AURA JEWELLERY

A Project Report

Submitted in partial fulfilment of the
Requirements for the award of the Degree of

BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)

By

ADINA AZIZ AHMED SHAIKH

Seat Number:

Under the esteemed guidance of

Mr. Harun Latif Khan

Assistant Professor & Co-Ordinator of B.SC-IT

DEPARTMENT OF INFORMATION TECHNOLOGY

ORIENTAL EDUCATION SOCIETY'S COLLEGE OF ARTS, COMMERCE & SCIENCE

(Affiliated to University of Mumbai)

MUMBAI, 400 102 MAHARASHTRA

2025-2026

PROFORMA FOR THE APPROVAL PROJECT PROPOSAL

(Note: All entries of the proforma of approval should be filled up with appropriate and complete information. Incomplete proforma of approval in any respect will be summarily rejected.)

PNR No: _____ Roll No: _____

1. Name of the Student

_____

2. Title of the Project

_____

3. Name of the Guide

_____

4. Teaching experience of the Guide _____

5. Is this your first submission? Yes No

Signature of the Student Signature of the Guide

Date: _____ Date: _____

Signature of the Coordinator

Date: _____

ORIENTAL EDUCATION SOCIETY'S COLLEGE OF ARTS, COMMERCE & SCIENCE

(Affiliated to University of Mumbai)

MUMBAI -MAHARASHTRA– 400102

CERTIFICATE

This is to certify that the project entitled, **"AURA JEWELLERY''**, is Bonafide work of **ADINA AZIZ AHMED SHAIKH** bearing Seat. Number:( ) submitted in partial fulfillment of the requirements for the award of degree of BACHELOR OF SCIENCE in INFORMATION TECHNOLOGY from University of Mumbai.

Signature of the principal

Internal Guide **Coordinator**

External Examiner

Date: **College Seal**

ABSTRACT

The project *Aura Jewellery E-Commerce Website* aims to create a premium online jewellery platform that blends elegance with customer-focused services. Beyond standard e-commerce features such as product catalog, cart, payment gateway, and order tracking, the system introduces **customizable gift hampers**, **returns and** repair management, and exclusive post-purchase offers to enhance trust and satisfaction. An **AI-powered** Chabot further enriches the shopping experience by answering queries, guiding purchases, and providing real-time order assistance.

Technically, the platform is developed using C# with Visual Studio for both front-end and back-end integration, with MySQL serving as the database. The development process follows the Waterfall model, ensuring systematic execution from requirement gathering to deployment. Security, reliability, and performance are prioritized through encryption, structured workflows, and both manual and automated testing.

In essence, Aura Jewellery is envisioned as a luxurious yet reliable e-commerce solution, combining robust technical design with innovative features that distinguish it from conventional jewellery websites. By offering personalization, secure transactions, and strong after-sales support, it aims to redefine the standards of online jewellery shopping.

ACKNOWLEDGEMENT

To list who all have helped me in a difficult time because they are so numerous and the depth is so enormous. I would like to acknowledge the following as being idealistic channels and fresh dimensions in the completion of this project. I take this opportunity to thank the **University of Mumbai** for giving me the chance to do this project.

I would like to thank Principal **Dr. Abdul Aziz Shaikh** for providing the necessary facilities required for completion of this project. I take this opportunity to thank our Coordinator **Asst. Prof. Harun Latif Khan,** for his moral support and guidance.

I would also like to express my sincere gratitude towards my project guide **Asst. Prof. Harun Latif Khan,** whose guidance and care made the project successful. I would like to thank my **College Library**, for having provided various reference books and magazines related to my project.

Lastly, I would like to thank each and every person who directly and indirectly helped me in the completion of the project, especially **my Parents and Peers** who supported me throughout my project.

DECLARATION

I hereby declare that the project entitled, "**AURA JEWELLERY**" done at **ORIENTAL EDUCATION SOCIETY'S COLLEGE OF ARTS, COMMERCE & SCIENCE**, has not been in any case duplicated to submit to any other university for the award of any degree.

To the best of my knowledge other than me, no one has submitted to any other university.

The project is done in partial fulfillment of the requirements for the award

Of degree of **BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)** to be submitted as a final semester project as part of our curriculum.

Name and Signature of the student

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

The jewellery industry has witnessed a significant shift from traditional retail to online platforms, driven by the growing demand for convenience, personalization, and secure digital transactions. Customers today not only expect access to a wide variety of jewellery collections but also seek additional services that build trust and enhance their overall shopping experience. In this context, e-commerce platforms must go beyond basic product listings and payment gateways to deliver differentiated, customer-centric solutions.

The project **Aura Jewellery E-Commerce Website** is designed to address these needs by creating a luxury-focused online platform that combines premium jewellery offerings with innovative features. Unlike conventional jewellery websites, Aura introduces customizable gift hampers, returns and repair management, and post-purchase offers to ensure reliability and satisfaction at every stage of the customer journey. Moreover, an integrated AI chatbot provides instant assistance for queries, order status updates, and product recommendations, improving both engagement and accessibility.

On the technical side, the platform leverages C# with Visual Studio for front-end and back-end integration, while MySQL manages the database. The development process follows the Waterfall model, ensuring a systematic approach to requirement analysis, design, implementation, testing, and deployment. Security measures, including encrypted transactions and data protection, are embedded to safeguard customer information. Overall, **Aura Jewellery** aims to set a new benchmark in online jewellery shopping by offering an elegant, secure, and customer-first digital experience.

1.1 BACKGROUND

In recent years, the global shift towards digital commerce has transformed the way people shop for luxury goods, including jewellery. Traditional jewellery retail is often limited by physical presence, high operational costs, and restricted accessibility. Customers now prefer online platforms that provide convenience, variety, and security, along with added services that replicate or even enhance the in-store shopping experience. This shift has created opportunities for specialized e-commerce platforms that focus on both luxury and customer satisfaction.

Despite the growth of jewellery e-commerce, many existing platforms lack features such as customization of gift hampers, streamlined returns and repair processes, and personalized post-purchase offers. Customers are often hesitant to buy expensive jewellery online due to concerns about trust, product authenticity, after-sales service, and ease of communication with sellers. Addressing these challenges requires an innovative platform that not only facilitates purchases but also ensures strong after-sales support, personalization, and reliability.

The Aura Jewellery E-Commerce Website is conceptualized to fill this gap by integrating premium jewellery offerings with customer-centric services. By combining traditional e-commerce features with advanced modules such as hamper customization, AI-powered Chabot assistance, and defect/repair management, the project aims to provide a luxurious yet dependable digital experience. With a focus on usability, security, and innovation, Aura Jewellery aspires to redefine online jewellery shopping and establish trust in the luxury e-commerce domain.

1.2 OBJECTIVES

The primary objective of the Aura Jewellery E-Commerce Website is to design and develop a premium, **secure,** and customer-focused online platform for jewellery shopping. To achieve this, the project is guided by the following specific objectives:

1. Develop a user-friendly e-commerce platform that allows customers to browse, search, and purchase jewellery products with ease.
2. Introduce customizable gift hampers where users can select multiple items, add personalized messages, and create unique gift packages.
3. Implement a secure payment gateway and ensure encrypted transactions to build customer trust and safeguard sensitive information.
4. Enable returns and repair management workflows for defective or damaged products, ensuring reliable after-sales support.
5. Provide personalized post-purchase offers and festival discounts to enhance customer loyalty and engagement.
6. Integrate an AI-powered Chabot for instant query resolution, product recommendations, and real-time order status updates.
7. Design a robust admin panel for managing inventory, customer data, orders, returns, and promotional offers efficiently.
8. Ensure system reliability, performance, and security through proper testing (manual and automated) and encryption techniques.
9. Deliver an elegant and luxury-themed user interface (UI) that reflects the brand identity while maintaining accessibility and ease of navigation.

