
   - Disaster Recovery Plan: A comprehensive disaster recovery plan should be in place, allowing for quick recovery of critical website functionalities, such as the product catalog, payment processing, and customer accounts.
9. Security Testing and Vulnerability Scanning:
   - Regular Security Audits: The website should undergo periodic security audits to identify and address potential vulnerabilities, including SQL injection, cross-site scripting (XSS), and cross-site request forgery (CSRF).
   - Vulnerability Scanning Tools: Automated vulnerability scanners should be integrated into the development pipeline to detect weaknesses during the development phase and before production releases.

By following these security measures, the Jewellry E-Commerce Web Application will protect both the personal and financial information of its users while ensuring a secure shopping experience. These requirements also help maintain trust with customers, ensuring compliance with industry standards and regulations.

3.7 Design Constraints

Design constraints refer to the limitations or standards that guide the development and user experience of the Jewellery E-commerce Web Application. These constraints help ensure consistency, accessibility, and usability across the application.

1. User Interface (UI) Standards:

- **Consistency in Design:**
  The UI should maintain a consistent look and feel across all pages. This includes standardized colors, typography, button styles, and layout patterns. Consistent design improves user recognition and familiarity with the system.
- **Accessibility Compliance:**
  The design must comply with accessibility standards (such as **WCAG 2.1**), ensuring that the application is usable by people with disabilities. This includes providing alternative text for images, ensuring contrast ratios for readability, and supporting keyboard navigation for non-mouse users.
- **Responsiveness:**
  The UI should be fully responsive, adapting to various screen sizes, including mobile, tablet, and desktop displays. The layout, images, and interactive elements should scale appropriately to ensure usability on all device types.
- **Cross-Browser Compatibility:**
  The application should be compatible with modern browsers (such as Chrome, Firefox, Safari, and Edge). Testing should be done to ensure that UI elements render correctly and function as intended across these browsers.
- **Minimalist Design:**
  The UI should avoid clutter and focus on simplicity. Only essential elements and information should be displayed on each screen to reduce cognitive load and improve user navigation.

2. Technology Stack Constraints:

- **Frontend Framework:**
  The application is built using **React** for the frontend. The UI design must leverage React components and adhere to best practices for component-based architecture, making it modular, reusable, and maintainable.
- **Styling Library:**
  Tailwind CSS is used for styling React components. All styling should conform to the conventions of Tailwind CSS to ensure consistency, reduce styling redundancies, and simplify the design process.

3. Performance and Optimization Constraints:

- **Asset Optimization:**
  All images, scripts, and stylesheets must be optimized to minimize loading times. Large assets should be compressed, and unnecessary scripts or styles should be removed to improve performance.
- **Lazy Loading:**
  Components and assets, particularly those that are not immediately visible on the screen, should use lazy loading techniques. This helps reduce initial page load times, especially for users on slower networks.

4. Security Design Constraints:

- **Data Privacy:**
  UI components that display personal or sensitive information (such as medical

data or user credentials) must be protected against unauthorized access. Sensitive data should not be visible in URLs or cached in the browser.

- **Form Validation and Error Handling:**
  All input forms should incorporate both client-side and server-side validation to prevent invalid data entry. Error messages should be clear, guiding users to correct input mistakes without revealing system information.

5. User Experience (UX) Guidelines:

 Ease of Navigation:

- **Intuitive Layout: The website should have a clean and intuitive layout that guides users through their shopping journey. Key actions like browsing products, viewing categories, and checking out should be easily accessible from the homepage and clearly visible in the navigation menu.**
- **Simplified Checkout Process: The checkout process should be straightforward, with a clear path from cart to payment, allowing users to complete their purchase with as few steps as possible.**
- **Search Functionality: A prominent, easy-to-use search bar should be available to help users quickly find specific products by keyword, category, or filter.**

 Feedback Mechanism:

- **Instant Confirmation: Whenever a user adds an item to the cart, selects a product, or completes a purchase, immediate confirmation should be displayed (e.g., a popup or notification saying "Item Added to Cart"). Similarly, errors should be communicated clearly (e.g., "Please select a size").**
- **Visual Indicators for Actions: Interactive elements like product buttons (Add to Cart, Buy Now) should provide clear visual feedback. Buttons should change color or display a loading animation to indicate that an action is being processed.**
- **Order Status Updates: Provide real-time feedback on the status of orders, such as "Order Confirmed," "Shipped," and "Delivered," through notifications or updates on the user dashboard.**

 Visual Hierarchy:

- **Prioritize Key Information: Important elements such as product names, prices, and "Add to Cart" buttons should be the most prominent, using larger fonts, bold colors, or other visual cues to catch the user's attention.**
- **Clear Call-to-Actions (CTAs): Buttons like "Shop Now," "Add to Cart," and "Checkout" should be clearly distinguished from other content with contrasting colors and prominent placement.**
- **Minimal Distractions: The design should avoid unnecessary clutter. Users should be able to focus on the main actions (browsing products, making purchases) without being distracted by non-essential information.**

