E-COMMERCE APPLICATION

Project Report Submitted

To

Gujarat University

In partial fulfilment of the requirements for

the award to the Degree of

5 YEAR INTEGRATED MASTER OF SCIENCE (COMPUTER SCIENCE)

SEMESTER – VI

GUIDED BY

Dr. Jigna Satani

Submited By:

Jhanvi Vekariya (60024)

Fenil Kachhadiya (60026)

Tanisha Rathod (60071)

DEPARTMENT OF COMPUTER SCIENCE
GUJARAT UNIVERSITY, AHMEDABAD

YEAR: 2024-25

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to all those who have provided immense help and guidance during our project. We extend our heartfelt thanks to our faculty guides, ***Dr. Jigna Satani*** and for providing us with a vision for the system and giving us the opportunity to undertake such a challenging and innovative project. We are deeply grateful for their guidance, encouragement, understanding, and insightful support throughout the development process.

Our special thanks to ***Dr. Hiren Joshi***, HOD, Department of Computer Science, Gujarat University, Ahmedabad, for giving us this opportunities.

We also extend our appreciation to ***Dr. Jyoti Pareek***, Our Course Coordinator, Department of Computer Science, Gujarat University, Ahmedabad, for their continuous encouragement and motivation.

Lastly, we wish to acknowledge everyone who has been associated with our project at any stage, even if their names do not appear in this acknowledgement. We are greatly indebted to each and every one of you.

With Thanks to All

TABLE OF CONTENT

PROJECT PROFILE

TITLE	DESCRIPTION
Project Name	E-commerce Application
Objective	Aim of our application is to offer users seamless shopping experience, enable customization, give detailed product information and secure transaction
Tools, Technologies and Framework	HTML,CSS,JAVASCRIPT,PHP, MYSQL,JAVA,XML, ANDROID
Duration	6 Months
Team size	3
Team members	Jhanvi Vekariya (60024) Fenil Kachhadiya (60026) Tanisha Rathod (60071)

ANALYSIS

A) REQUIREMENT ANALYSIS

1)Software requirement specification :

This SRS describes what is needed for an online jewellery shopping application. It explains what the application should do, how it will interact with others . This helps developers to understand what to build. This document explains all the features needed for the online jewellery shopping application. It includes managing user accounts, displaying products, shopping cart functionality, processing payments and admin controls.

A) Overall Description

1)Product Functions

- **User Registration and Login:** Allow users to create accounts and log in using email
- **Product Catalog:** Display a wide range of jewelry items categorized (e.g. necklaces, earrings, phone charms)
- **Product Browsing and Searching:** Users can browse and search for jewellery items.
- **Product Details:** Provide detailed product descriptions, available sizes, materials, and other specifications.
- **Shopping Cart Management:** Facilitate adding, removing, and updating items in the cart and also display subtotal, applicable taxes, and total cost dynamically.
- **Checkout and Payment:** Secure checkout process and payment options.
- **Customer Reviews:** Users can leave reviews and ratings for products.
- **Notifications and Alerts:** Push notifications for discounts, seasonal offers, and order updates.
- **Customization Features (Future Scope):** Allow users to design personalized jewelry by selecting materials, beads, charms, etc.
- **Order Tracking (Future Scope):** Allow users to track order statuses like "Processing," "Shipped," and “Delivered”.
- **Admin Dashboard:** Admins can manage products, orders, and users.

2). User Characteristics

- **Customers:** People who browse, select, and purchase jewellery items.
- **Administrators:** People who manage the product catalog, process orders, and handle customer inquiries.

B) Specific Requirements

The project requirements are categorized into functional and non-functional needs to ensure a comprehensive understanding of the system's expectations.

1.1 Functional Requirements

These requirements define the primary operations and functionalities of the e-commerce platform:

1) User Management

- **Registration:** Users can sign up with a username, password, and email.
- **Login:** Users can log in with their credentials.
- **Profile Management:** Users can update personal information like address and contact details.

2) Product Management

- **Catalog Display:** Products are displayed in categories with options to sort and filter.
- **Product Search:** Users can search for products by keywords.
- **Product Details:** Each product page shows detailed information and images.

3) Shopping Cart

- **Adding/Removing Items:** Users can add items to their cart and remove them if needed.
- **View Cart:** Users can view what's in their cart anytime.
- **Checkout:** Users can proceed to checkout, review their order, and enter payment information.

4) Payment Integration

- **Payment Options:** Support payment methods like RazorPay, UPI, credit/debit cards, and net banking.

5) Order Management

- **Order History:** Users can view their past orders.

6) Review System

- **Customer Feedback:** Users can leave reviews and ratings for products.

7) Administrative Functions

- **Admin Dashboard:** Admins have an interface to manage products, orders, and users.
- **Product Management:** Admins can add, update, and delete products.
- **Order Processing:** Admins can view and update the status of orders.
- **User Management:** Admins can manage user accounts, including suspending or deleting accounts.
- **Notifications and Alerts:** Push notifications for discounts, seasonal offers, and order updates.

1.2 Non-Functional Requirements

1. Security Requirements

- All user data should be encrypted during transmission and storage.
- Regular security checks should be done to find and fix vulnerabilities.

2. Usability Requirements

- The application should be easy to use and navigate.

3. Reliability Requirements

- The application should be available 99.9% of the time.

4. Maintainability Requirements

- The system should be easy to update and maintain

2) FEASIBILITY STUDY:

1. Economic Feasibility

Economic feasibility checks if the project will be worth the money we might spend.

1. Benefit Analysis

- **Increased Sales:** More sales from reaching customers online.
- **Market Expansion:** Ability to sell to a wider audience.
- **Customer Convenience:** Easier shopping experience for customers.

1.2 Revenue Potential

- **Sales Increase**: How the app can boost jewellery sales by providing a better shopping experience.
- **New Customer Acquisition**: The potential of the app to attract more customers through convenience and accessibility.

1.3 Return on Investment (ROI)

- Estimating the profit from the website based on increased sales .

2. Technical Feasibility

Technical feasibility checks if the technology needed is available and suitable.

2.1 Technology Assessment

• Platform Choice

- **Web**: Does the application need to be web-based? (e.g., built using HTML, CSS, JavaScript, PHP).
- **Mobile**: Is it intended for mobile devices (Android)? Will you use native development (Java)

• Backend and Database

- Are the backend technologies (e.g., PHP) suitable for handling inventory, user data, and transactions?
- Is there a database system (e.g., MySQL) capable of managing customer details, product catalogs, and order histories?

3. Schedule Feasibility

Schedule feasibility checks if the project can be completed on time.

3.1 Project Timeline

- **Planning:** Initial planning and requirement gathering.
- **Design:** Creating the website's design.
- **Development:** Building the website.
- **Testing:** Ensuring everything works correctly.
- **Deployment:** Launching the website.

3.2 Milestones

- **Design Completion**
- **Development Completion**
- **Successful Testing**
- **Website Launch**

4. Resource Feasibility

Resource feasibility checks if the necessary resources are available.

4.1 Human Resources

- **Development Team:** Developers and designers to build and maintain the application.

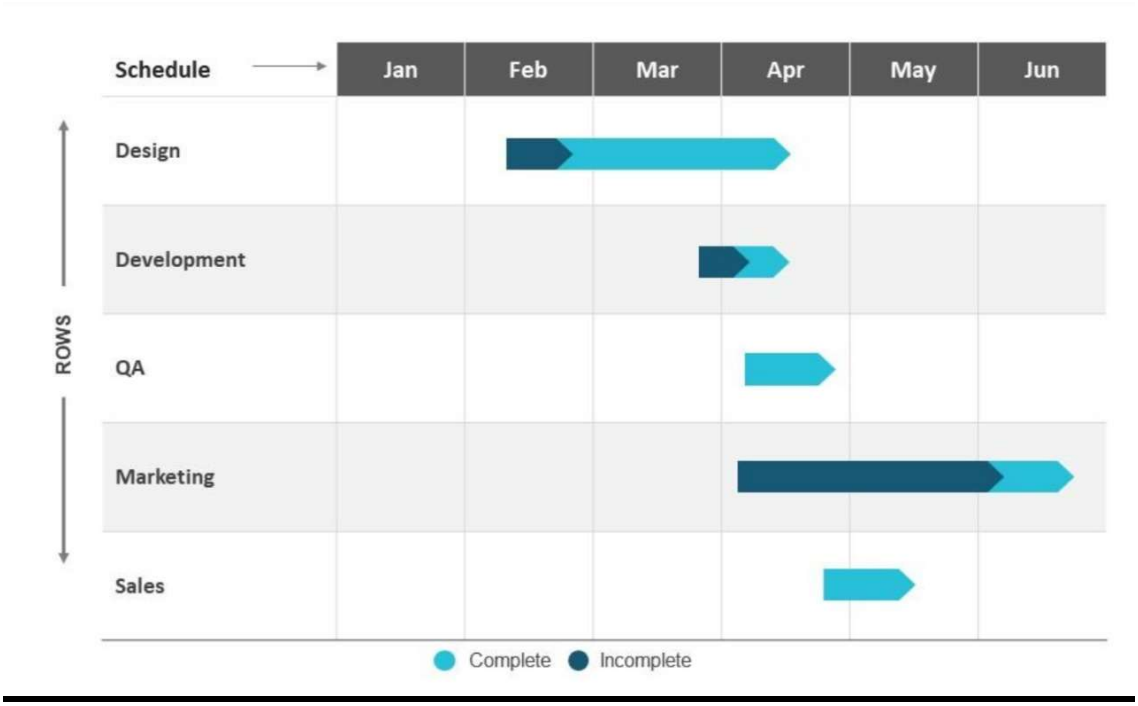
4.2 Material Resources

- **Software:** Tools for development and design.

4.3 Financial Resources

- **Funding:** Sources of money, whether from savings or investing by own money

B) PROJECT TIMELINE CHART



C) TOOLS AND TECHNOLOGIES USED

FRONTEND	**HTML,CSS,JAVASCRIPT,XML**
BACKEND	**PHP,MYSQL,JAVA**

D) SOFTWARE DEVELOPMENT MODEL USED

To solve an actual problems in an industry, software developer or a team of developers must integrate with a development strategy that include the process, methods and tools layer and generic phases. This strategy is often referred to a process model or a software developing paradigm. Our project follows the waterfall model. The steps of waterfall model are:

1. Requirement Gathering

- A detailed study of existing e-commerce platforms (e.g., Tanishq, Giva) provided insight into user expectations.
- Initial requirements were documented and reviewed.

2. System Design

- Use case diagrams, system flow diagrams, and ER diagrams were created to visualize system architecture and data flow.

3. Implementation

- Frontend and backend modules were developed sequentially.
- Unit testing was performed during this phase.

4. Testing

- Comprehensive testing was conducted after the implementation phase to ensure all functionalities were working as intended.

5. Deployment

- The platform was deployed on a local server for demonstration and testing.

E) WORK DISTRIBUTION

WORK DIVISION	TEAM MEMBERS
JAVA , XML ,ANDROID, MYSQL	FENIL KACHHADIYA TANISHA RATHOD
HTML , CSS , PHP , ANDROID	FENIL KACHHADIYA JHANVI VEKARIYA

Design

A) SYSTEM ARCHITECTURE

System Overview :

The system architecture outlines the overall structure of the application, including its components and their interactions. The architecture ensures scalability, maintainability, and efficient communication between layers.

System Architecture Description :

- Provide a detailed breakdown of the system architecture layers:

1.Frontend Layer

- This layer provides the user interface and handles interactions for various user types:
  - Visitor Interface: Allows browsing categories, products, and feedback.
  - Customer Interface: Includes functionalities for registration, login, product search, shopping cart, order tracking, and reviews.
  - Admin Interface: Enables management of products, categories, orders, and user feedback.

2. Backend Layer

- The backend layer handles business logic and ensures the smooth functioning of the application by managing requests from the frontend and interfacing with the database. Key responsibilities include:
  - User authentication and authorization.
  - Managing product inventory and categories.
  - Processing orders.
  - Handling reviews and feedback.