1.3PURPOSE, SCOPE, AND APPLICABILITY

1.3.1 PURPOSE

The purpose of the Aura Jewellery E-Commerce Website is to create a premium, reliable, and customer-**centric** online jewellery shopping platform that overcomes the limitations of traditional retail and existing e-commerce solutions. While most jewellery websites focus only on product listings and basic transactions, customers today expect personalization, trust, and strong after-sales support when purchasing luxury items online.

This project is designed to fulfill those expectations by offering customizable gift hampers, returns and repair management, and exclusive post-purchase offers, thereby enhancing customer satisfaction and loyalty. Additionally, features such as a secure payment system and an AI-powered chatbot ensure convenience, transparency, and instant assistance throughout the shopping journey.

In essence, the purpose of this project is not only to facilitate jewellery purchases online but also to build long-term trust by providing a luxurious, secure, and interactive shopping experience. It seeks to redefine online jewellery retail by combining technology, elegance, and customer care into one complete solution.

1.3.2 SCOPE

The Aura Jewellery E-Commerce Website is designed to provide a comprehensive and luxury-focused online shopping experience for jewellery customers. Its scope extends beyond basic e-commerce functionality, covering the entire journey from browsing products to after-sales services.

The system includes essential modules such as product catalog management, shopping cart, secure checkout, and user account management. It also introduces advanced features like **cu**stomizable gift hampers, where customers can combine multiple products with personalized messages, and a returns and repair management system to handle defective or damaged items. Additionally, festival discounts, post-purchase offers, and promotional campaigns are integrated to enhance customer engagement and loyalty.

The platform also features an AI-powered Chabot that provides real-time assistance, answers product queries, gives order updates, and offers personalized recommendations. On the administrative side, the project includes an admin panel for managing inventory, customers, orders, offers, and reports. Technical scope covers development using C# with Visual Studio, MySQL database, and thorough manual and automated testing to ensure performance, reliability, and security.

Overall, the project scope spans front-end design, back-end processing, secure payments, user experience, and post-purchase support, ensuring that Aura Jewellery emerges as a complete, innovative, and trustworthy e-commerce platform for online jewellery retail.

1.3.3 APPLICABILITY

The Aura Jewellery E-Commerce Website can be applied in multiple real-world contexts within the jewellery and luxury retail industry. Its applicability extends to both customers and business stakeholders.

1. Jewellery Retailers and Brands – The platform provides jewellery businesses with an online marketplace to showcase their products, manage inventory, and expand their reach beyond physical stores.
2. Customers – It serves as a convenient and secure platform for customers to browse, customize hampers, purchase jewellery, and access after-sales services such as returns and repairs.
3. Luxury Gifting Market – With its customizable gift hampers and personalized offers, the system can be applied to festive seasons, weddings, anniversaries, and corporate gifting.
4. Customer Support Enhancement – The integrated AI Chabot ensures 24/7 assistance, making the platform applicable for improving customer service and engagement.
5. E-Commerce Expansion – Beyond jewellery, the project's framework can be adapted for other luxury goods, offering scalability for broader e-commerce applications.

In essence, the project is applicable to modern jewellery businesses aiming to establish a digital presence, customers seeking secure and personalized shopping experiences, and the luxury retail sector as a whole by combining technology, trust, and elegance.

1.4 ACHIEVEMENTS

The Aura Jewellery E-Commerce Website successfully meets its objectives by delivering a premium and customer-focused online shopping platform. One of the major achievements of the project is the integration of customizable gift hampers, which allows customers to create personalized jewellery packages for special occasions. This feature enhances the gifting experience and differentiates the platform from conventional jewellery websites.

Another key achievement is the implementation of returns and repair management workflows, ensuring customer trust and satisfaction even after purchase. The addition of post-purchase offers and festival discounts further strengthens customer engagement and loyalty. The integration of an AI-powered chatbot stands out as an innovative achievement, enabling real-time assistance for product queries, order tracking, and recommendations, thereby improving overall user interaction.

On the technical side, the project achieves a secure, reliable, and scalable system developed using C# with Visual Studio and MySQL database. Both manual and automated testing ensured functionality, performance, and security across all modules. The combination of luxury-themed UI design, robust back-end processing, and strong after-sales support reflects the project's achievement of its goal—delivering an elegant, trustworthy, and innovative jewellery e-commerce solution.

CHAPTER 2

SURVEY OF TECHNOLOGIES

1.DATABASE:

Oracle is a multinational technology company best known for its database software, cloud solutions, and enterprise applications. It provides tools for data management, analytics, and business operations, helping organizations run efficiently. Oracle is also a leader in cloud infrastructure and AI-driven enterprise solutions.

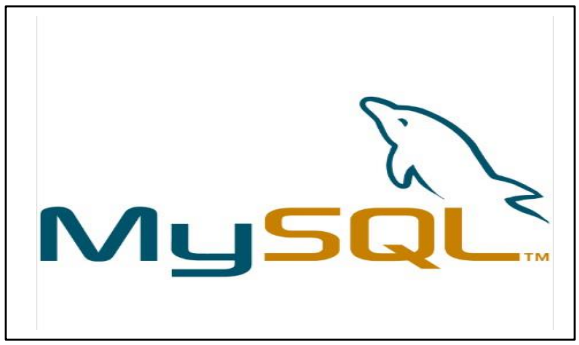
2.FRONTEND:

Visual Studio is a powerful Integrated Development Environment (IDE) by Microsoft, used for developing applications in languages like C#, C++, and Python. It provides features such as code editing, debugging, and database connectivity, making it suitable for building secure and efficient web applications like the Aura Jewellery E-Commerce Website.

VISUAL STUDIO:

3.DATABASE:

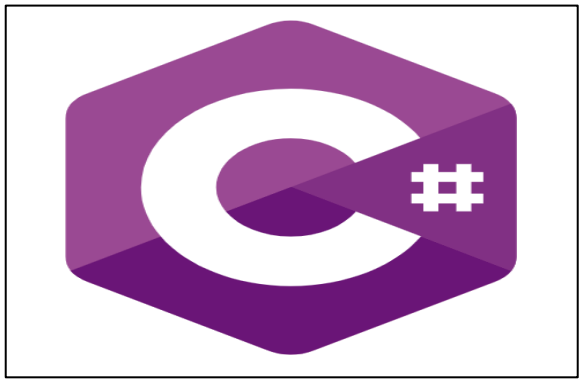
MYSQL:



MySQL is an open-source **relational database management system (RDBMS)** used to store and manage data. It is fast, reliable, and widely used in **web applications** due to its ease of use and support for large-scale databases. MySQL works with many programming languages and is highly preferred for **dynamic websites and applications**.

4. BACKEND:

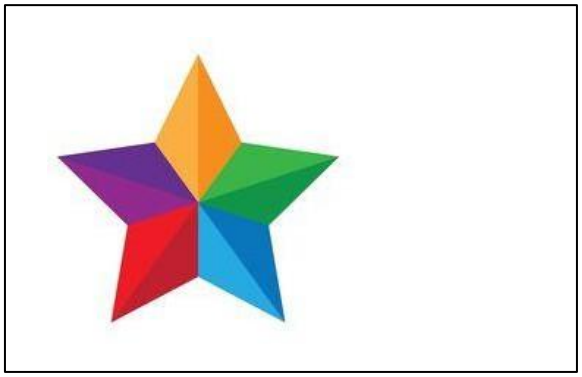
C#:



C# is a modern, object-oriented programming language developed by Microsoft. It is widely used for building desktop, web, and mobile applications on the .NET framework. C# is known for its simplicity, efficiency, and strong security features

5. DESIGN&MODELING:

STARUML:



StarUML is a powerful software modeling tool that supports UML (Unified Modeling Language) for designing and visualizing system architecture. It helps developers create diagrams like class, use-case, and sequence diagrams to understand and document software requirements. StarUML makes complex system design easier and more structured.

