

Health HUB-Empowering Wellness eCommerce Platform

A Project Report Submitted

to

MANIPAL ACADEMY OF HIGHER EDUCATION

For Partial Fulfillment of the Requirement for the

Award of the Degree

Of

Bachelor of Technology

in

Information Technology

by

Aditya Shah, Jyotishman Shandilya, Kajal Madhusudan

210911310, 210911330, 210911334

Under the guidance of

Dr. Girija Attigeri

Assistant Professor - Senior Grade

Department of I&CT

Manipal Institute of

Technology

Manipal, Karnataka, India

Mrs. Swathi B P

Assistant Professor – Senior Grade

Department of I&CT

Manipal Institute of

Technology

Manipal, Karnataka, India

November 2023



MANIPAL INSTITUTE OF TECHNOLOGY

MANIPAL

A Constituent Unit of MAHE, Manipal

ABSTRACT

Health HUB is an online wellness shop that offers a wide variety of supplements, nutritional products, and other health-related items. This project is to provide its customers with high-quality products. Health HUB also offers a variety of educational resources to help its customers make informed decisions about their health.

Health HUB's expansive product range caters to diverse health needs, encompassing vitamins, minerals, health supplements, and organic nutritional options. Through a user-friendly interface, customers can explore well-curated collections designed to support physical health, mental well-being, and overall vitality. The platform integrates the latest advancements in nutritional science, ensuring that each product meets the highest standards of quality and efficacy. The platform aims to replace traditional ledger systems with efficient inventory management, emulating larger e-commerce applications on a smaller scale.

ACM Taxonomy:

[Human-centered computing]: Health informatics;[Information systems]: E-commerce; Customer relationship management; Supply chain management; Decision support systems [Computing Methodologies]: Information retrieval;[Social and professional topics]: Wellness

Sustainable Development Goal (SDG):

SDG 3: Good Health and Well-being - This goal aims to ensure healthy lives and promote well-being for all at all ages. It addresses all major health priorities and aims to prevent needless suffering from preventable diseases and premature death. By providing access to health and wellness products, your e-commerce site can contribute to this goal.

SDG 12: Responsible Consumption and Production - By promoting health and wellness products, your site encourages responsible consumption. It could also contribute to this goal by implementing sustainable practices in its operations.

LISTS

List of tables

Customer: Stores customer information like usernames, passwords, phone numbers, emails, credit card numbers, and delivery addresses. This table serves as the repository for essential customer data, facilitating order tracking and history. It allows the platform to manage customer accounts, preferences, and transactions efficiently.

Category: Acts as a cataloging system by categorizing products through unique category IDs and names. This table enables the classification and organization of products into distinct groups, aiding in easier navigation and browsing for users.

Review: Stores user-generated reviews, including detailed descriptions and ratings for products. This table is crucial for building product credibility and assisting other users in making informed purchase decisions.

Product: Represents individual products available for purchase, housing vital information such as product names, category IDs, prices, quantities, descriptions, and images. It establishes a detailed inventory of items offered on the platform, linked to respective categories for streamlined browsing.

Seller: Holds pertinent data regarding sellers, encompassing their unique IDs, usernames, and passwords. This table manages seller profiles, facilitating their participation in the platform and ensuring secure access to their accounts.

Sells: Forms a connection between sellers and the products they offer, specifying the available quantities for sale. This table establishes the inventory relationship between sellers and the products they manage, enabling efficient stock monitoring.

Login: Manages user login credentials and roles, associating customer IDs with designated roles such as 'Seller' or 'Customer.' This table ensures secure access to the platform and distinguishes between different user types for appropriate permissions.

Gives: Links customers to their submitted reviews along with the review date. This table creates an association between users and their feedback, aiding in the organization and display of user-generated content.

Has: Establishes connections between products and reviews, indicating which reviews are related to specific products. This table enables the platform to display relevant reviews alongside respective products, enhancing user experience.

AddsToCart: Tracks items added to a customer's shopping cart, including quantities for each selected item. This table aids in managing temporary selections made by users before completing the purchase process.

OrderTable: Represents individual orders placed by customers, storing unique order IDs and associated prices. This table serves as a repository for order information, facilitating order tracking and management.

PlacesOrder: Connects customers, orders, and products, detailing quantities and prices for each product within an order. This table provides a comprehensive view of order specifics, associating products with respective customers and orders.