3. Database Layer

- This layer stores and organizes all application data. Key tables include:
  - Customer Table: Stores user profiles and credentials.
  - Product Table: Manages inventory and product details.
  - Order Table: Tracks purchase history and order status.
  - Payment Table: Records transaction details.
  - Feedback Table: Collects user reviews and feedback.

B) MODULE HIERARCHY

The system is divided into modules, each handling specific functionality. This modular design ensures ease of maintenance and scalability.

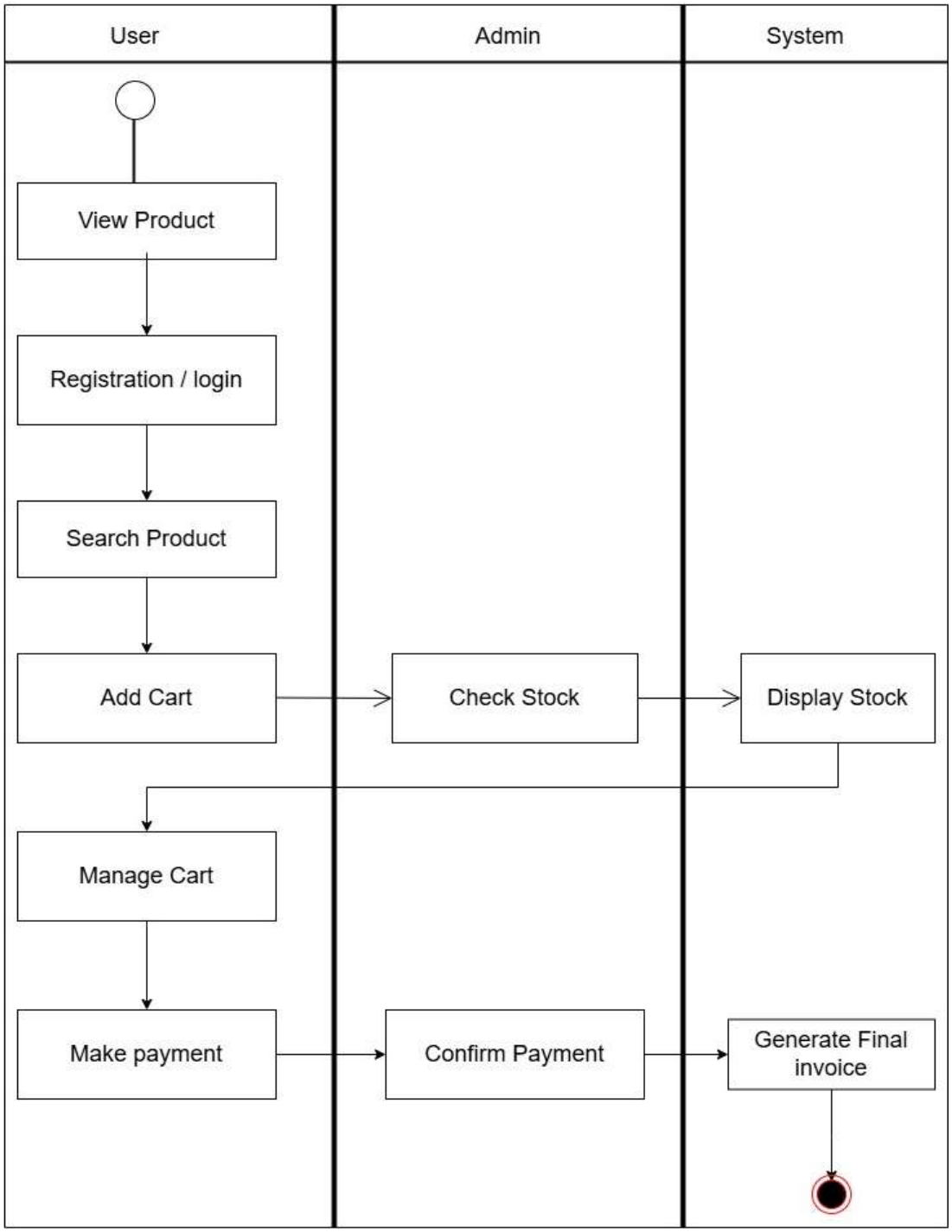
1. **User Management Module:**
   - This module manages:
     - User registration, login, and logout.
     - Profile management for customers.
     - Admin access control.
2. **Product Management Module**
   - Key features include:
     - Adding, updating, and deleting products.
     - Managing product images, descriptions, and prices.
     - Monitoring inventory levels to prevent stockouts.
3. **Order Management Module**
   - Handles the following:
     - Adding items to the shopping cart.
     - Processing orders and confirming payments.
     - Calculating total costs, including taxes and discounts.

4. **Review Management Module**
    - This module facilitates:
        - Collecting and displaying customer reviews.
        - Enabling customers to rate products.
        - Filtering inappropriate content for quality control.

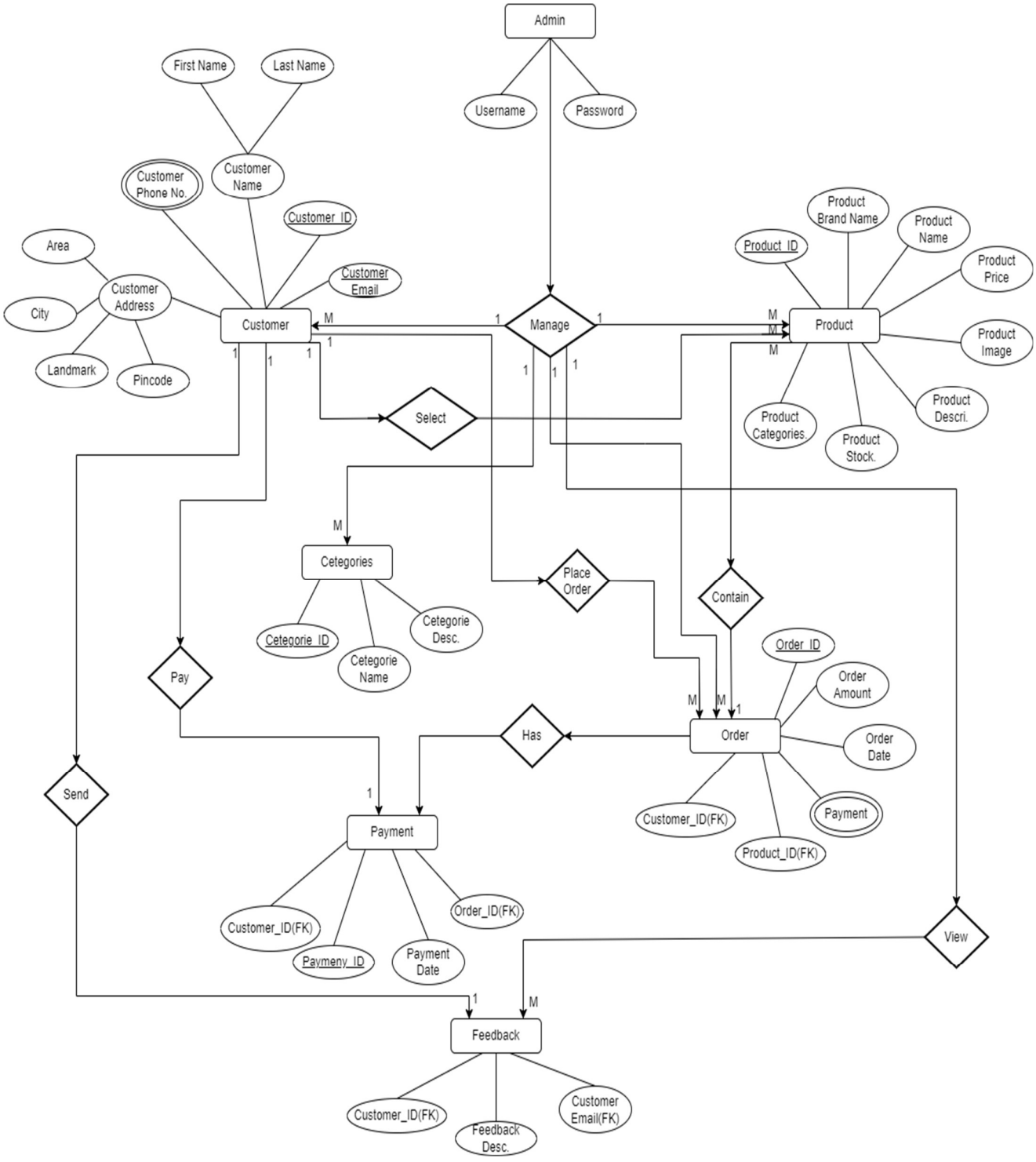
5. **Category Management Module**
    - This module is responsible for:
        - Defining and organizing product categories.
        - Allowing easy navigation for customers.
        - Updating category hierarchy as new products are introduced.

C) USE CASE DIAGRAMS

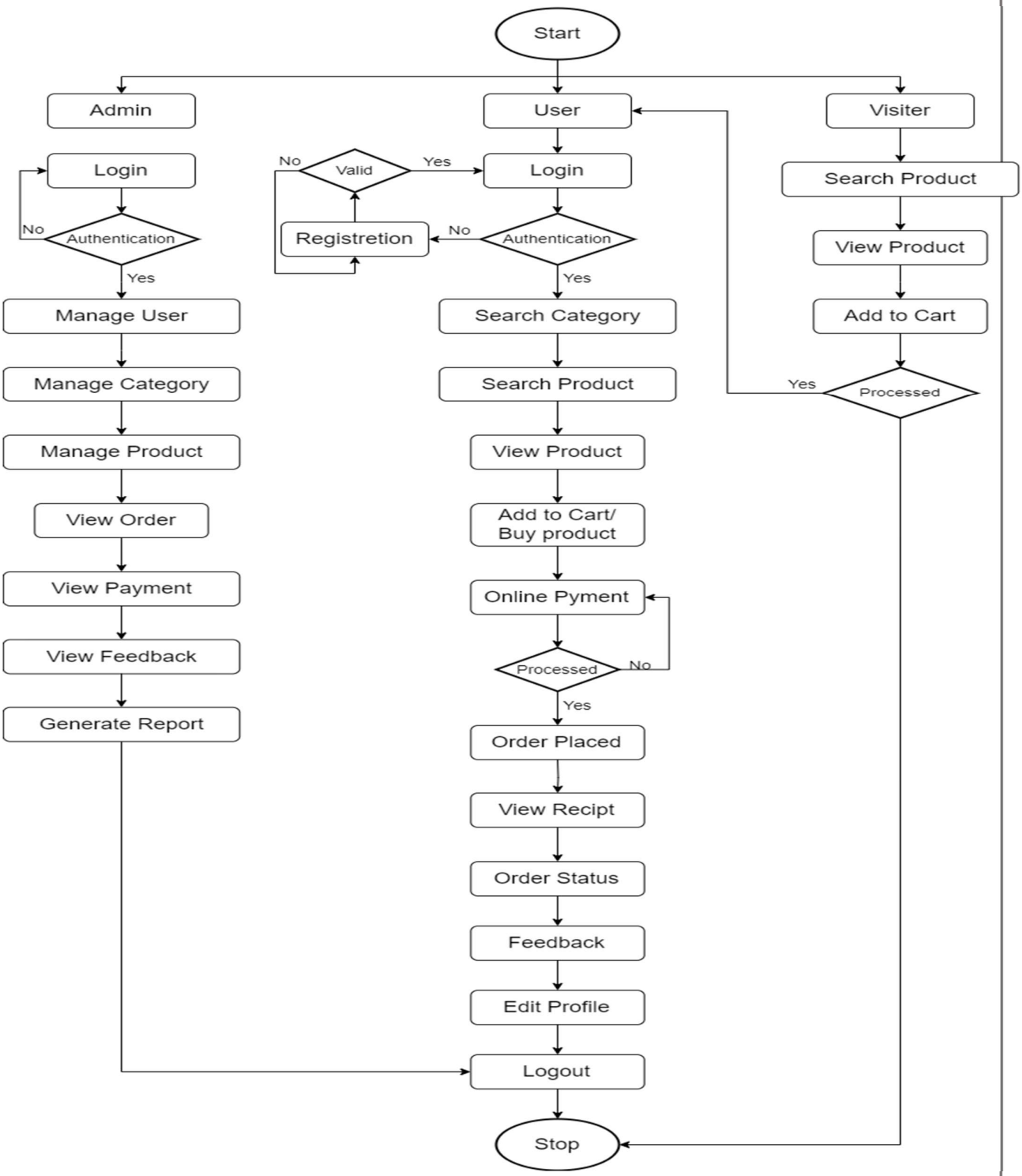
D) ACTIVITY DIAGRAM



E) ER DIAGRAM


Admin
Username
Password
First Name
Last Name
Customer Name
Customer Phone No.
Customer_ID
Customer Email
Area
Customer Address
City
Landmark
Pincode
Customer
Manage
Product
Product_ID
Product Brand Name
Product Name
Product Price
Product Image
Product Descri.
Product Stock.
Product Categories.
Select
Cetegories
Cetegorie_ID
Cetegorie Name
Cetegorie Desc.
Place Order
Contain
Pay
Order_ID
Order Amount
Order Date
Order
Has
Send
Payment
Customer_ID(FK)
Payment
Product_ID(FK)
Payment
Customer_ID(FK)
Order_ID(FK)
Paymeny_ID
Payment Date
View
Feedback
Customer_ID(FK)
Feedback Desc.
Customer Email(FK)