6.BACKEND**:**

JAVA:

6.**Java** is a versatile, object-oriented programming language developed by Sun Microsystems (now owned by Oracle). It follows the principle of “write once, run anywhere”, making applications platform-independent. Java is widely used for web, mobile, and enterprise applications due to its portability, security, and robust performance. It also provides a rich set of libraries, frameworks, and tools that support efficient software development.

AI INTEGRATION:

2.3 Deep Learning Frameworks

Suitable for advanced classification tasks involving complex language patterns and document structures:

1. Convolutional Neural Networks (CNNs) – Useful for image-based document analysis and layout detection.
2. Recurrent Neural Networks (RNNs) – Ideal for sequence-based tasks such as document flow modelling.
3. Long Short-Term Memory (LSTM) – A type of RNN designed to learn long-term dependencies in text.
4. Transformers – State-of-the-art models (like BERT and GPT) for language understanding and classification.

2.4 Document Processing Libraries

Used to extract text and metadata from documents:

1. Apache Tika – Detects and extracts metadata and text from various file types.
2. PyPDF2 – Reads and manipulates PDF files using Python.
3. OpenCV – Handles image-based document processing (e.g., scanned documents).
4. PDFMiner – Specialized in extracting text from PDFs with layout preservation.

CHAPTER 3

REQUIREMENTS AND ANALYSIS

3.1 PROBLEM DEFINITION

The jewellery industry is rapidly moving **toward digital platforms, but most existing e-commerce websites focus only on** basic product listings, shopping carts, and payments**. They lack advanced features** that build trust, personalization, and long-term customer satisfaction, which are crucial for luxury items like jewellery. Customers face several challenges such as:

- Lack of **customizable gifting options** (e.g., combining multiple jewellery items into hampers).
- Limited or unclear **returns and repair processes**, making customers hesitant to purchase expensive jewellery online.
- Weak **after-sales support**, which reduces customer trust in digital jewellery shopping.
- Absence of **AI-driven assistance**, leading to poor customer engagement and delayed query resolution.
- Security concerns regarding **online payments and data privacy**.

These problems highlight the need for a platform that not only enables secure and convenient online jewellery shopping but also provides **personalization, transparency, and strong after-sales services**.

The **Aura Jewellery E-Commerce Website** is proposed as a solution to overcome these limitations by integrating **gift hamper customization, secure returns/repairs, post-purchase offers, AI chatbot support, and robust security features**, creating a reliable and customer-friendly luxury shopping experience.

3.2 REQUIREMENTS SPECIFICATION

1. Functional Requirement

- **User Requirements:**
  - Register, log in, and manage personal accounts securely.
  - Browse jewellery catalog with categories, filters, and search options.
  - Add products to cart, proceed with checkout, and make secure payments.
  - Create custom gift hampers with multiple jewellery items and personalized messages.
  - Track placed orders and receive notifications (email/SMS).
  - Request returns, refunds, or repair services for damaged/defective jewellery.
  - Access post-purchase offers and festival discounts.
  - Interact with an AI-powered chatbot **for order updates, FAQs, and product recommendations.**
- **Admin Requirements** :
  - Add, update, and delete jewellery products in the catalog.
  - Manage customer accounts, orders, and payment records.
  - Approve and process return/repair requests.
  - Create and manage promotional campaigns, discounts, and offers.
  - Generate reports on sales, products, and customer activity.
  - **System Requirements:**
  - Secure **payment gateway integration** with data encryption.
  - Maintain **database** for products, orders, customers, and transactions.
  - Send automated notifications for order status, returns, and promotions.
  - Provide search optimization for faster product retrieval.

2. **Non-Functional Requirements:**
   - **Security** – All sensitive data (user details, payments) must be encrypted.
   - **Performance** – Quick response time and ability to handle multiple users simultaneously.
   - **Usability** – Luxury-themed, intuitive, and accessible interface for all users.
   - **Reliability** – Accurate order processing and error-free transactions.
   - **Scalability** – Ability to expand product catalog, users, and new features.
   - **Availability** – 24/7 platform accessibility with minimal downtime.
   - **Maintainability** – Modular design to simplify updates and upgrades

Data Requirements

User Data

- User ID, username, encrypted password.
- Name, email, phone number.
- Billing and shipping addresses.
- Saved payment details and order history.

Product Data

- Product ID, name, description, category.
- Material type (gold, silver, diamond, etc.).
- Price, discounts, stock availability.
- Product images and warranty information.

Gift Hamper Data

- Hamper ID, selected product IDs.
- Custom greeting/message text.
- Packaging options and total hamper price.

Order & Transaction Data

- Order ID, customer ID, product/hamper details.
- Quantity, total amount, tax details.
- Payment method and transaction status.
- Delivery details (courier, tracking number, status).

Return & Repair Data

- Request ID linked to Order ID.
- Reason for return/repair.
- Status (pending, approved, completed).
- Refund or replacement/repair details.

Admin Data

- Admin ID, login credentials, and role.
- Records of product management, offers, and reports.

Offer & Promotion Data

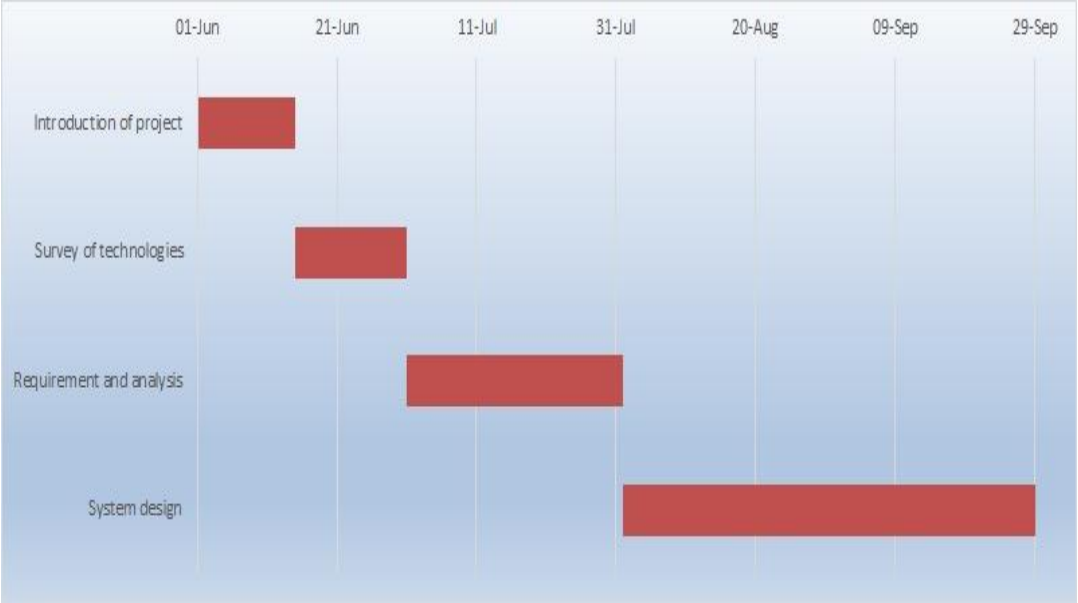
- Offer ID, type (festival, coupon, post-purchase).
- Validity period, usage limits, and applicable products.

Chatbot Data

- Session ID, user queries, responses.
- FAQs and escalation records.

3.3 PLANNING AND SCHEDULING

GANTT CHART



3.4 SOFTWARE AND HARDWARE REQUIREMENTS

1. Software Requirements

- **Operating System:** Windows 10 or higher (development), cross-platform support for
- **Programming Language:** C# (.NET framework).
- **IDE / Development Tool:** Microsoft Visual Studio.
- **Database:** MySQL (for storing products, customers, orders, payments).
- **Web Server:** IIS / Apache (for hosting).
- **Testing Tools:** Manual testing tools, Selenium (for automated testing).
- **Browser Compatibility:** Google Chrome, Mozilla Firefox, Microsoft Edge, Safari.
- **Security Tools:** SSL/TLS encryption, secure payment gateway integration.