Payment: Manages payment details, linking payment IDs to specific orders. This table ensures proper handling and tracking of payment transactions associated with placed orders.

DeliveryService: Stores information related to delivery services associated with orders, including unique delivery IDs, service names, and contact details. This table assists in managing the logistics of order delivery and tracking.

List of Figures

1. Three-Tier Architecture
2. ER diagram
3. Relational Schema
4. Block Diagram
5. Landing Page
6. Login Page
7. Registration Page
8. Product Page
9. Cart
10. Checkout page
11. Delivery Message box
12. Seller Route

Abbreviations

1. DBS: Database System
2. DBMS: Database Management System
3. SQL: Structured Query Language
4. JS: JavaScript

Table of Contents

Health HUB-Empowering Wellness eCommerce Platform.....	0
ABSTRACT.....	1
ACM Taxonomy:.....	1
Sustainable Development Goal (SDG):.....	1
LISTS.....	2
List of tables.....	2
List of Figures.....	3
Abbreviations.....	3
Chapter 1.....	5
1.1. Introduction.....	5
Chapter 2.....	7
2.1. Literature Survey.....	7
Background:.....	8
Chapter 3.....	9
3.1. Objective.....	9
3.2. Problem Statement.....	9
Chapter 4.....	10
Data Designs.....	10
4.1. ER Diagram.....	10
4.2. Schema Diagram.....	11
4.3. Reduction Schema.....	11
4.4. Normalization (with Justification).....	12
Chapter 5.....	16
Methodology.....	16
5.1. Initial Stage.....	16
5.2. Trigger Creation:.....	18
5.3. Development.....	19
5.4. Block Diagram.....	20
5.5. Testing and Maintenance.....	20
5.6. Technologies Used.....	21
Chapter 6.....	22
6.1. Results.....	22
Chapter 7.....	28
7.1. Conclusions and Future Works.....	28
Conclusion:.....	28
Future Scope:.....	28
7.2. References.....	28

Chapter 1

1.1. Introduction

Outline:

The project revolves around the development and implementation of a comprehensive platform, Health HUB, designed to facilitate the sale of health and wellness products. The platform is intended for fitness enthusiasts, pharmaceutical vendors, payment gateway admins, logistics managers, and professionals involved in developing or implementing software systems for the Health and wellness industry.

Context:

In today's fast-paced world, maintaining a healthy lifestyle can be challenging due to busy schedules and demanding workloads. Health HUB is created as a response to this challenge, offering a wide variety of supplements, nutritional products, and other health-related items. The platform is committed to providing customers with the products they need to support their health goals. The convergence of health informatics, e-commerce methodologies, and information retrieval techniques forms the backbone of Health HUB's technological infrastructure.

Scope:

The Health and Wellness E-commerce Site, Health HUB, is designed to offer an extensive range of health and wellness products to users. It allows users to browse, select, and purchase products securely. The platform not only offers a diverse array of health products but also facilitates a user-friendly interface for an enhanced shopping experience. Additionally, Health HUB employs information retrieval techniques to provide personalized recommendations and expert-backed content, transforming the platform into a comprehensive resource for individuals seeking to optimize their health and lifestyle. The platform aims to transcend traditional boundaries, emerging as a beacon in the pursuit of good health and well-being for individuals worldwide.