F) System Flow Diagram



G) DATA DICTIONARY

Table no : 1

Table name : Customer

Column Name	Type	Size	Description
id	Int	10	Id of the user /P.K
Name	Varchar	50	Name of the user
Surname	Varchar	20	User surname
Username	Varchar	30	User UserName
Password	Varchar	10	User Password
Email	Varchar	10	User EmailId

Table no :2

Table name : Cart

Column name	Type	Size	Description
id	Int	10	In this the id of the item is filled / P.K
jewel_id	Varchar	120	Name of the product id
Qty	Int	10	Quantity of the item

Table no :3

Table name : Product

Column Name	Type	Size	Description
Id	Int	10	Id / P.K
Prodname	Varchar2	50	Name of the product
Path	Varchar2	50	Path of the image of product
Category	Varchar		Category of the product
price	Decimal	10,2	Price of the product
Description	Varchar2	100	Details about the product
Type	Varchar2	50	Type of product
no views	Number		Number of views of the product
top sell	Varchar2	120	Top sell product
Path	Varchar2	200	Path of an image
grand_total	Decimal	10,2	Total amount payable

Table no : 4

Table name : Admin

Column Name	Type	Size	Description
id	Int	2o	In this id of Admin is to be filled / P.K and auto generated
email	Varchar	50	Admin Email Id
Password	Varchar	10	Password
Address	Varchar	120	Address of Admin
Mobile No	Varchar	120	Mobile No of Admin

7) IMPLEMENTATION

A. Tools and technologies used for implementation

The implementation phase of the e-commerce project for online jewelry shopping involves creating and integrating various technical components to bring the system to life. Here is a breakdown of the key elements involved in the implementation:

1. Frontend Development

The user interface (UI) was designed to ensure a smooth and engaging user experience.

Tools Used:

- HTML: For structuring the content of the web pages.
- CSS: For styling elements and creating visually appealing layouts.
- JavaScript: For adding interactivity, such as product search, animations, and real-time updates in the shopping cart.

Key Features:

- Responsive Design: Ensures the application adapts to various devices, including desktops, tablets, and smartphones.

2. Backend Development

The backend handles the core functionality and logic of the system.

Tools Used:

- PHP: Server-side scripting language used for processing user requests, managing sessions, and connecting to the database.
- MySQL: A relational database management system to store and retrieve structured data.

Key Features:

- User Management: Backend scripts handle registration, login, password recovery, and profile updates.
- Product Management: Admins can add, update, or delete product details.
- Order Management: Manages the shopping cart, processes orders, and generates invoices.
- Payment Processing: Implements secure communication with the payment gateway for transaction handling.

3. Database Design

A robust database was created to manage data efficiently.

Key Tables and Their Relationships:

- Customer Table: Stores user details with a unique Customer ID.
- Product Table: Maintains product information like name, price, description, and stock with Product ID as the primary key.
- Order Table: Tracks orders, linking customers and products using foreign keys.
- Payment Table: Records payment details tied to orders and customers.
- Feedback Table: Stores user reviews and ratings.

- Category Table: Organizes products into categories for easier navigation.

4. Payment Gateway Integration

The secure payment gateway (RazorPay) is integrated to handle financial transactions.

Steps in Integration:

- API Setup: Configured RazorPay API keys to enable secure communication.
- Transaction Flow: Implemented server-side verification to ensure the transaction's authenticity.
- Error Handling: Added mechanisms to address transaction failures or incomplete payments.
- Encryption: Secured sensitive user data during transactions.

5. Android Application

A mobile app was developed to complement the web application and cater to smartphone users.

Tools Used:

- XML: For designing layouts in the Android application.
- Java: For implementing the app's business logic and connecting it to the backend.

Key Features:

- Intuitive navigation for browsing categories and products.
- Integrated cart and checkout system.
- Notifications for order updates and promotions.
- Support for multiple payment options through the RazorPay SDK.

6. Testing and Debugging

Thorough testing was conducted throughout the implementation phase to ensure functionality and performance.

Types of Testing:

- Unit Testing: Verified individual modules like login, search, and payment processing.
- Integration Testing: Ensured smooth interaction between frontend, backend, and the database.
- Performance Testing: Assessed system responsiveness under varying loads.
- Security Testing: Checked for vulnerabilities in user authentication, data encryption, and payment processing.