2. Hardware Requirements

For Development Environment (Developers/Admin):

- Processor: Intel i5 or higher.
- RAM: Minimum 8 GB.
- Storage: 500 GB HDD / 256 GB SSD.
- Display: 15.6” monitor or higher with 1366x768 resolution.
- Internet: Stable broadband connection.

For Server (Deployment):

- Processor: Quad Core (Intel Xeon or equivalent).
- RAM: Minimum 16 GB.
- Storage: 1 TB HDD / SSD recommended.
- Network: High-speed internet with backup connection.
- Security: Firewall, SSL certificate, backup & recovery system.

For End Users (Customers):

- Device: Smartphone / Tablet / PC / Laptop.
- Browser: Chrome, Firefox, Edge, or Safari.
- Internet: Minimum 3G/4G or broadband connection.

3.5 PRELIMINARY PRODUCT DESCRIPTION

Aura Jewellery E-Commerce Website is designed as a premium online platform that provides a luxury shopping experience while ensuring trust, security, and convenience for customers. It aims to overcome the limitations of traditional jewellery websites by offering features such as customizable gift hampers, secure return and repair management, festival discounts, and AI-powered chatbot assistance.

From the customer's perspective, the system ensures a smooth and elegant shopping journey. Users can browse jewellery collections, filter items by category, price, or material, and view detailed product information with high-quality images. They can register, log in securely, and manage their profiles, orders, and addresses. The gift hamper customization option allows buyers to create unique gifting experiences with personalized messages and premium packaging. Customers also receive order tracking updates, real-time notifications, and reliable after-sales support through returns and repairs.

The administrator module provides an easy-to-use panel for managing jewellery products, customer accounts, orders, and promotional campaigns. Admins can approve or reject return/repair requests, track business performance through reports, and manage discounts and festival offers. This ensures smooth coordination between business operations and customer needs.

On the technical side, the platform is built using C# with Visual Studio for front-end and back-end development, while MySQL is used for data storage. The system ensures security through encrypted logins and payment gateways, and reliability through manual and automated testing. Designed with scalability in mind, it can handle a growing product catalog, increasing user base, and future features.

Key Features of the Product

User-Friendly Shopping – Browse jewellery, filter products, and view details with images and descriptions.

Gift Hamper Customization – Combine multiple jewellery items, add personalized messages, and choose packaging.

Secure Transactions – Multiple payment methods with SSL encryption and secure gateway integration.

Order Management – Place orders, track delivery, download invoices, and get real-time notifications.

Returns & Repairs – Easy workflow for refunds, replacements, and repair requests.

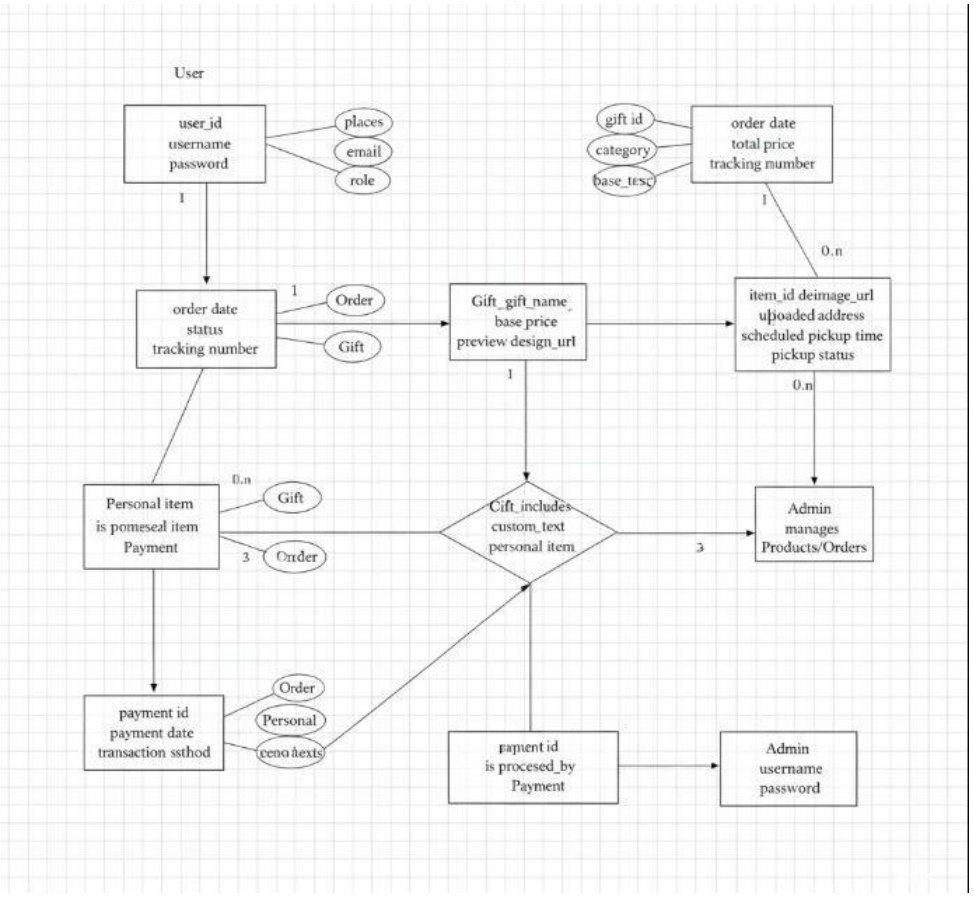
Post-Purchase Offers – Festival discounts, loyalty offers, and special promotions for repeat customers.

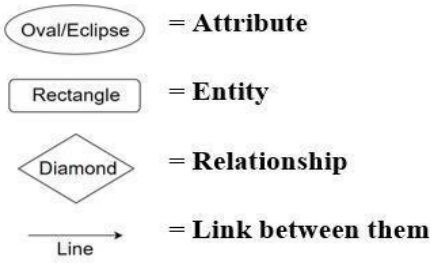
AI Chatbot – Provides 24/7 assistance for queries, product recommendations, and order updates.

Admin Panel – Manage products, customers, orders, offers, and generate sales reports.
CONCEPTUAL MODEL