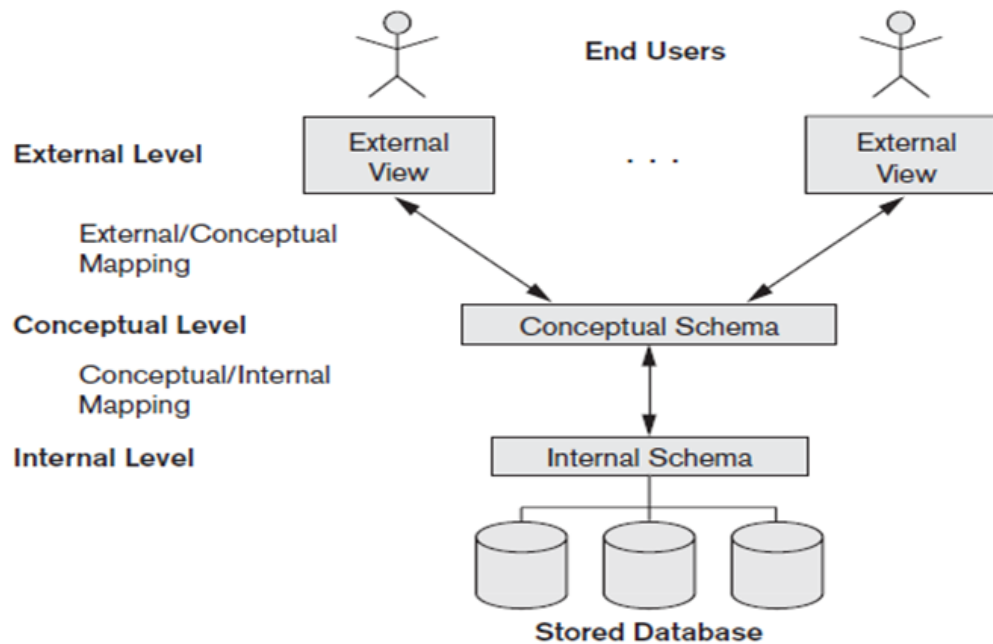


Fig.1. Basic Database Flowchart

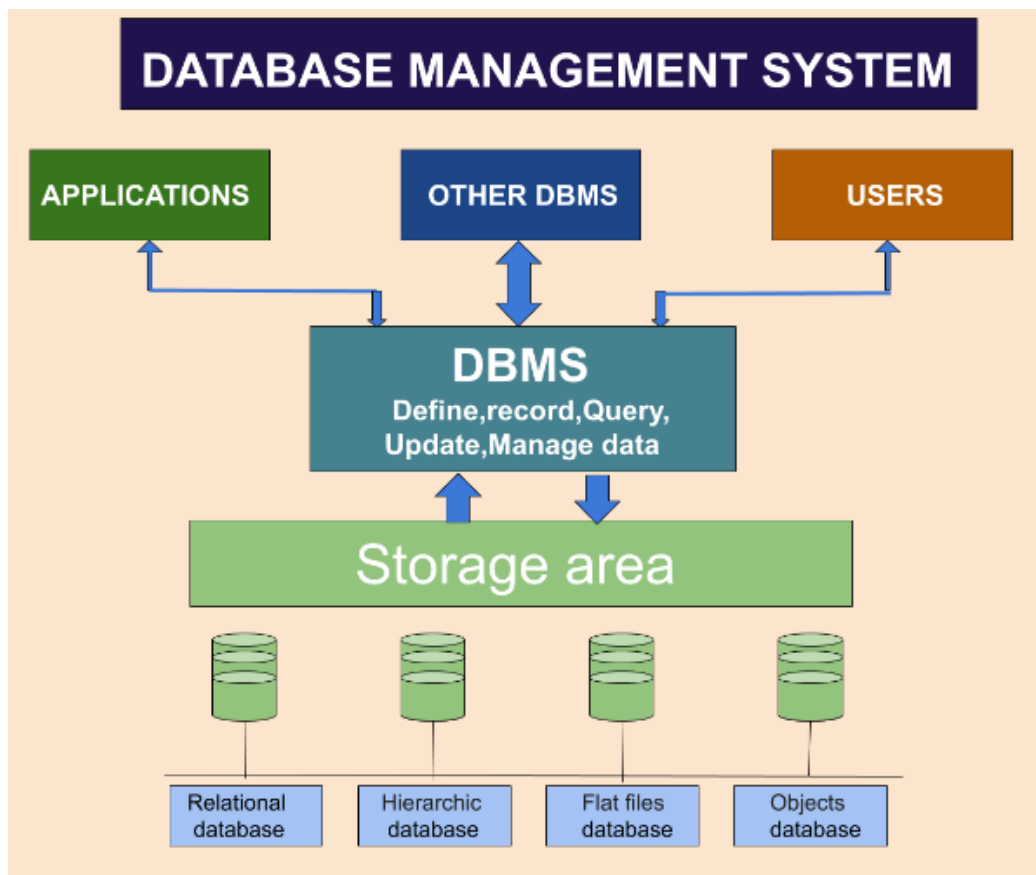


Fig.2. Representation of using DBMS

Chapter 2

2.1. Literature Survey

Health HUB is designed to integrate insights derived from cutting-edge research, ensuring a user-centric approach in the realm of health and wellness e-commerce. Employing Next.js and Tailwind CSS, Health HUB offers an intuitively navigable interface and personalized experiences, resonating with the emphasis on user-centric design in e-commerce studies. Leveraging PostgreSQL, the platform curates tailored content and recommendations, aligning with research advocating information retrieval for personalized user experiences. Embracing human-centered design principles, Health HUB merges technological advancements seamlessly into a user-friendly platform. Beyond technology, Health HUB encompasses a comprehensive array of health products, embodying its vision to be the ultimate destination for users seeking personalized wellness solutions.