B. Screen shots of Admin Pannel

Figure 7.1 Admin login page

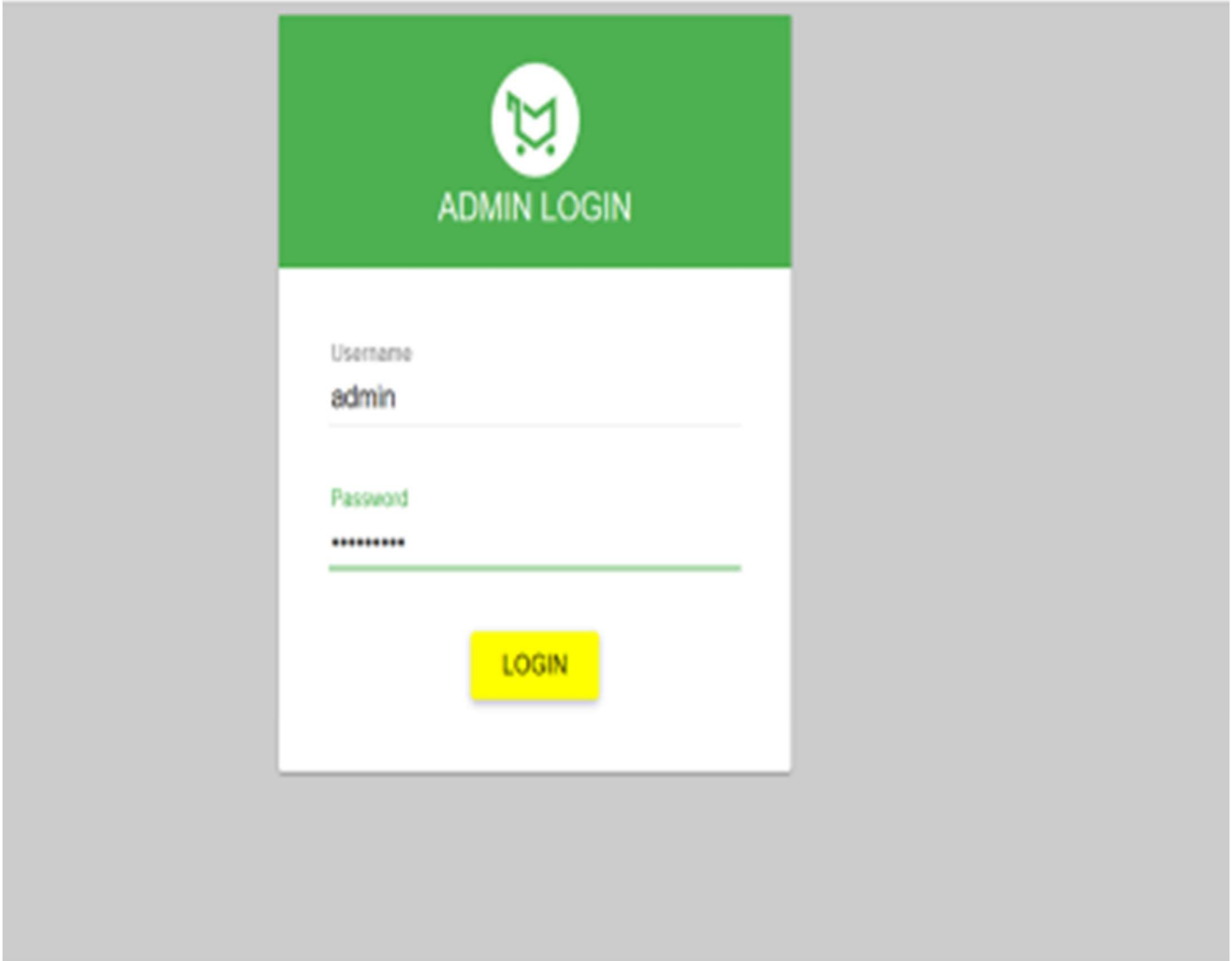


Figure 7.2 Admin dashboard

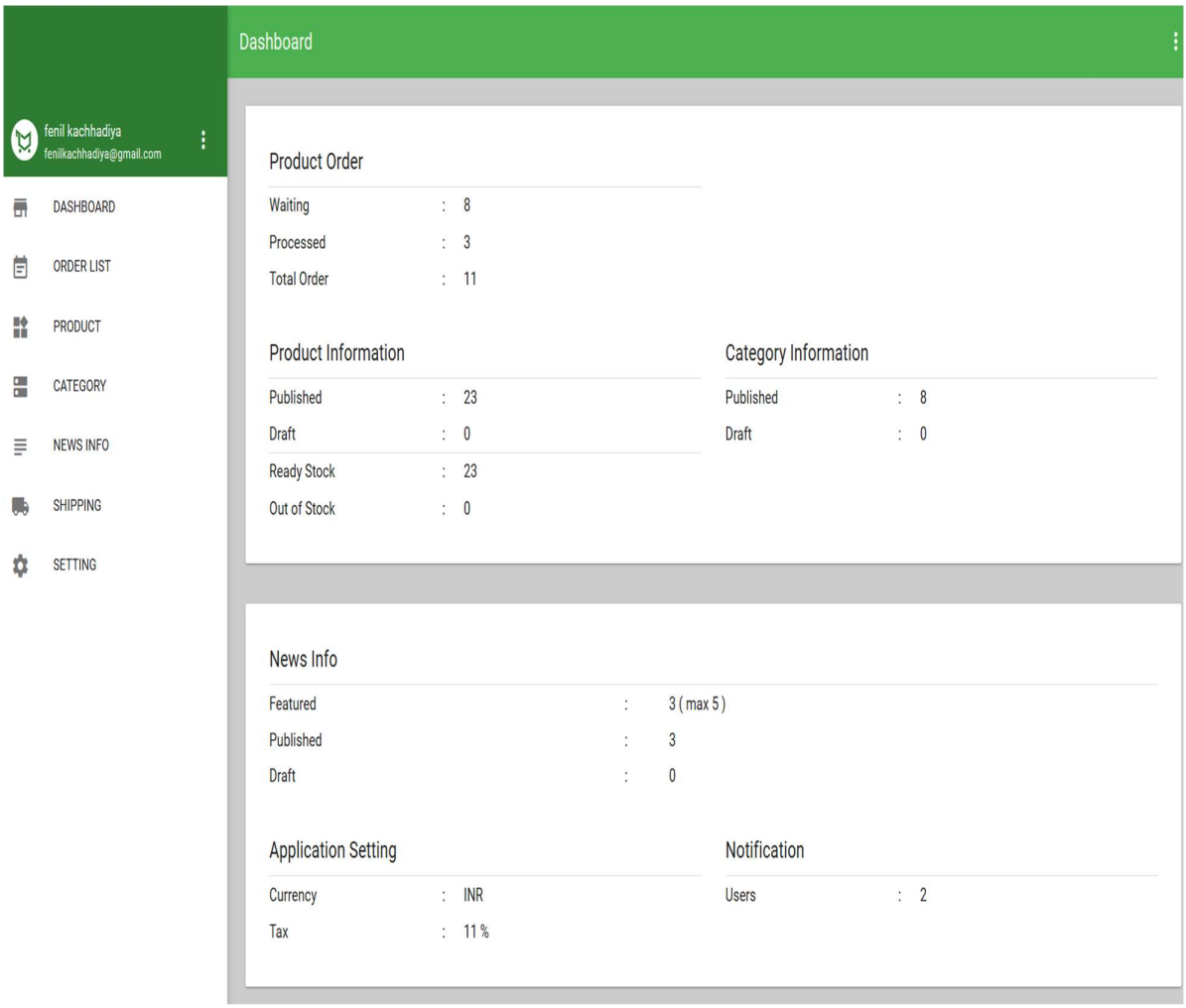
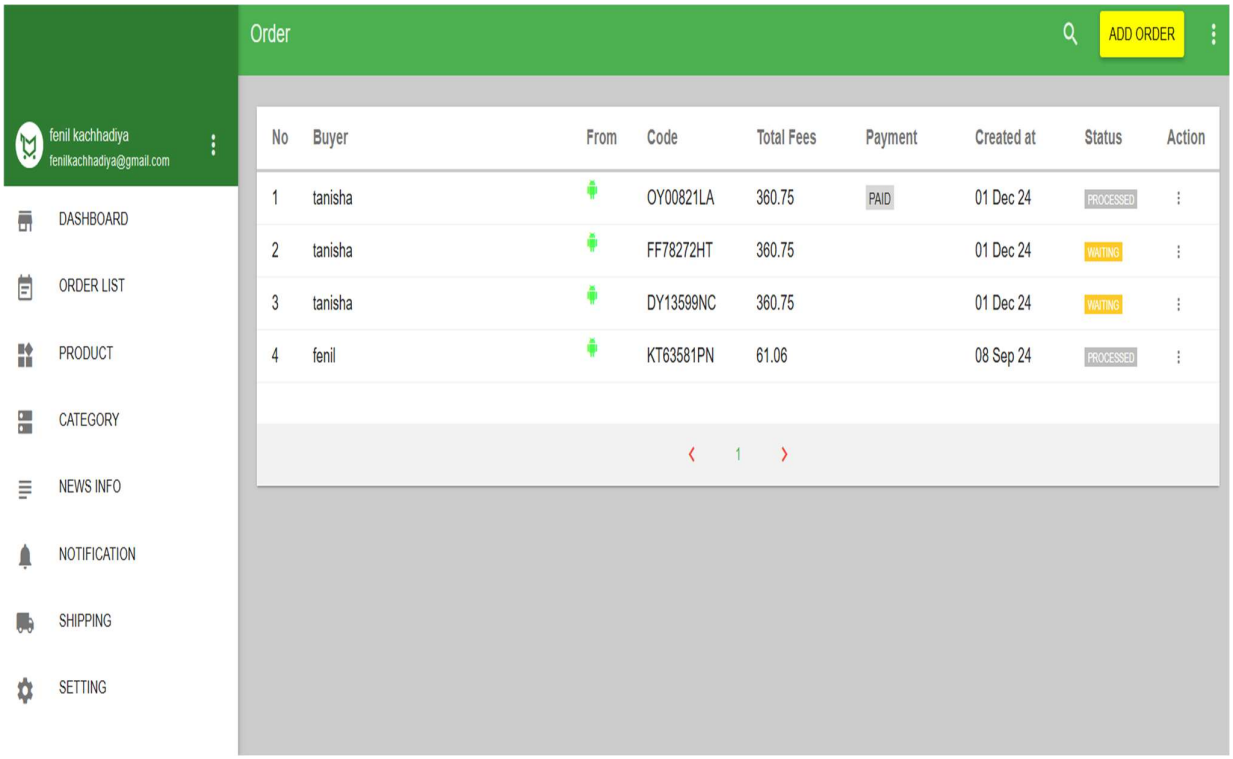


Figure 7.3 User order list



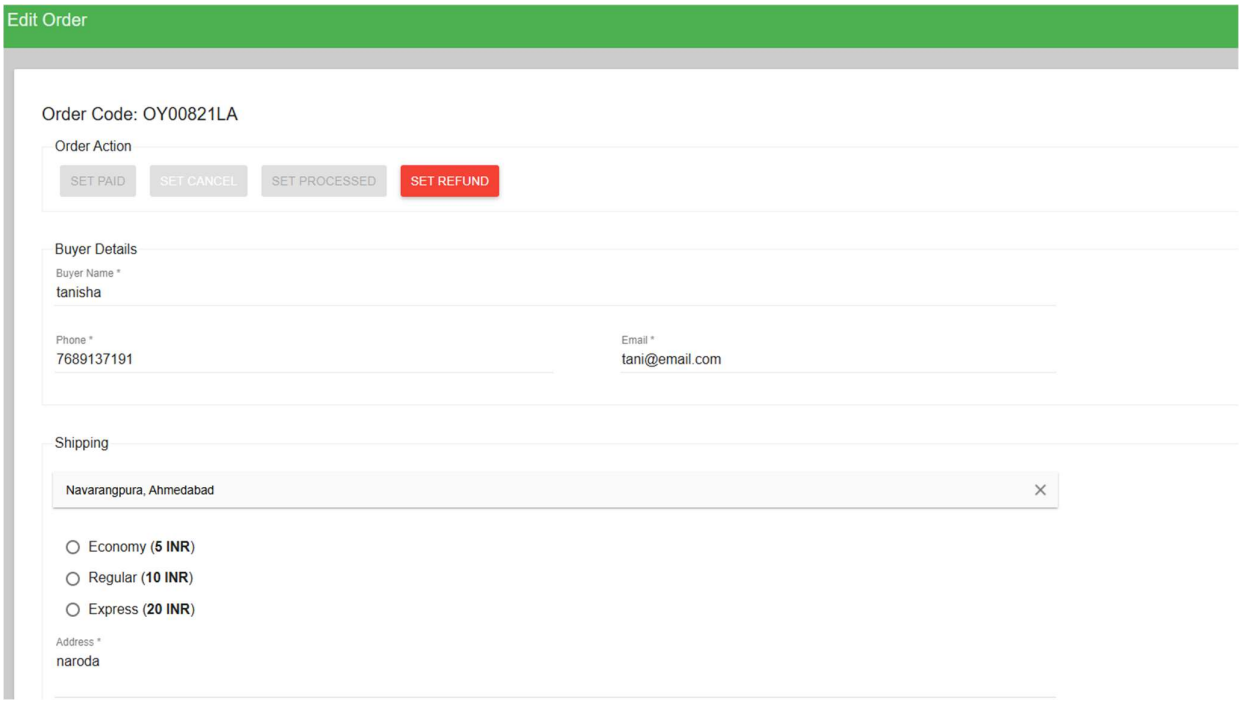


Figure 7.4 User order invoice

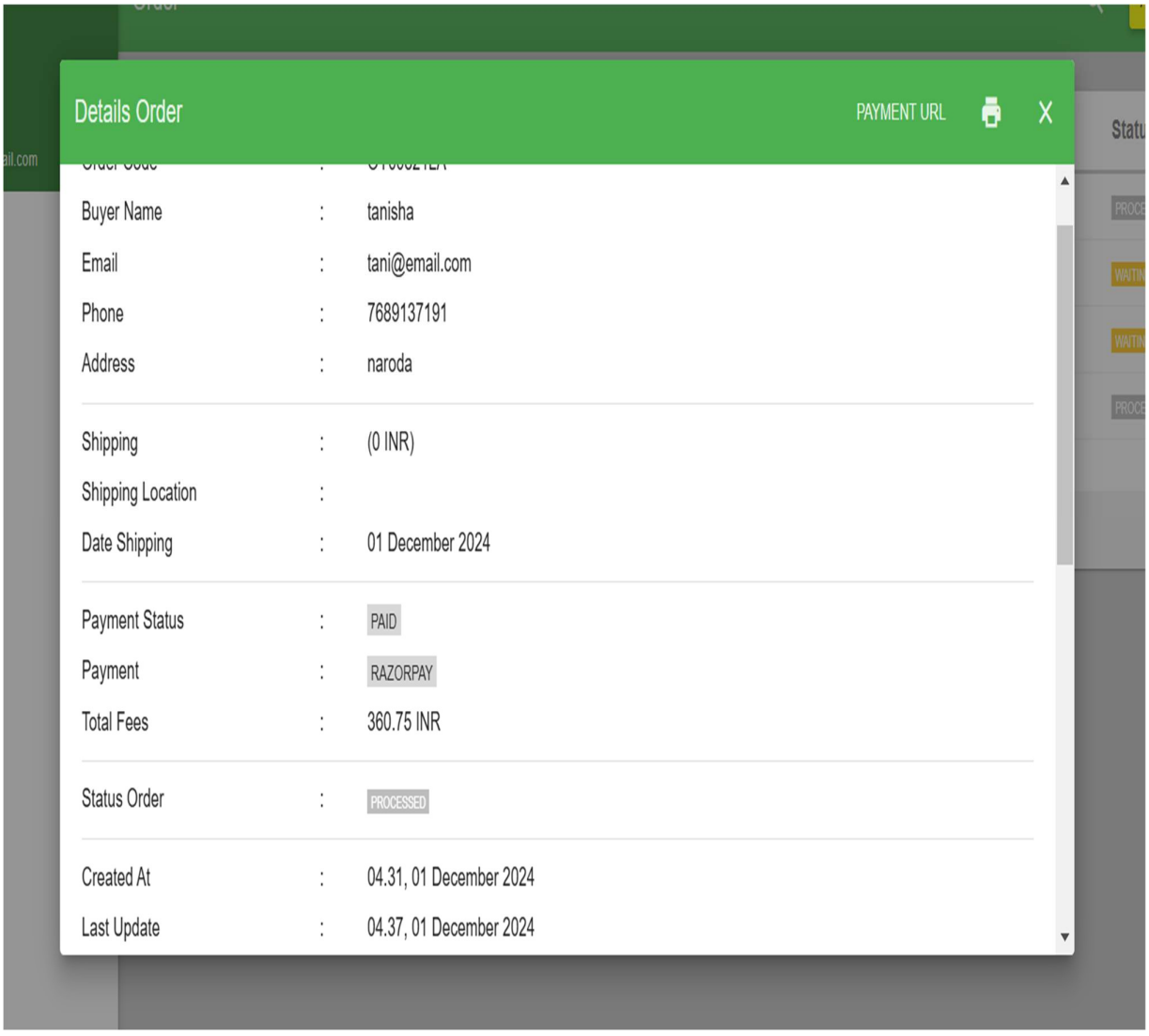
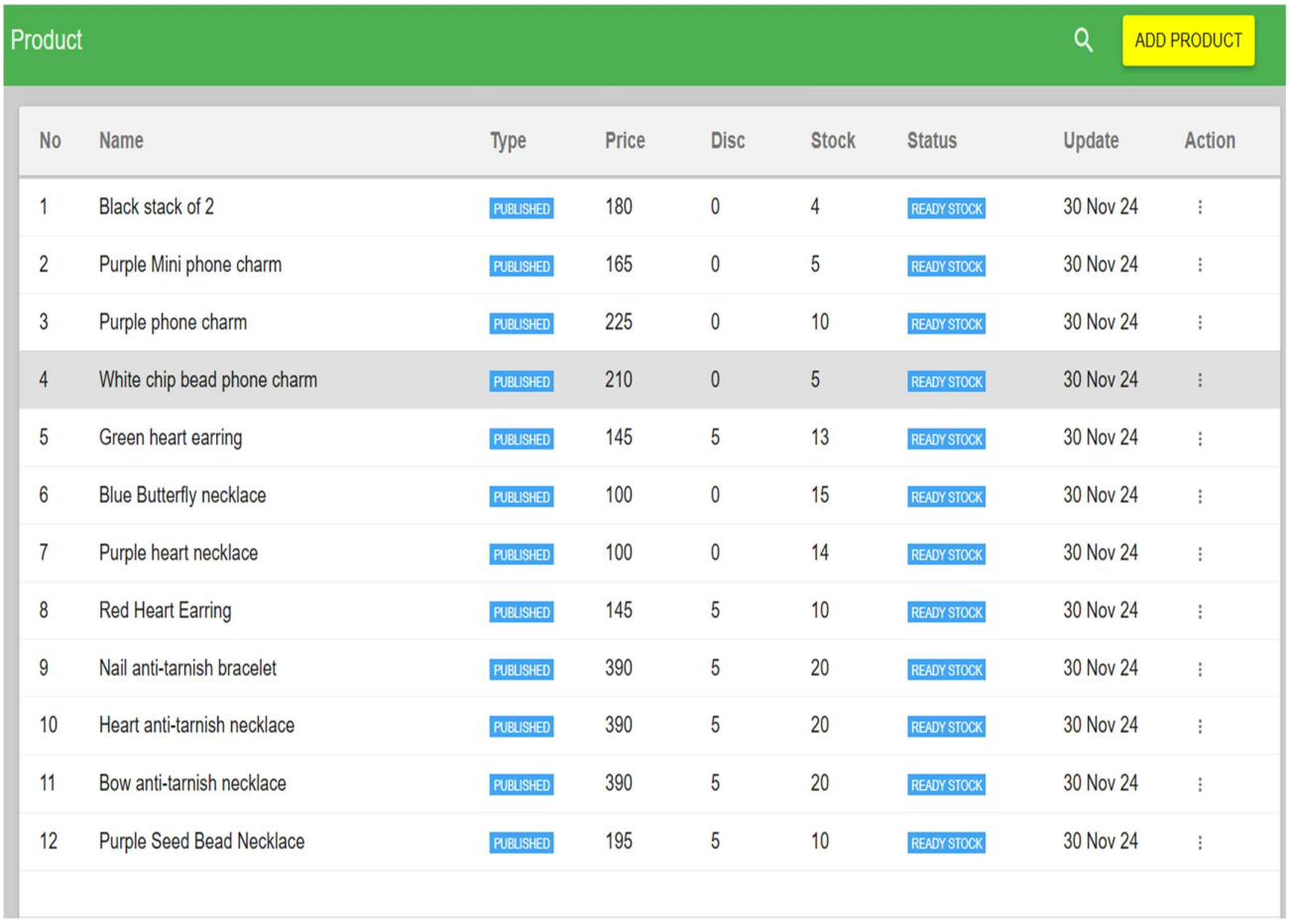