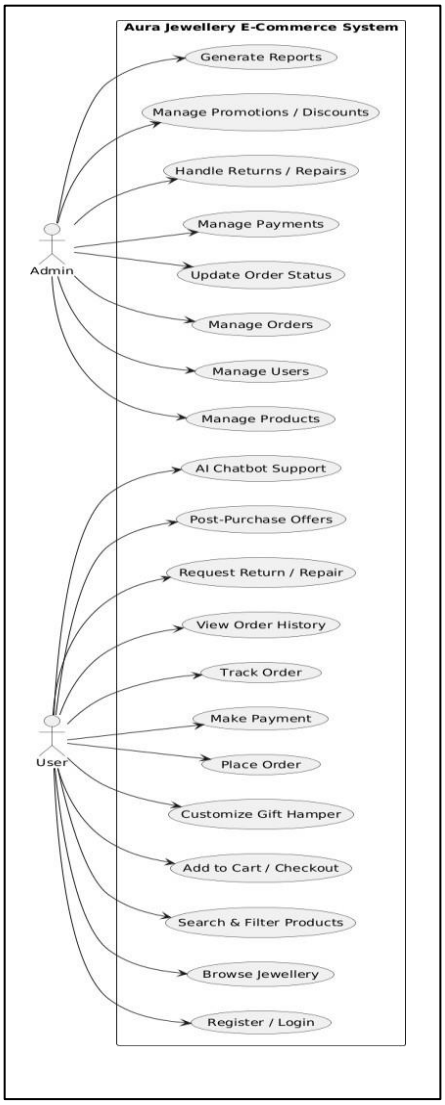
3.6 CONCEPTUAL MODEL

1. Entity Relationship Diagram





2. Use case Diagram

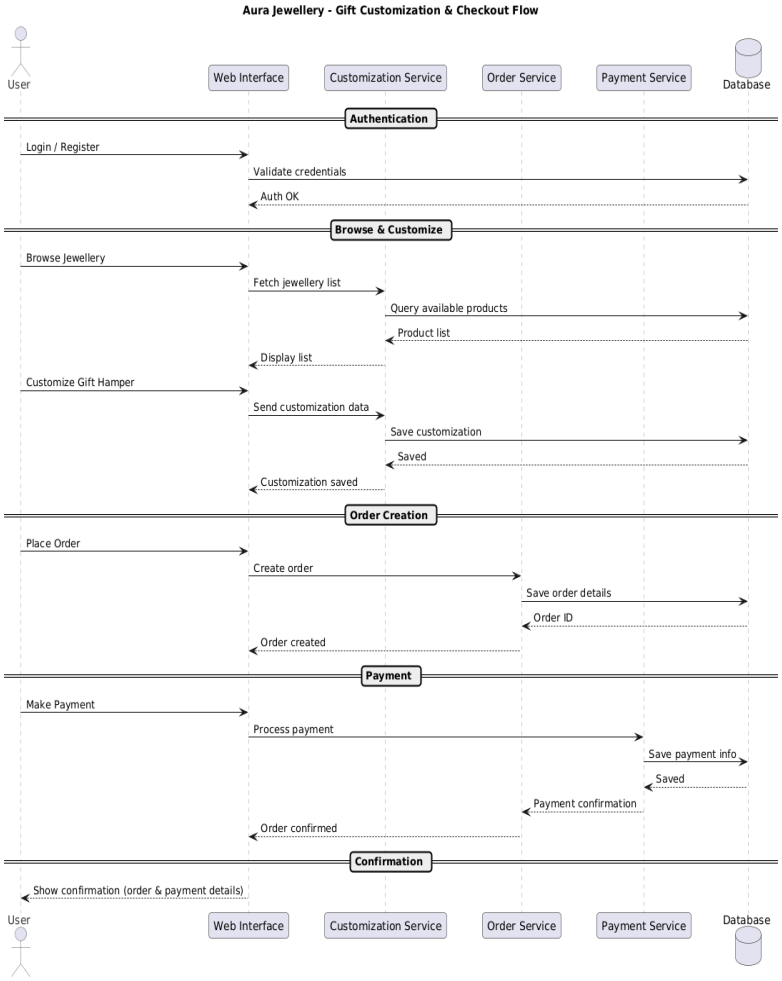


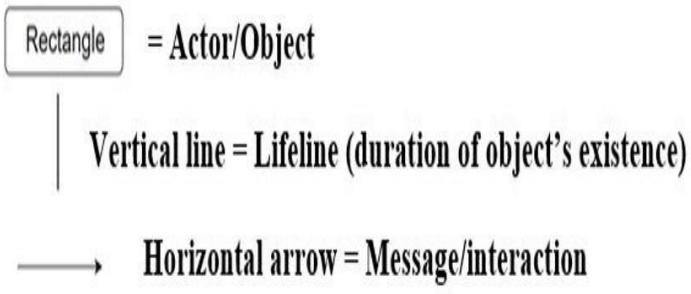
Stick Figure = Actor (user/system)

= Use Case (system functionality)

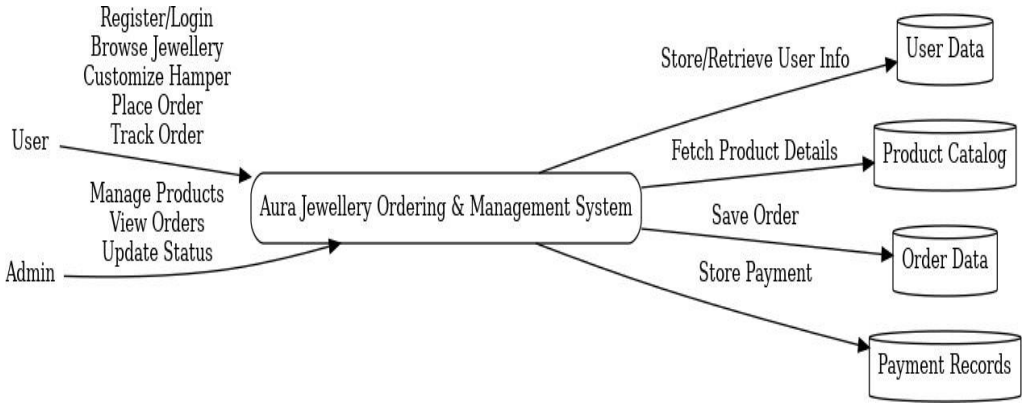
= Relationship (who performs what action)

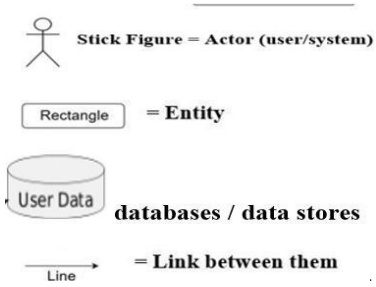
3. Sequence Diagram



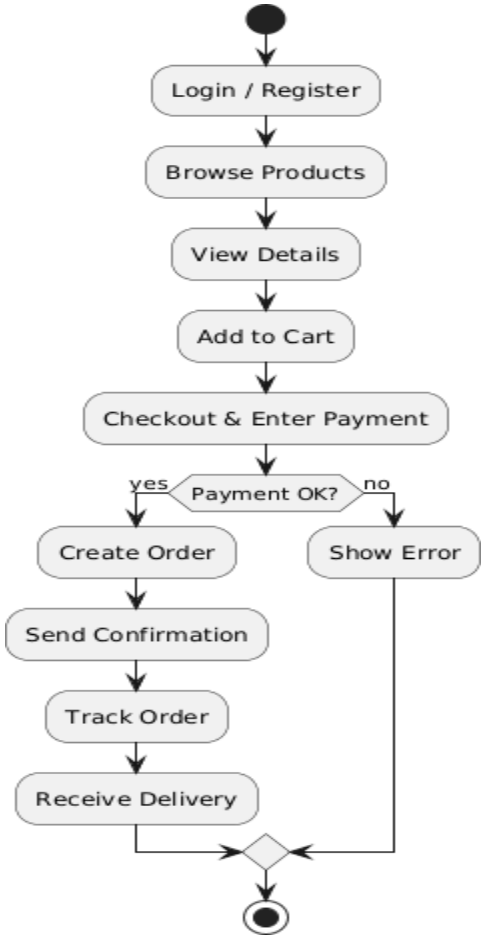


4. Data Flow Diagram





5. Activity diagram



= Start

Rectangle — Activity (task/action)

Flow of control

= End

CHAPTER 4

SYSTEM DESIGN

4.1 Basic Modules

User Module

- User registration and login
- Profile management (personal details, address, password reset)
- Browse and search jewellery catalog
- Add to cart, place orders, and make secure payments
- Track orders and view order history

Product & Catalogue Module

- Jewellery categories (rings, necklaces, bangles, etc.)
- Product details (images, price, description, stock)
- Search and filter products
- Update stock automatically after purchase

Gift Hamper Module

- Select multiple products to create a hamper
- Add greeting messages and choose packaging
- Final hamper price calculation and checkout

Order & Payment Module

- Manage cart and checkout
- Secure payment gateway integration
- Generate invoices and send order confirmations
- Real-time order tracking

Returns & Repairs Module

- Request return, refund, or repair

- Generate RMA (Return/Repair ticket)
- Track return/repair status

Offers & Promotions Module

- Festival discounts
- Post-purchase offers and loyalty rewards
- Coupon/discount code management

AI Chatbot Module

- Answer FAQs and product queries
- Provide order updates and recommendations
- 24/7 support with escalation to admin if needed

Admin Module

- Manage users, products, and categories
- View and process orders, returns, and repairs
- Create offers and promotions
- Generate reports on sales and customers

4.2 Data design

The data design of the Aura Jewellery E-Commerce Website defines how information will be structured, stored, and manipulated to support all system functionalities. It ensures that data remains consistent, secure, and easily retrievable for customers, administrators, and the system itself.