 Mobile Optimization:

- **Responsive Design: The website should be fully responsive, providing an optimized experience for both desktop and mobile users. Mobile users should have easy access to key features like product browsing, cart management, and payment.**

- **Touch-Friendly Interface: For mobile devices, ensure that buttons and links are large enough to be easily tapped, and that interactions like swiping through product images are smooth and intuitive.**

⍰ Consistency Across Pages:

- **Uniform Visual Style: The design should maintain consistent fonts, colors, and button styles throughout the site to reinforce brand identity and ensure a cohesive experience.**
- **Standardized Product Pages: Each product page should have the same layout with consistent placement for product images, descriptions, pricing, and options (e.g., size, color), so users can easily navigate between different product pages without confusion.**

3.9 Other Requirements

1. **Data Compliance and Privacy:**
   - **Data Protection Laws:**
     - The application must comply with relevant data protection regulations such as GDPR (General Data Protection Regulation) for European users and CCPA (California Consumer Privacy Act) for users in California, USA. This ensures that customer data is handled securely, and user consent is obtained for processing personal and payment information.
     - Customer personal details, purchase history, and payment information must be stored and processed in compliance with these laws, and users should have the option to review, delete, or modify their data in accordance with legal rights.
   - **Data Retention Policy:**
     - Customer data, including order histories, should be retained only for as long as necessary to fulfill legal, operational, and business needs. Data retention policies should specify how long user data is kept (e.g., for a specific number of years after the last purchase) and how outdated or unnecessary data will be securely deleted.
2. **Logging and Monitoring:**
   - **System Logging:**
     - Detailed logs should be maintained for key user actions, such as account logins, product purchases, and changes to account or payment information. These logs should be encrypted and stored securely to prevent unauthorized access, helping with troubleshooting and auditing if necessary.
   - **Error Tracking and Monitoring:**
     - Use of error tracking and monitoring tools (e.g., Sentry or Rollbar) to capture and report issues like payment failures, checkout issues, or product display errors. These tools will alert the development team to any critical issues impacting users, allowing for quick fixes and reducing downtime.
3. **Backup and Recovery:**
   - **Automated Backups:**

- The application should implement automated daily backups of customer data, product information, orders, and payment histories to secure off-site locations (such as cloud storage). This ensures that data is protected and can be recovered quickly in case of server failure or data loss.
- **Disaster Recovery Plan:**
  - A comprehensive disaster recovery plan should be in place, detailing the steps for system restoration in the event of a failure. This plan should be tested regularly to confirm that backups are recoverable and that the system can return to operation within an acceptable time frame.

4. **Performance Monitoring:**
   - **Server Performance Tracking:**
     - Monitoring tools should track key performance indicators such as server CPU usage, memory consumption, and website response times. This allows the development team to detect potential performance issues before they affect users, ensuring the website runs smoothly even during high traffic periods (e.g., holiday shopping).
   - **User Analytics:**
     - Collect non-personal analytics, such as page views, product views, and shopping cart activity, to understand user behavior. This data can be used to optimize the website’s layout, product offerings, and overall shopping experience, while adhering to privacy laws.
5. **Scalability and Future Expansion:**
   - **Horizontal Scalability:**
     - The application should be built to scale horizontally, meaning additional server resources (e.g., database instances or application servers) can be added as user traffic and sales volume increase. This ensures that the website can handle high traffic volumes, especially during sales events or peak shopping periods.
   - **Modularity for Feature Expansion:**
     - The website’s architecture should support modular development, allowing for the addition of new features (such as personalized product recommendations, a wish list feature, or virtual try-ons) without requiring a major overhaul of the codebase.
6. **Localization and Internationalization:**
   - **Multi-Language Support:**
     - The website should support multiple languages to cater to a diverse customer base. Users should be able to select their preferred language, and the site should adapt content, including product descriptions and checkout options, accordingly.
   - **Time Zone Support:**
     - For global customers, the website should handle different time zones, ensuring that promotions, offers, and delivery times are accurately displayed based on the user's location.
7. **Deployment and Environment Requirements:**
   - **Environment Setup:**
     - Separate environments for development, testing, and production should be maintained. This helps ensure that new features are tested

        thoroughly before deployment and that issues in the live environment can be quickly addressed.

    - **Continuous Integration/Continuous Deployment (CI/CD):**
        - CI/CD pipelines should be set up to automate the testing, building, and deployment process. This reduces human error, increases deployment speed, and ensures that updates are delivered smoothly to users.

8. **User Documentation and Training:**
    - **User Guides:**
        - Provide comprehensive user guides and FAQs for customers to understand how to browse products, make purchases, manage their accounts, and track orders. Clear, concise instructions will enhance the user experience and reduce support inquiries.
    - **Training for Admins and Staff:**
        - Provide training for internal staff and administrators on managing product listings, processing orders, handling customer inquiries, and maintaining the system. This ensures that the team can effectively run the e-commerce platform.