Figure 7.5 Product details



No	Name	Type	Price	Disc	Stock	Status	Update	Action
1	Black stack of 2	PUBLISHED	180	0	4	READY STOCK	30 Nov 24	⋮
2	Purple Mini phone charm	PUBLISHED	165	0	5	READY STOCK	30 Nov 24	⋮
3	Purple phone charm	PUBLISHED	225	0	10	READY STOCK	30 Nov 24	⋮
4	White chip bead phone charm	PUBLISHED	210	0	5	READY STOCK	30 Nov 24	⋮
5	Green heart earring	PUBLISHED	145	5	13	READY STOCK	30 Nov 24	⋮
6	Blue Butterfly necklace	PUBLISHED	100	0	15	READY STOCK	30 Nov 24	⋮
7	Purple heart necklace	PUBLISHED	100	0	14	READY STOCK	30 Nov 24	⋮
8	Red Heart Earring	PUBLISHED	145	5	10	READY STOCK	30 Nov 24	⋮
9	Nail anti-tarnish bracelet	PUBLISHED	390	5	20	READY STOCK	30 Nov 24	⋮
10	Heart anti-tarnish necklace	PUBLISHED	390	5	20	READY STOCK	30 Nov 24	⋮
11	Bow anti-tarnish necklace	PUBLISHED	390	5	20	READY STOCK	30 Nov 24	⋮
12	Purple Seed Bead Necklace	PUBLISHED	195	5	10	READY STOCK	30 Nov 24	⋮

Figure 7.6 Category details

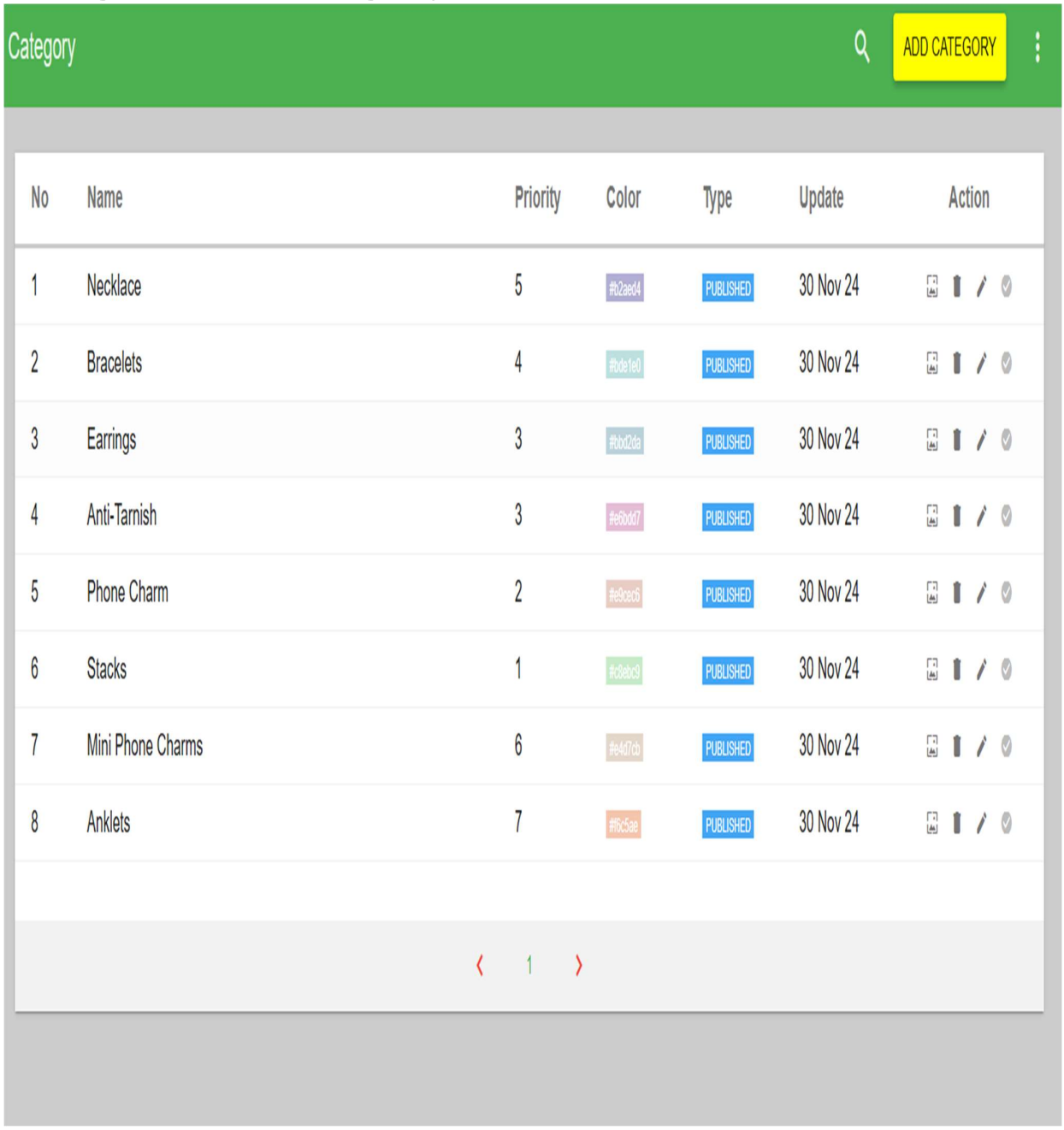
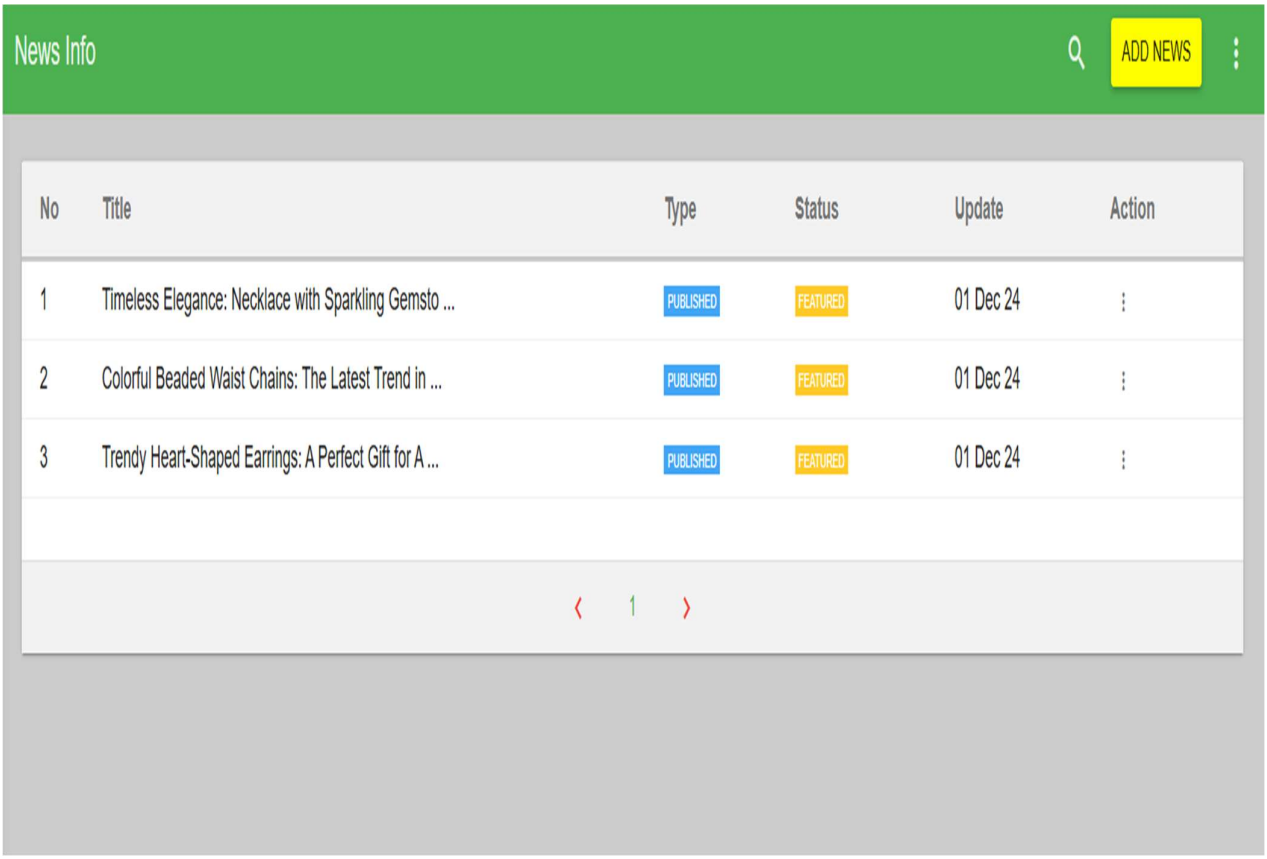
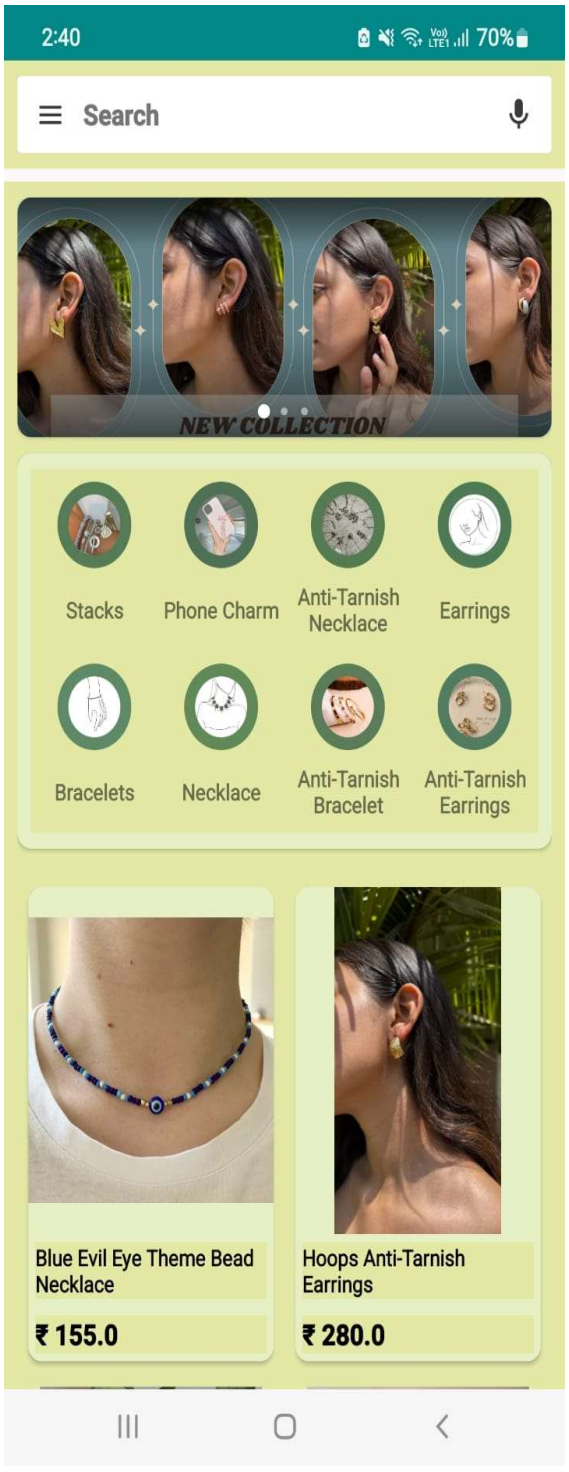