1. **Data Organization**

Data is divided into different entities (tables) based on the requirements of the jewellery e- commerce platform. Key entities include:

- User Data: Stores customer login credentials, personal details, contact numbers, and shipping/billing addresses.
- Product Data: Contains details such as product ID, name, category (ring, necklace, bangle, etc.), material, price, size, description, stock availability, and product images.
- Hamper Data: Stores customized gift hampers created by customers, including selected products, packaging details, messages, and hamper price.
- Order Data: Stores details of customer orders, including order ID, user ID, products/hamper IDs, payment details, order status, and delivery information.
- Returns and Repairs Data: Maintains records of defective or damaged product requests, repair tracking, refund details, and approval status.
- Offers & Promotions Data: Stores festival discounts, post-purchase offers, coupon codes, and their validity periods.
- Admin Data: Stores administrator credentials and access rights for managing products, users, orders, offers, and returns.
- Chatbot Data: Logs customer queries, chatbot responses, and escalation records for support.

2. Data Management

The system manages data using the following operations:

- Insertion: When a new customer registers, a new product or hamper is added, or a new order is placed.
- Updating: When product details, stock, order status, offers, or customer information are modified.
- Deletion: When a product is discontinued, an order is cancelled, or expired offers are removed.
- Retrieval: When customers browse or search products, track orders, or check return/repair status.

3. Data Manipulation

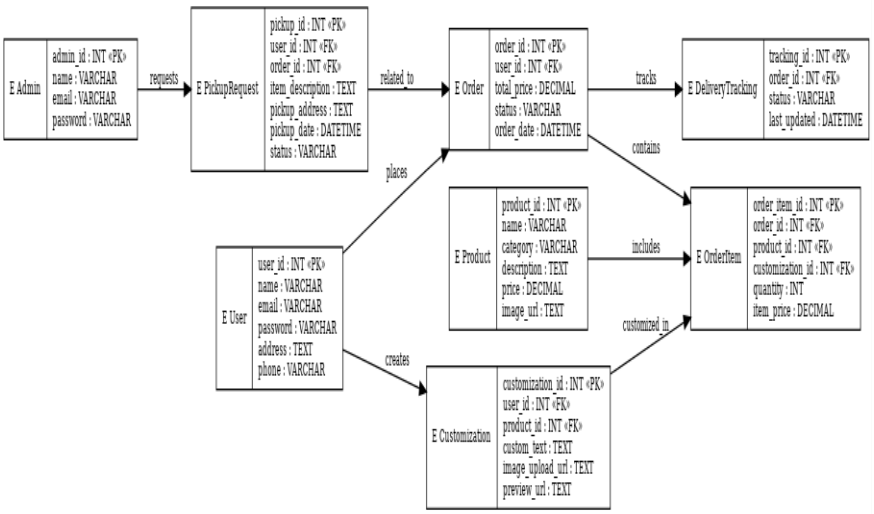
Data manipulation enables the system to deliver its functionalities. Examples include:

- Product Search & Filtering by price, category, material, or availability.
- User Authentication to verify login credentials against the User Database.
- Order Management where admins and customers can view, update, or track orders.
- Hamper Management where users create personalized hampers with selected jewellery items.
- Return & Repair Handling where requests are stored, reviewed, and updated until resolution.
- Offer Management where applicable discounts are applied to user orders.
- Chatbot Interaction where queries and responses are recorded for continuous support.

4. Database Design

- The website uses a Relational Database (MySQL/PostgreSQL) for structured and secure data storage.
- Each entity is mapped to a table with primary keys and foreign keys to maintain relationships.
- Example relationships:
  - A User can place multiple Orders (One-to-Many).
  - An Order can contain multiple Products or Hampers (Many-to-Many via Order Details).
  - A Hamper can include multiple Products (One-to-Many).
  - A Return/Repair Request is linked to one Order Item (One-to-One).
  - An Offer can apply to multiple Orders and users (Many-to-Many**).**

4.2.1 Schema Design



4.2.2 Data integrity and Constraints

4.2.2.1 Data Integrity

Data integrity ensures that all information in the system remains **accurate, consistent, secure, and reliable** throughout its lifecycle. Since this project deals with sensitive data such as customer profiles, jewellery product details, payments, and returns/repairs, maintaining strong data integrity is essential.

1. Entity Integrity

- Every table has a **primary key (PK)** that uniquely identifies each record.
- Examples:
  - user_id uniquely identifies each customer.
  - product_id uniquely identifies each jewellery item.
  - order_id uniquely identifies each order.
- This prevents duplicate or null entries in key fields.

2. Referential Integrity

- Relationships between tables are enforced using **foreign keys (FK)**.
- Examples:
  - Each order must be linked to a valid user_id in the User table.
  - Each hamper_item must reference valid product_id and hamper_id.
  - Each payment must be tied to a valid order_id.
- Cascading updates/deletes are used carefully to avoid orphaned records.

3. Domain Integrity

- Data fields accept only values within defined rules and constraints.
- Examples:
  - price and quantity must always be positive.
  - status fields use enums (e.g., PLACED, SHIPPED, DELIVERED, RETURNED).
  - email must follow proper format and be unique per user.
  - payment status restricted to (PENDING, SUCCESS, FAILED, REFUNDED).

4. User & Transaction Integrity

- User authentication ensures only authorized users/admins modify data.
- Transactions (orders, payments, returns) are processed **atomically** (all steps succeed or none).
- Payment details are stored with gateway tokens, not raw card numbers, to prevent corruption or leak

5. Data Validation & Consistency

- Input validation ensures correct formats (e.g., phone numbers, emails).
- Inventory updates (stock decrement on purchase, increment on returns) are handled in real-time.
- Returns/repairs automatically adjust order and stock records to remain consistent.

6. Audit & Security Integrity

- An **Audit Log** table records all critical actions (order updates, returns processed, admin changes).
- Data changes are traceable, ensuring accountability.
- Encryption is applied for sensitive data (passwords with hashing, secure payment tokens

4.2.2.2 Common Constraints

Constraints are rules applied at the database level to ensure data integrity and prevent invalid entries.

1. **Primary Key Constraint**

- Each table has a unique identifier.
  - Examples: UserID in Users, ProductID in Products, HamperID in Hampers, OrderID in Orders, PaymentID in Payments.

2. **Foreign Key Constraint**

- Maintains valid relationships between tables.
  - UserID in Orders refers to Users (UserID).
  - ProductID in Hamper Items refers to Products (ProductID).
  - OrderID in Payments refers to Orders (OrderID).
  - OrderItemID in Returns/Repairs refers to Order Items(OrderItemID).
    - Prevents invalid references or orphan records.

3. **Unique Constraint**

- Certain fields must remain unique.
  - Example: Email and Phone in Users must be unique (no duplicate registrations).
  - TxnRef in Payments must be unique to prevent duplicate transactions.
  - Offer Code in Offers must be unique.

4. **Not Null Constraint**

- Essential fields cannot be left blank.
  - Examples: User Name, Email, Password; Product Name, Price, Stock; Order Status; Payment Method.

5. Check Constraint

- Ensures values fall within valid ranges.
  - Example:
    - Price > 0
    - Stock >= 0
    - Quantity > 0
    - Status in Orders must be one of {Placed, Packed, Shipped, Delivered, Cancelled, Returned}.
    - Payment Status must be one of {Pending, Success, Failed, Refunded}.