By addressing these project-specific requirements, the Jewelry E-Commerce Web Application will provide a secure, scalable, and user-friendly platform, ensuring smooth operation, user satisfaction, and future growth.

4.1 Data Flow Diagrams (DFDs)

Data Flow Diagrams (DFDs) for Jewelry E-Commerce Web Application

The Data Flow Diagrams (DFDs) for the Jewelry E-Commerce Web Application illustrate the movement of data between users, processes, and data stores. These diagrams offer a clear understanding of the system's data structure and interactions, detailing the flow of information from external entities like customers, administrators, and the system's various modules.

4.1.1 Level 0 DFD: Context Diagram

The Level 0 DFD, or Context Diagram, provides a high-level overview of the Jewelry E-Commerce Web Application. It shows how external entities interact with the system without detailing the internal processes.

External Entities:

- **Customers:** Users who browse products, add items to the cart, make purchases, and manage their accounts.
- **Admins:** Users who manage product listings, process orders, handle customer support, and access analytics.
- **Payment Gateway:** External system responsible for processing payments.

Data Flows:

- **Product Requests:** Customers request to view products, categories, and detailed information.
- **Order Submissions:** Customers submit their orders, including payment details.
- **Order Processing:** Admins receive order information, including customer details and order status.
- **Payment Authorization:** The payment gateway authorizes transactions and sends confirmation or failure messages.

4.1.2 Level 1 DFD: Major Functional Modules

The Level 1 DFD breaks down the system into major functional modules, showing the flow of data between them. Each module represents a core function, such as product browsing, order management, and payment processing.

Main Functional Modules:

1. **User Management Module:**
   - Handles user registration, login, and authentication.
   - Manages customer and admin accounts in the **User Database**.
   - Stores customer preferences, order history, and billing information.
2. **Product Management Module:**
   - Displays available products and categories to customers.
   - Retrieves product information from the **Product Database**.
   - Allows admins to add, update, or remove products from the catalog.
3. **Shopping Cart Module:**
   - Manages the products that customers add to their cart.
   - Communicates with the **Product Database** to display items and their prices.
   - Stores cart information temporarily in the **Cart Database**.
4. **Order Management Module:**
   - Manages order placement, modification, and tracking.
   - Stores completed orders in the **Order Database**.
   - Allows customers to view their past orders and current order status.
5. **Payment Processing Module:**
   - Handles payment authorization through the **Payment Gateway**.
   - Verifies and processes customer payment information.
   - Updates the order status after payment confirmation.
6. **Admin Dashboard Module:**
   - Provides admins with tools to manage orders, view product analytics, and access customer data.
   - Retrieves data from the **User Database**, **Order Database**, and **Product Database**.
   - Allows for reporting and insights generation.

4.1.3 Level 2 DFD: Detailed Data Flow for Order Management

The Level 2 DFD provides a more granular view of the **Order Management Module**. It illustrates the steps and interactions involved in placing, processing, and tracking customer orders.

Processes in Order Management:

1. **Order Request Processing:**
   - Receives order request from customers, including product details, quantities, and shipping information.
   - Validates availability and ensures the order is complete (e.g., products in stock).
2. **Payment Authorization:**
   - Sends payment request to the **Payment Gateway** with the customer’s payment details.
   - Receives payment confirmation or failure response.
   - Updates the **Order Database** with payment status.
3. **Order Confirmation:**
   - Upon successful payment, generates an order confirmation for the customer.

   - Sends a notification with order details, expected delivery time, and tracking number.
   - Notifies the admin to process and prepare the shipment.
4. **Shipment and Tracking:**
   - Admins process the order, prepare it for shipment, and update the order status in the **Order Database**.
   - Customers receive shipment tracking information via email or SMS.
5. **Order Retrieval and Updates:**
   - Customers can view and track their orders in the **Order Database**.
   - Updates in order status (e.g., shipped, delivered) are automatically displayed to the customer.

5.GitHub Link :

https://github.com/Piyush-xd/frontendreact

6.Deployed Link:

7.Acknowledgement Email

To formally acknowledge receipt of the project requirements and express gratitude for the opportunity to work on the Jewellery E-commerce Web Application, you can draft a professional acknowledgment email as follows:

Subject:Acknowledgment of Project Requirements – Jewelry E-Commerce Web Application

I would like to formally acknowledge receipt of the requirements and documentation for the Jewelry E-Commerce Web Application project. Thank you for entrusting me with this exciting opportunity to work on a platform dedicated to providing customers with a seamless and enjoyable online jewelry shopping experience.

I am eager to bring our shared vision to life by developing a secure, feature-rich, and visually appealing e-commerce platform that meets all functional and non-functional requirements outlined. Throughout the project, I will provide regular updates on progress and will reach out promptly if any clarifications are needed to ensure alignment with your expectations.

Thank you once again for this opportunity. Please feel free to reach out at any time with additional information or guidance.

Best regards,
Piyush Vishwakarma
Reg No:(12216662)