Figure 7.7 News information detail



C.Screen shots of the Developed Application



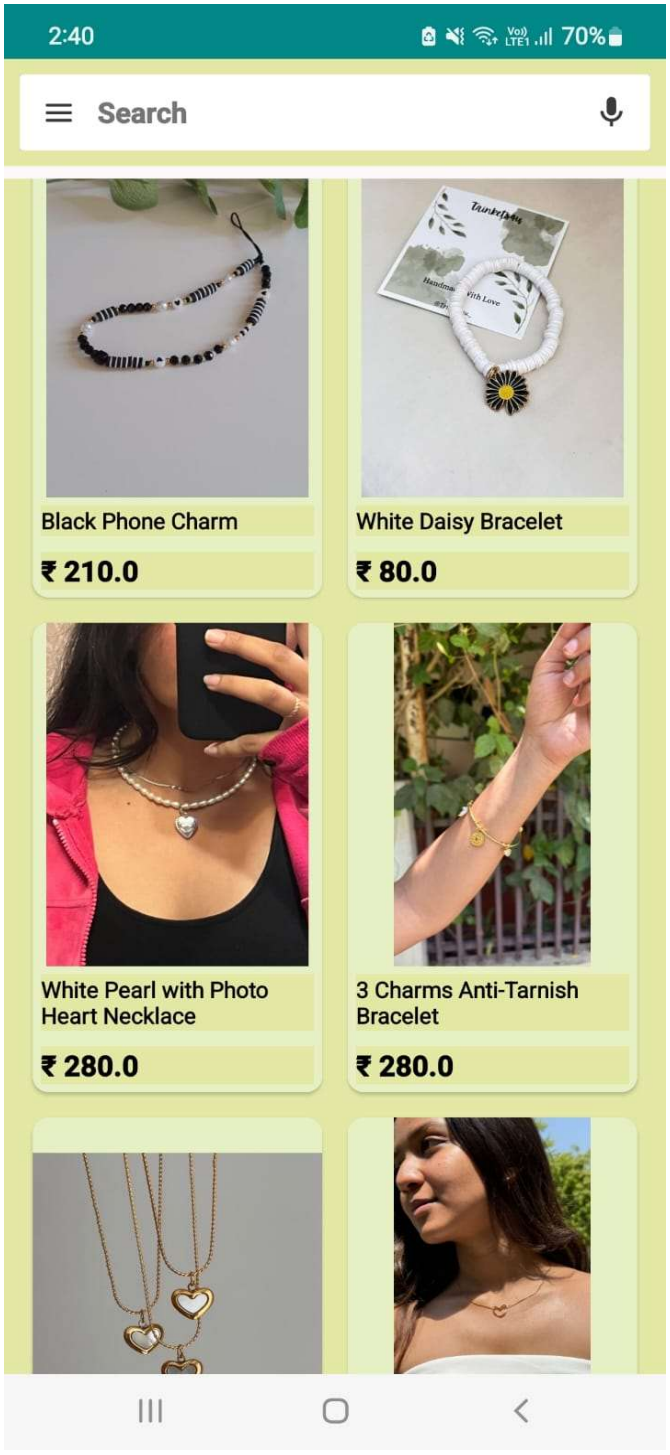


Figure 7.8 Android Application Views

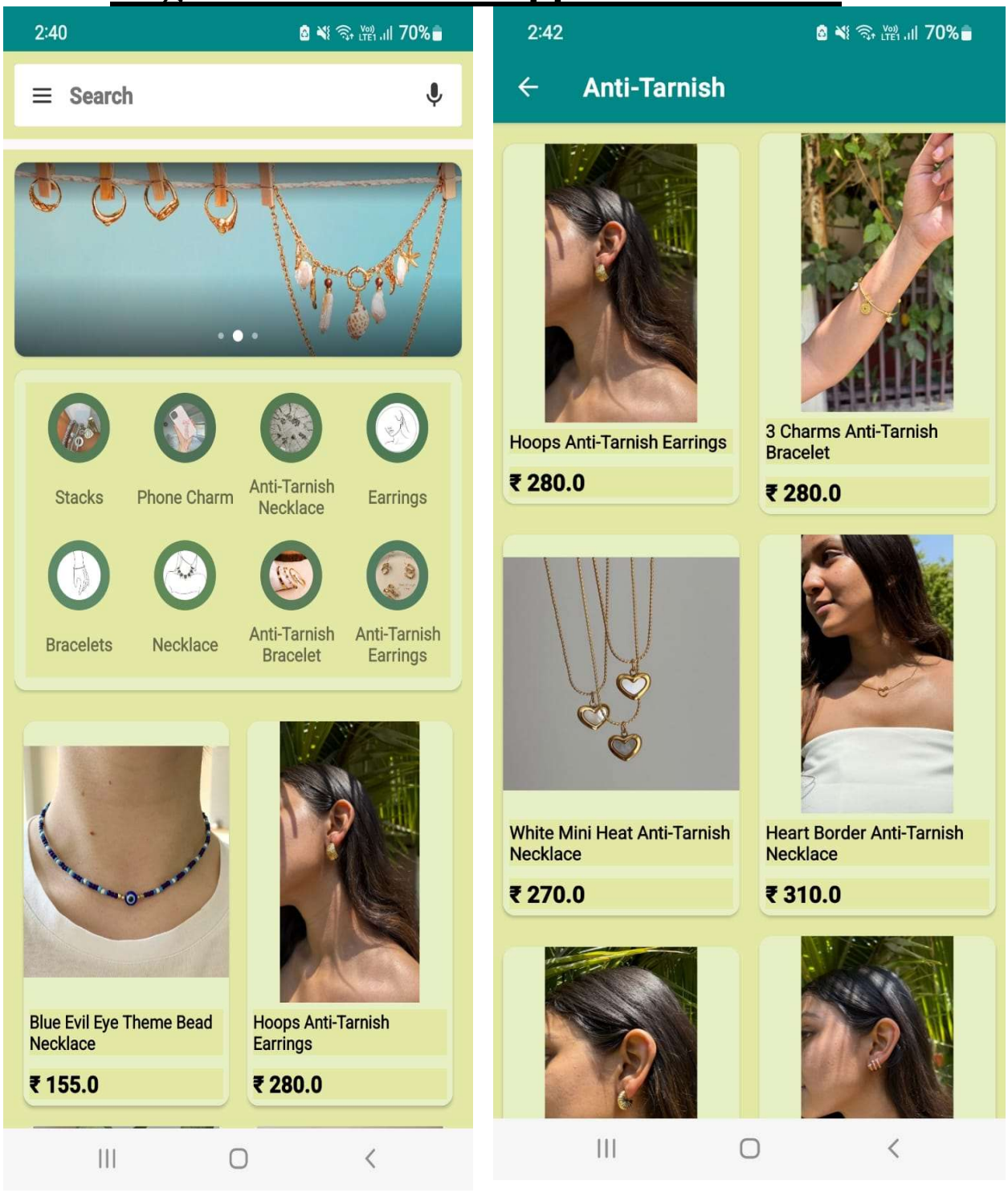
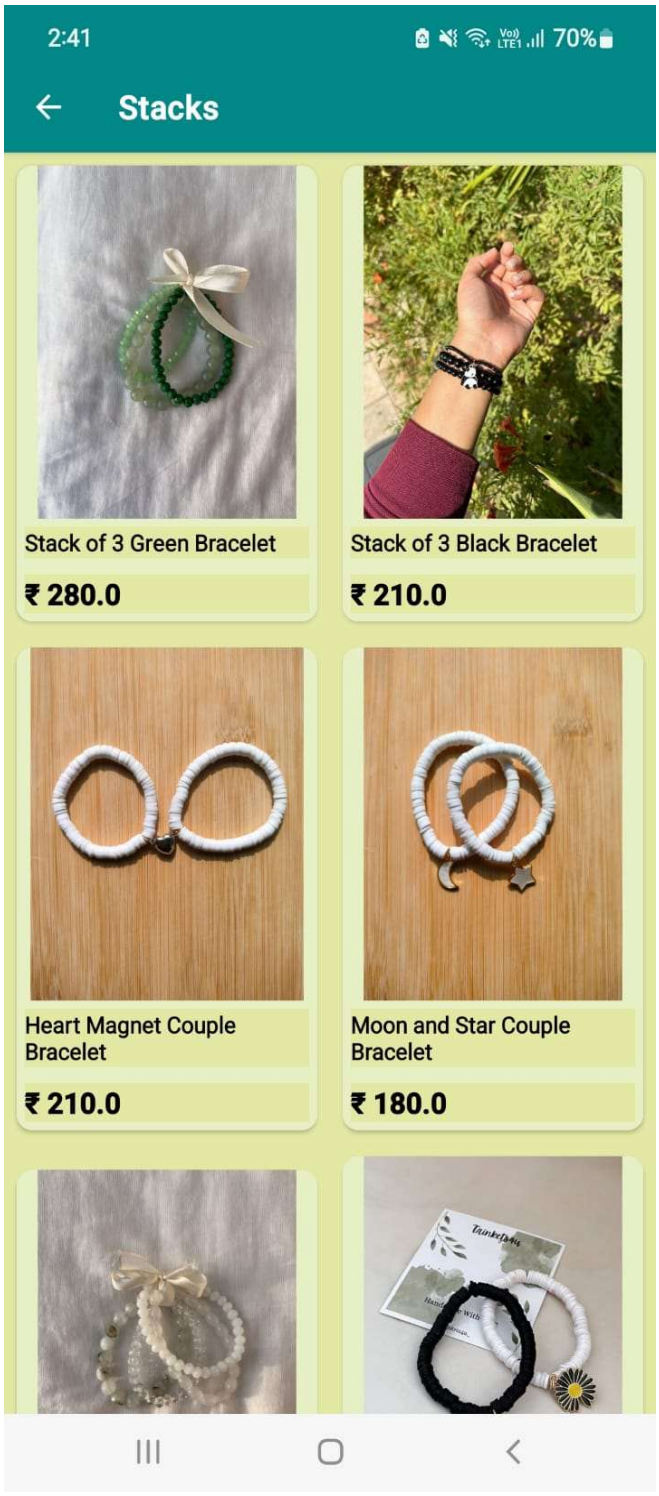