6. **Default Constraint**

- Provides default values where applicable.
  - Example:
    - Status of a new Order defaults to *Placed.*
    - Payment status defaults to *Pending*.
    - Created At timestamp defaults to current system time.
    - Stock defaults to 0 if not provided.

4.3 Procedural Design

The procedural design describes the **step-by-step workflows** of the system, outlining how users, administrators, and the system itself interact to complete major operations. It ensures that all processes are carried out in a logical, consistent, and secure sequence.

1. User Registration & Login Procedure

1. User opens the website and chooses Register/Login.
2. Registration form collects details like name, email, phone, password, and address.
3. System validates data (unique email/phone, password rules).
4. On success, user details are stored in the **User table**.
5. For login, credentials are matched against the database, and access is granted on success.

2. Product Browsing & Selection Procedure

1. User browses the jewellery catalogue or searches by category, price, or material.
2. The system fetches product data from the **Product table**.
3. User views details such as name, description, price, stock, and images.
4. User adds chosen products or hampers to the shopping cart.

3. Gift Hamper Customization Procedure

1. User selects “Create Hamper.”
2. System displays product list; user chooses multiple products.
3. User adds personal message and selects packaging.
4. System calculates total hamper price and stores details in the **Hamper and Hamper Item tables**.

4. Order Placement & Payment Procedure

1. User reviews cart and proceeds to checkout.
2. System validates stock, calculates taxes, discounts, and offers.
3. User enters delivery address and selects payment method.
4. Payment request is sent to the payment gateway.

5. If successful, an order record is created in the **Order table** with “Placed” status.
6. System generates invoice and sends confirmation (email/SMS).

5. Order Tracking & Delivery Procedure

1. User tracks order status via the website.
2. Admin updates order status as Packed, Shipped, or Delivered.
3. Delivery tracking data is updated in the **Delivery Tracking table**.
4. User is notified at each stage until delivery is completed.

6. Returns & Repairs Procedure

1. If the product is defective/damaged, user raises a return/repair request.
2. System records the request in the **Return Repair table**.
3. Admin reviews and approves/rejects the request.
4. If approved, pickup is arranged and status is updated (In Progress, Resolved).
5. Refund, repair, or replacement is processed and user is notified.

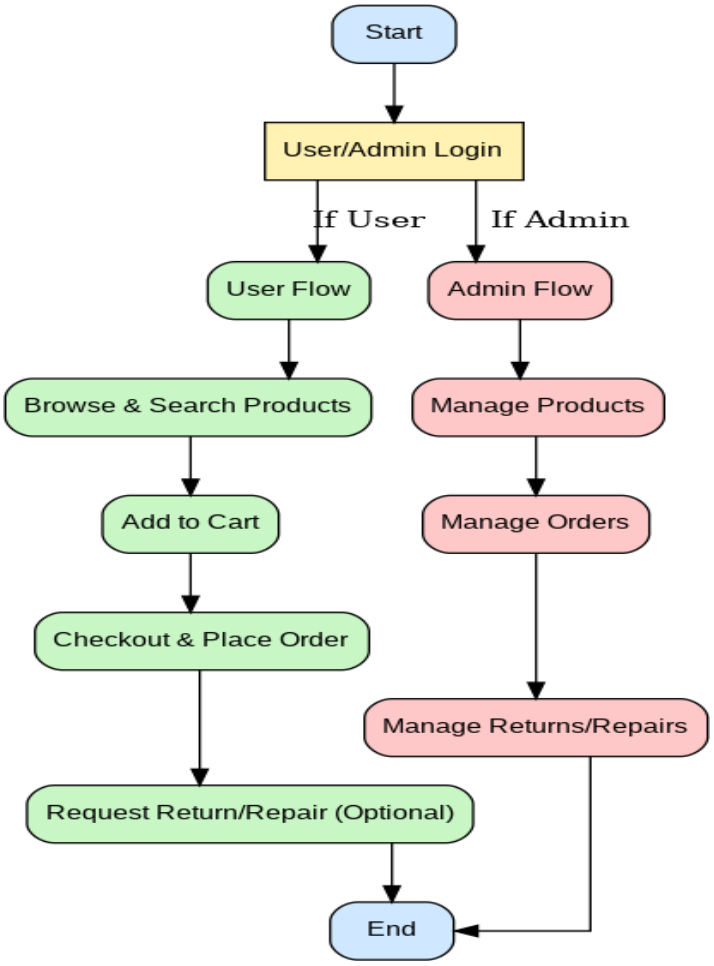
7. Offers & Promotions Procedure

1. Admin creates offers or discount codes in the **Offer table**.
2. During checkout, the system checks for valid offers.
3. Discounts are applied to the order amount if eligible.
4. Offer redemption is recorded in the **Offer Redemption table**.

8. AI Chatbot Support Procedure

1. User initiates a query through the chatbot.
2. System processes the query and matches it to FAQs or product/order data.
3. Response is given instantly (e.g., product info, order tracking).
4. If unresolved, query is escalated to admin support.

4.3.1 Logic Diagram



4.3.2 Data Structures

In the Aura Jewellery E-Commerce Website, data structures are designed to efficiently handle jewellery products, customized hampers, user accounts, orders, payments, returns/repairs, offers, and chatbot interactions. These structures ensure secure storage, smooth retrieval, and reliable manipulation of data.

1. **Arrays / Lists**

- Used to store collections of products, hampers, and cart items.
- Example: Product list when browsing jewellery or items in a customer's shopping cart.

2. **Dictionaries / Hash Maps**

- Used to store and quickly access data using keys such as IDs.
- Example: UserID → User details, ProductID → Product info, OrderID → Order record.

3. **Queues**

- Used in order processing to ensure sequential handling.
- Example: Orders placed by users are added to the order queue and processed in First In First Out (FIFO) manner.

4. Classes & Objects (OOP Structures)

The project uses **object-oriented programming** for key entities:

- **User Class** → manages registration, login, and profile data.
- **Product Class** → stores jewellery details and stock information.
- **Hamper Class** → handles hamper creation, selected products, and total pricing.
- **Order Class** → manages checkout, order status, and linking with payments.
- **Payment Class** → stores transaction details and payment status.
- **Return/Repair Class** → handles defect/repair requests and their status.
- **Offer Class** → manages promotional codes and discounts.
- **Chatbot Class** → supports user queries and escalation.
- **Admin Class** → manages product updates, order approvals, offers, and returns.

5. Database Tables (Relational Structures)

Persistent storage is managed with normalized relational tables:

- **Users** (UserID, Name, Email, Phone, Password, Address).
- **Products** (ProductID, Name, Category, Material, Price, Stock, Image, Warranty).
- **Hampers** (HamperID, UserID, Product List, Message, Packaging, Total Price).
- **Orders** (OrderID, UserID, Order Date, Total Amount, Status, Payment Status).
- **Order Items** (OrderItemID, OrderID, ProductID/HamperID, Quantity, Unit Price).
- **Payments** (PaymentID, OrderID, Method, Transaction, Amount, Status, Paid At).
- **Returns/Repairs** (RMA_ID, OrderItemID, Type, Reason, Status, Opened At, Closed At).
- **Offers** (OfferID, Code, Type, Value, StartDate, End Date).
- **Admins** (AdminID, Name, Email, Password, Role).
- **Chat Sessions** (SessionID, UserID, Query, Response, Timestamp).

4.3.3 Algorithms Design

In the **Aura Jewellery E-Commerce Website**, algorithms are designed to handle user registration, product browsing, gift hamper customization, order placement, payment processing, returns/repairs, offers, and chatbot support. The key algorithms are:

1. User Registration and Login Algorithm

Steps:

1. Input user details (name, email, phone, password).
2. Validate if email/phone already exists.
3. Encrypt password before storing.
4. On login, compare entered password with stored hash.
5. If valid, create a session and allow access; else show error.