Figure 7.9 Search bar



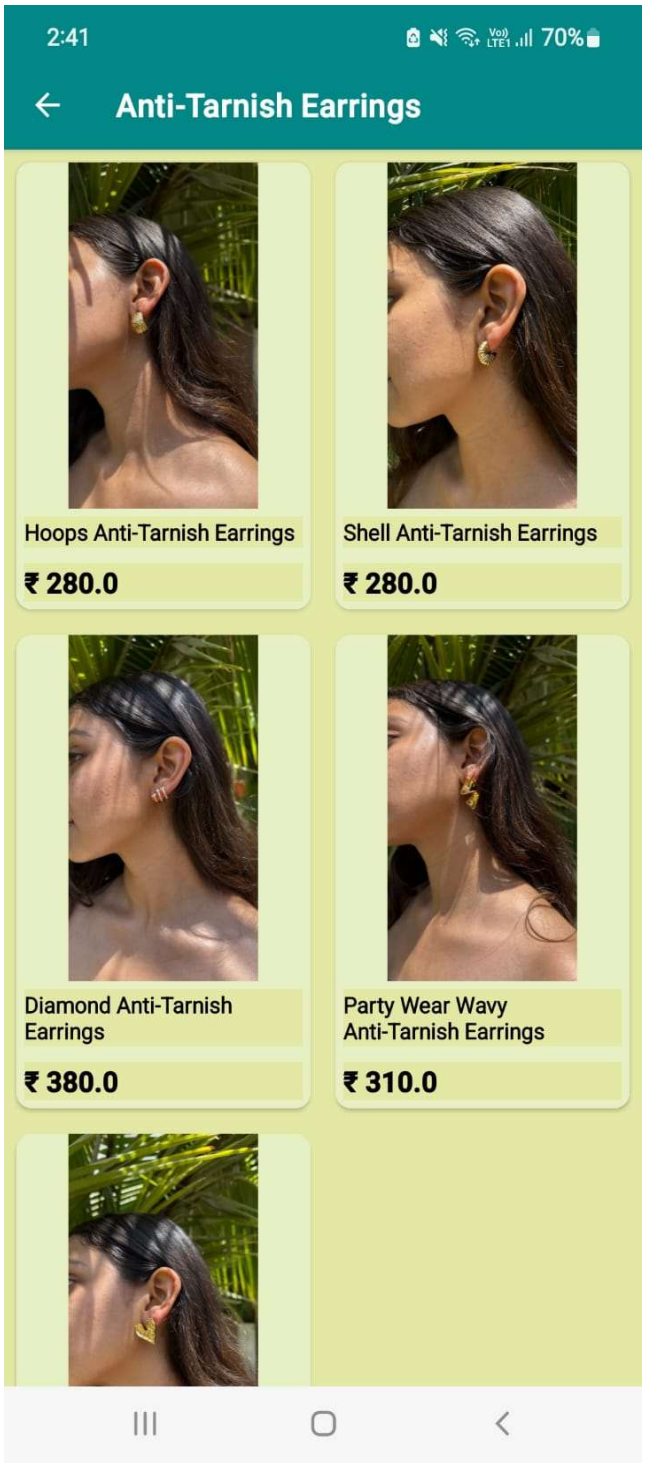
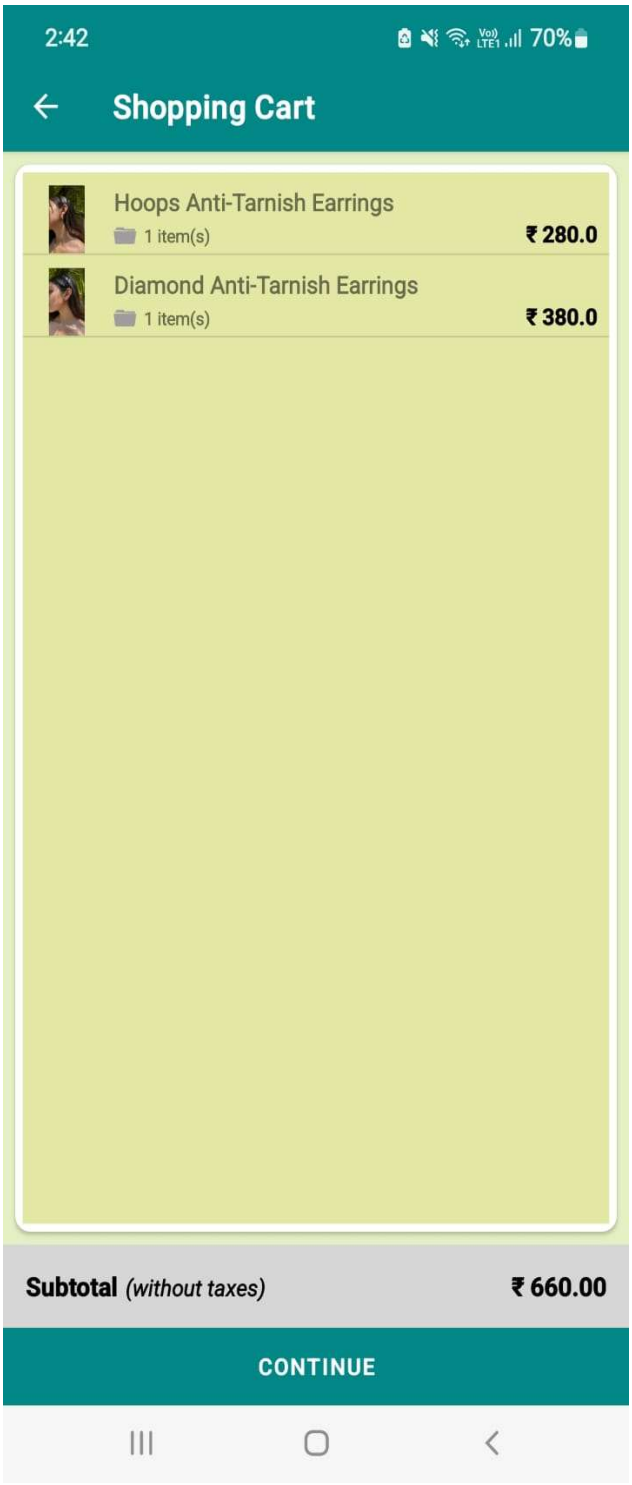


Figure 7.10 Product details



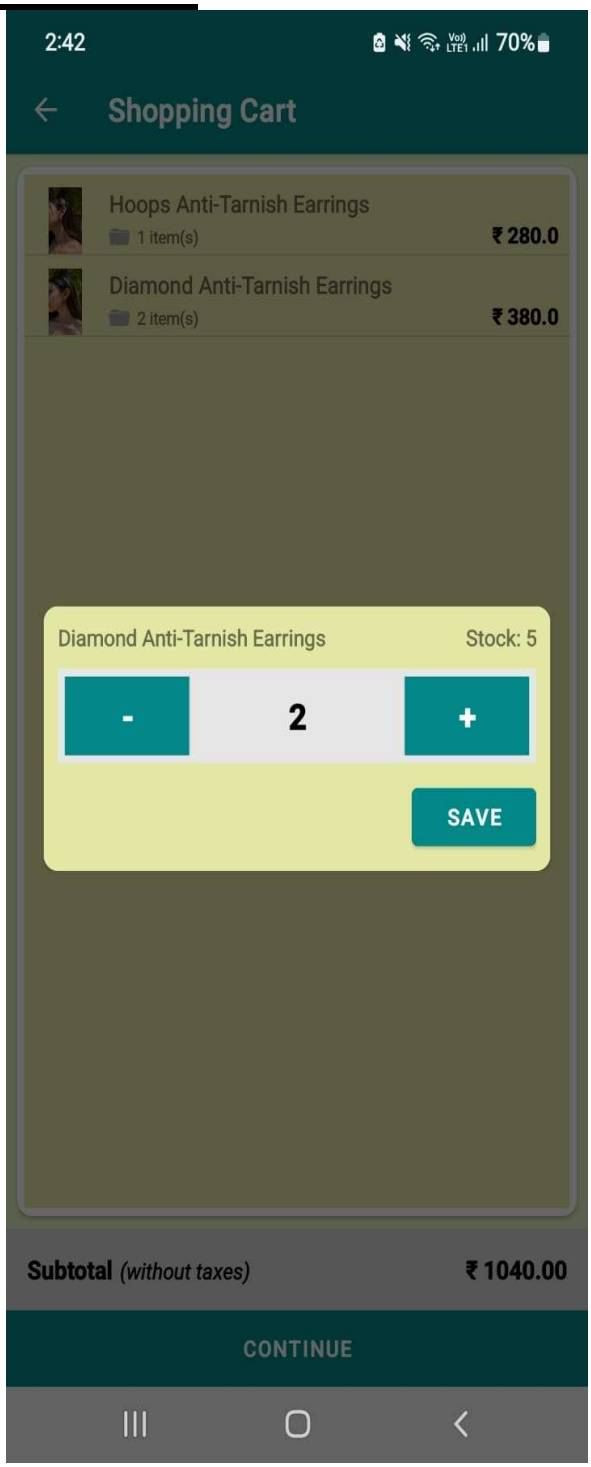
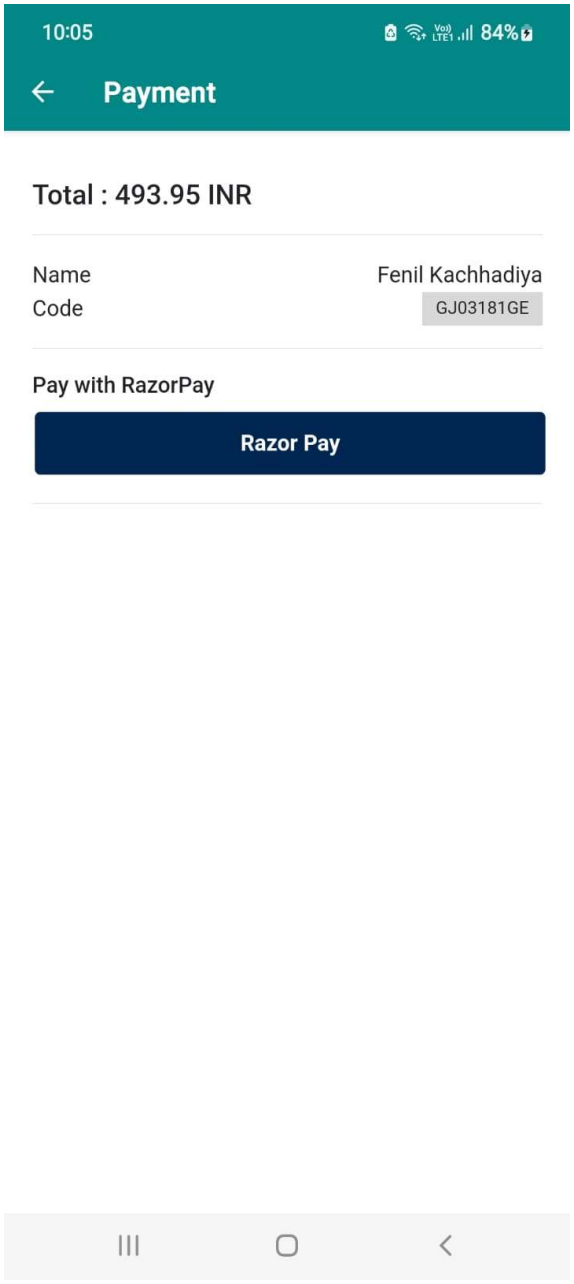