2. Product Browsing and Search Algorithm

Steps:

1. User enters keyword or applies filters (category, price, material).
2. System searches product database.
3. Matching products are displayed in a list.
4. User can select products for viewing or adding to cart.

3. Gift Hamper Customization Algorithm

Steps:

1. User selects “Create Hamper” option.
2. Selects multiple products from catalogue.
3. Adds message and packaging preference.

4. System calculates total price = sum of product prices + packaging charges.
5. Customized hamper is saved in the database.

4. Order Placement and Checkout Algorithm

Steps:

1. User adds products or hampers to cart.
2. At checkout, system verifies stock and calculates total cost.
3. User enters shipping address and selects payment method.
4. System creates an order record with "Placed" status.

5. Payment Processing Algorithm

Steps:

1. System forwards payment details to secure gateway.
2. If payment successful → update order status to "Paid".
3. If payment fails → order is cancelled or kept pending.
4. Generate invoice and send confirmation to user.

6. Returns and Repair Algorithm

Steps:

1. User raises a return/repair request for an order item.
2. System checks eligibility (policy, time limit, condition).
3. If valid, request is forwarded to admin for approval.
4. Admin updates request as Approved/Rejected.
5. If approved → refund/repair/replacement is initiated and status updated.

7. Offers and Promotions Algorithm

Steps:

1. User enters coupon code or system applies festival/loyalty offer.
2. Validate offer (expiry date, usage limit, eligibility).
3. If valid → apply discount to total order amount.
4. If invalid → display error message.

8. AI Chatbot Algorithm

Steps:

1. User enters query (e.g., “Track my order”, “Return policy”).
2. Chatbot detects intent from the query.
3. If query matches FAQ/product/order info → give direct reply.
4. If complex → escalate to human admin support.
5. Log conversation in system.

9. Admin Management Algorithm

Steps:

1. Admin logs in securely.
2. Can add/update/remove products and offers.
3. Can monitor and update order statuses.
4. Can approve/reject return/repair requests.
5. Can generate reports for sales and performance.

4.4 User interface Design

The **User Interface (UI) Design** of the Aura Jewellery E-Commerce Website focuses on creating a **user-friendly, visually appealing, and responsive experience** for both customers and administrators. The design uses clear navigation menus, attractive product displays, and simple forms to ensure smooth interaction across all modules of the system.

1. User Module Interfaces (Customer Side)

- **Home Page** → Displays jewellery categories, featured products, offers, and navigation bar.
- **Registration/Login Page** → Allows new users to create accounts and existing users to log in securely.
- **Product Catalogue Page** → Shows jewellery products with filters (category, material, price, etc.).
- **Product Details Page** → Displays product images, description, price, stock, and "Add to Cart" option.
- **Gift Hamper Customization Page** → Lets users select multiple products, add packaging and personal message.
- **Cart Page** → Lists selected items (products/hamper) with total price and option to update/remove.
- **Checkout Page** → Collects delivery address, applies offers, and selects payment method.
- **Order Tracking Page** → Allows users to check live status of their order.
- **Return/Repair Request Page** → Users can request return/repair for defective items.
- **Chatbot Interface** → Provides AI-powered query assistance and FAQs.

2. Admin Module Interfaces

- **Admin Login Page** → Secure login for administrators.
- **Dashboard** → Overview of sales, orders, returns, and offers.
- **Product Management Page** → Add, update, or delete jewellery items and stock levels.
- **Order Management Page** → View and update customer order statuses (Placed, Shipped, Delivered).
- **Return/Repair Management Page** → Approve/reject customer requests.
- **Offer/Promotion Management Page** → Create and manage discount codes or festival offers.
- **Reports Page** → Generate sales, revenue, and performance reports.

3. UI Features

- **Responsive Design** → Works on desktop, mobile, and tablets.
- **Attractive Layout** → High-quality jewellery images, modern fonts, and clean color scheme.
- **Easy Navigation** → Simple menus, breadcrumb navigation, and search bar.
- **Security Elements** → Captcha in login/register, SSL for payments, and session timeout.

4.5 Security Issues

The **Aura Jewellery E-Commerce Website** faces several important security issues since it deals with sensitive customer details, financial transactions, and high-value jewellery products. One of the major risks is **user** authentication and identity theft, were weak passwords, phishing, or brute force attacks can allow unauthorized access to accounts. Data privacy concerns are also critical, as user information such as names, phone numbers, addresses, and emails may be stolen or misused if not properly encrypted. Online payments are another vulnerable area, with possible threats like fake payment confirmations, transaction replay, and man-in-the-middle (MITM) attacks that can compromise financial security. At the system level, poor input validation could lead to SQL injection or database breaches, enabling attackers to steal or manipulate jewellery catalogues, customer orders, or payment data. Similarly, Cross-Site Scripting **(XSS)** and Cross-Site Request Forgery (CSRF) attacks may inject harmful scripts or force users into making unauthorized transactions. Another issue is session hijacking, where attackers capture cookies or tokens to impersonate legitimate users. The returns and repair process is also at risk of being exploited by dishonest customers who may try to obtain refunds or replacements fraudulently. Finally, the admin panel represents a high-value target; if compromised, attackers could alter product prices, approve false return requests, generate fake offers, or access business reports. These security issues highlight the importance of implementing strong authentication, data encryption, secure coding, and regular monitoring to protect both customers and the business.

4.6 Test Cases Design

The testing of the Aura Jewellery E-Commerce Website is carried out using black-box testing techniques, focusing on functionality, input validation, system workflows, and security. Each test case specifies the input, expected output, and actual outcome to verify correctness.

1. User Module Test Cases

- **Registration Test Case**
  - Input: New user details (name, email, phone, password).
  - Expected Output: Account created successfully.
  - Invalid Case: Duplicate email → Show "Email already registered" error.

- **Login Test Case**
  - Input: Correct email and password.
  - Expected Output: Login successful, redirected to homepage.
  - Invalid Case: Wrong password → Show "Invalid credentials" message.

2. Product & Hamper Test Cases

- **Product Search Test Case**
  - Input: Keyword = "Necklace".
  - Expected Output: List of products matching keyword displayed.
- **Hamper Customization Test Case**
  - Input: Select multiple products, add packaging and message.
  - Expected Output: Hamper created with total price calculation.

3. Cart & Checkout Test Cases

- **Add to Cart Test Case**
  - Input: Select product and click "Add to Cart".
  - Expected Output: Product appears in cart with correct quantity and price.

- **Checkout Test Case**
  - Input: Valid address + payment method.
  - Expected Output: Order placed, order ID generated.
  - Invalid Case: Empty cart → Show "Cart is empty" error

4. Payment Test Cases

- **Successful Payment**
  - Input: Valid card/UPI details.
  - Expected Output: Transaction success, order status updated to "Paid".
- **Failed Payment**
  - Input: Wrong card number.
  - Expected Output: Payment failed, order not confirmed.

5. Order Tracking & Returns Test Cases

- **Order Tracking**
  - Input: Enter order ID.
  - Expected Output: Status displayed (Placed / Shipped / Delivered).
- **Return/Repair Request**
  - Input: Select order item → Reason: "Defective product".
  - Expected Output: Request logged, pending admin approval.

6. Admin Module Test Cases

- **Product Management**
  - Input: Add new product with all details.
  - Expected Output: Product appears in catalogue.
- **Return Approval**
  - Input: Approve return request.
  - Expected Output: Status updated to "Approved" and user notified.

7. Chatbot Test Cases

- **FAQ Query**
  - Input: “What is the return policy?”
  - Expected Output: Chatbot replies with correct policy.
- **Escalation**
  - Input: Complex query not in database.
  - Expected Output: Escalated to admin support.