Figure 7.11 Checkout order and view order number



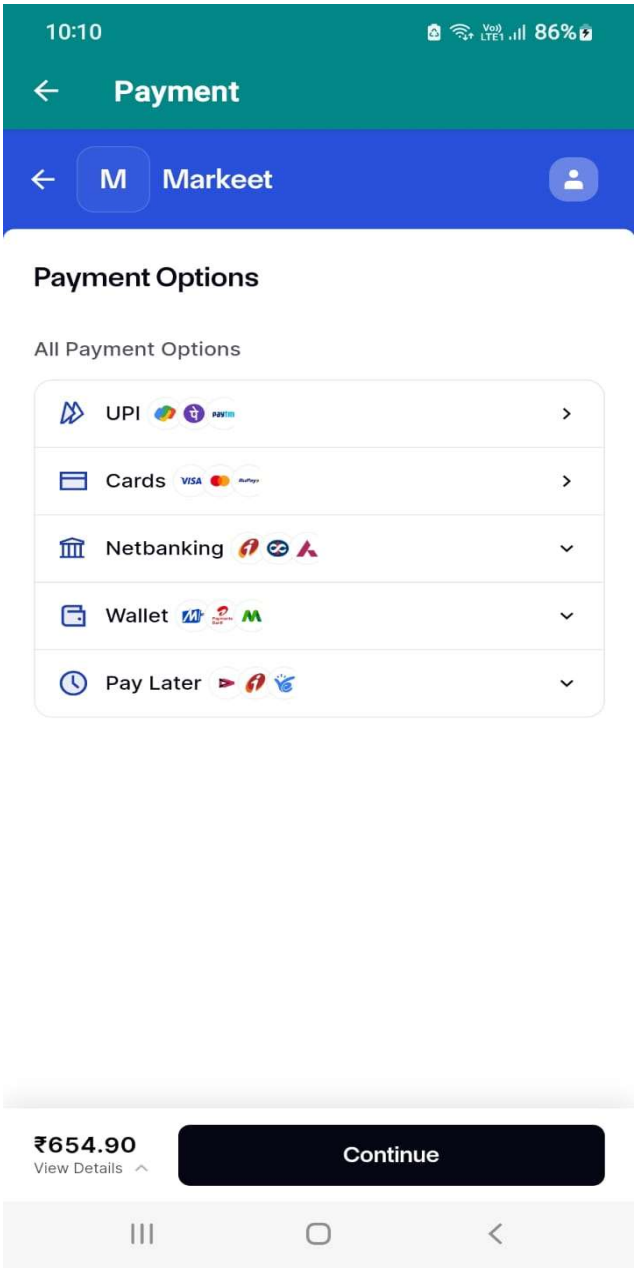
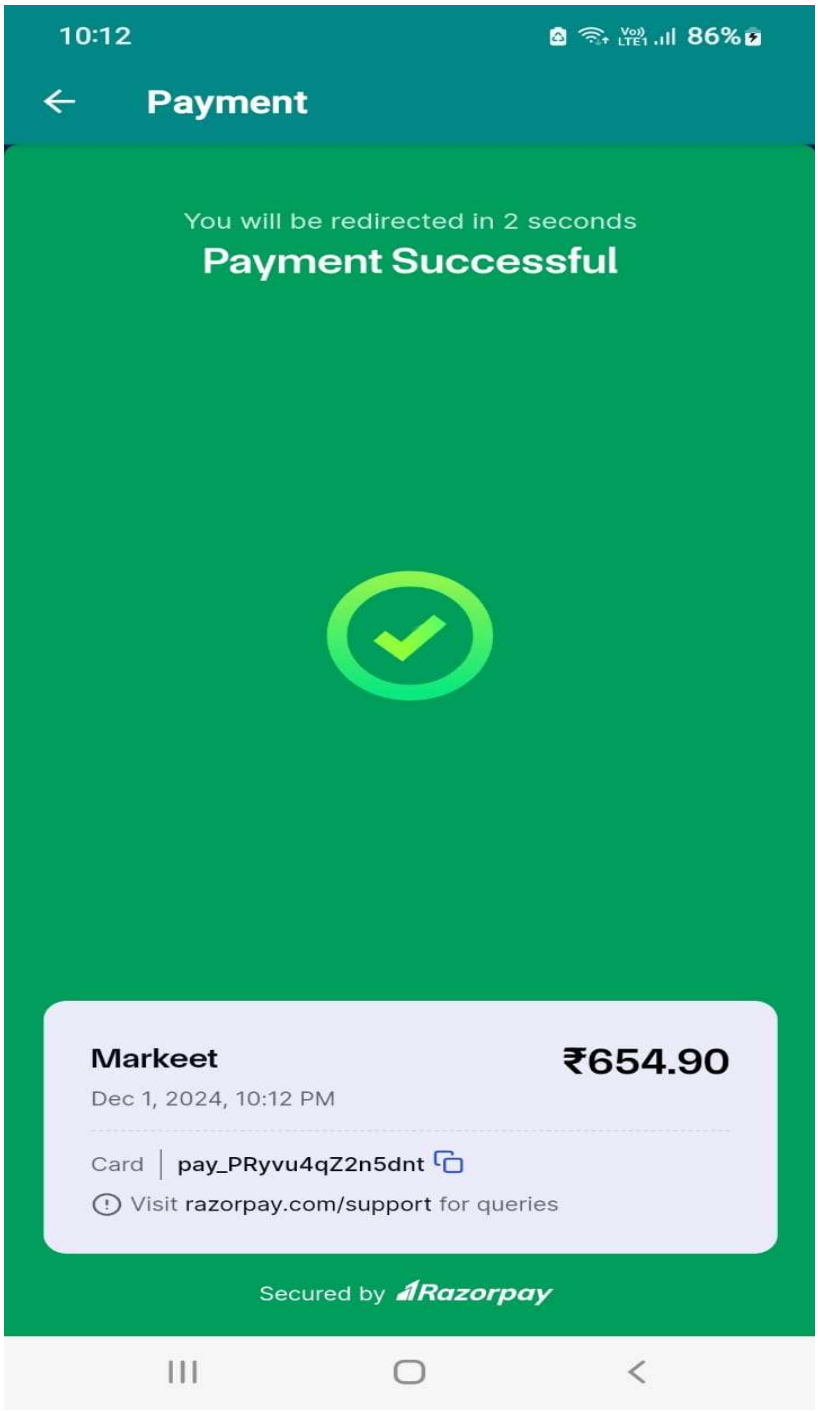


Figure 7.13 Payment Confirming Process



TESTING

Test cases

Testing is a critical phase of the software development lifecycle that ensures the e-commerce application meets its functional and non-functional requirements. A robust testing strategy was implemented to verify the system's reliability, performance, and user experience.

Types of Testing Performed

1. Unit Testing:
Each module (e.g., User Management, Product Management, Order Management) was tested independently to verify its correctness.
- Objective: Validate individual functions and methods.
- Example: Testing the login function to ensure correct credential validation.

2. Integration Testing:
Integration points between different modules were tested to ensure smooth communication and data flow.
- Objective: Verify that modules interact correctly.
- Example: Ensuring data from the Product Management module appears correctly in the shopping cart.

3. System Testing:
The entire application was tested as a complete system to verify that all requirements are met.
- Objective: Ensure end-to-end functionality of the application.
- Example: Simulating a full user journey from registration to payment and order confirmation.

4. Regression Testing:
After adding or modifying features, the system was retested to ensure new changes did not introduce defects.
- Objective: Maintain application stability.
- Example: Ensuring the new payment gateway did not disrupt order processing.

5. Performance Testing:
The system's performance under varying loads was evaluated.
- Objective: Ensure the application remains responsive and efficient.
- Example: Simulating high user traffic during a sale event.

6. Usability Testing:
The user interface was tested to ensure a seamless and intuitive user experience.
- Objective: Enhance user satisfaction.
- Example: Verifying the ease of navigating through product categories and placing orders.

Conclusion of Testing
Testing ensured the e-commerce application is functional, reliable, and user-friendly. Identified issues were resolved, and the application is now prepared for deployment with confidence in its performance and quality.

Actions:

A) User Registration and Authentication:

1. User Registration with valid inputs

- **Input:** Fill out registration form with valid details.
- **Expected Result:** User successfully registered and redirected to the dashboard.

2. User login with incorrect password

- **Input:** Enter valid email and incorrect password.
- **Expected Result:** Display error message: "Invalid credentials."

3. Forgot password functionality

- **Input:** Request a password reset email.
- **Expected Result:** Receive password reset link via email.

B) Product Search and Filter:

1. Search for a product by keyword

- **Input:** Enter "watch" in the search bar.
- **Expected Result:** Display all matching products with "watch" in the name or description.

2. Apply price filter

- **Input:** Set price range from ₹50 to ₹200.
- **Expected Result:** Display only products within the selected price range.

3. Sort by popularity

- **Input:** Select "Popularity" in sort options.
- **Expected Result:** Display products sorted by popularity.

C) Shopping Cart:

1. **Add product to cart**
   - **Input:** Click "Add to Cart" for a product.
   - **Expected Result:** Product added to the cart; cart count updated.

2. **Remove product from cart**
   - **Input:** Click "Remove" for a product in the cart.
   - **Expected Result:** Product removed; cart total updated.

3. **Update product quantity in cart**
   - **Input:** Change quantity to 3 for a product.
   - **Expected Result:** Total price updated based on new quantity.

D) Payment and Checkout:

1. **Successful payment with credit card**
   - **Input:** Complete checkout using valid credit card details.
   - **Expected Result:** Payment successful; order confirmation page displayed.

2. **Payment gateway timeout**
   - **Input:** Attempt payment but simulate a gateway timeout.
   - **Expected Result:** Display error message: "Payment failed. Please try again."

3. **Apply discount code**
   - **Input:** Enter a valid promo code during checkout.
   - **Expected Result:** Discount applied; total price updated.

E) Order Management:

1. **View order history**
   - **Input:** Navigate to "My Orders" in user dashboard.
   - **Expected Result:** Display list of past orders.

2. **Cancel an order**
   - **Input:** Click "Cancel" for a pending order.

- **Expected Result:** Order status updated to “Cancelled.”

3. Track an order
- **Input:** Click “Track” for a shipped order.
- **Expected Result:** Display order tracking details.

F) Admin Panel:

1. Add a new product
- **Input:** Fill out and submit product details form.
- **Expected Result:** Product successfully added to the catalog.

2. Update inventory
- **Input:** Modify stock levels for a product.
- **Expected Result:** Inventory levels updated.

3. Generate sales report
- **Input:** Select date range and generate report.
- **Expected Result:** Sales report generated and displayed.

G) Performance Testing:

1. Test under normal load
- **Input:** Simulate 100 users browsing simultaneously.
- **Expected Result:** Website performs without delays or errors.

2. Test under peak load
- **Input:** Simulate 1,000 users browsing simultaneously.
- **Expected Result:** Website remains functional with minimal performance degradation.

H) Security Testing:

1. Unauthorized access attempt
- **Input:** Access protected resources without logging in.
- **Expected Result:** Displayed with an error message.

2. Session timeout
- **Input:** User remains inactive for a defined period.
- **Expected Result:** User session expires, and they are logged out.

8. Conclusion & Future Work

The e-commerce application is a user-friendly platform designed for online jewellery shopping. It provides essential features like product catalog, product browsing and searching, secure payments, user registration and login, shopping cart management, and efficient admin management. The project successfully meets its goals, offering a reliable and scalable solution for users.

In future, we can plan to integrate features like

- **Customization Features:** Allow users to design personalized jewelry by selecting materials, beads, charms, etc.
- **Order Tracking:** Allow users to track order statuses like "Processing," "Shipped," and “Delivered”.
- **Enhanced Security**: Add two-factor authentication and stronger encryption.
- **International Expansion**: Support multiple languages and currencies.
- **Loyalty Programs**: Reward repeat customers with points or discounts.

9. BIBLOGRAPHY

- www.giva.com
- www.tanishq.com
- www.blueston.com