Recent studies in health informatics, such as Yao Zheng's work [1], highlight technology's crucial role in enhancing health outcomes. This parallels Health HUB's emphasis on transparency and user empowerment. Similarly, research by Yi Jin Lim et al. [2] explores e-commerce factors, emphasizing user-friendly interfaces and personalized experiences, aligning with Health HUB's mission to offer a seamless shopping journey. Moreover, Martina et al.'s focus [3] on human-centered design aligns with Health HUB's commitment to intuitively designing a technologically advanced platform for user satisfaction in the wellness landscape.

Furthermore, insights from e-commerce research underline the significance of user-centric design and personalized experiences in shaping online shopping behavior. Health HUB resonates with these findings, employing technologies like Next.js and Tailwind CSS to craft a user-friendly interface. Next.js, as a React framework, enables server-rendered applications, ensuring a smooth, interactive, and personalized shopping experience. Meanwhile, Tailwind CSS, with its utility-first approach, allows for agile and adaptable styling, catering to diverse user preferences.

Moreover, the emphasis on information retrieval techniques for personalized recommendations in studies echoes Health HUB's approach to leveraging PostgreSQL. This open-source relational database management system plays a pivotal role in tailoring content and recommendations to individual user preferences. It helps Health HUB deliver expert-backed content aligned precisely with unique health needs, fostering a more engaging and relevant user experience.

In essence, Health HUB's tech stack, comprising JavaScript for interactivity, Next.js for server-rendered applications, Tailwind CSS for adaptable styling, and PostgreSQL for personalized content delivery, aligns seamlessly with the research findings. These technologies

collectively form the backbone of Health HUB's innovative platform, ensuring a user-centric approach and personalized experience in the realm of health and wellness e-commerce.

Background:

In response to the challenges posed by modern lifestyles and the increasing need for accessible wellness solutions, Health HUB emerged. This innovative Health and Wellness E-commerce Site aims to bridge the gap between demanding schedules and the pursuit of improved health. Leveraging advancements in technology, commerce, and health informatics, Health HUB endeavors to provide a holistic platform. It not only offers diverse wellness products but also prioritizes user experience and expert-backed information dissemination. With a vision to revolutionize the intersection of health, technology, and commerce, Health HUB aspires to be a global beacon for enhanced health and well-being.

Chapter 3

3.1. Objective

The primary objective of the "HealthHub" initiative is to revolutionize the way individuals engage with and prioritize their health. Leveraging technological advancements and a strong commitment to user-centric design, the HealthHub platform has been meticulously crafted to seamlessly merge a broad spectrum of health products with immersive wellness experiences. Its core principle is to provide an intuitive, user-friendly interface that simplifies the process of exploring and selecting products. The platform transcends traditional e-commerce environments by offering personalized customer engagement and tailored recommendations that align perfectly with individual health goals and preferences.

One of the standout features of the HealthHub platform is its robust inventory management system, which is seamlessly integrated within a seller dashboard. This strategic integration ensures streamlined operational efficiency and empowers administrators to manage and distribute products with precision and ease. The ultimate vision is to position HealthHub as a beacon of trust and excellence, deeply rooted in the ethos of wellness and relentless innovation in the health and e-commerce sectors. This is achieved through the various functionalities implemented, such as user registration, product display, cart management, order placement, and review system, all of which are supported by a well-structured database design.

3.2. Problem Statement

The current healthcare landscape confronts significant challenges, including fragmented services, limited accessibility to information, and a dearth of community-oriented support. These issues hinder individuals' ability to access comprehensive healthcare resources and contribute to suboptimal health outcomes. Establishing a Health HUB seeks to combat these obstacles by creating a centralized platform that adopts a holistic approach to healthcare. This platform aims to facilitate collaboration among healthcare providers, offer easily accessible health information, and cultivate a supportive community. By addressing these challenges, Health HUB endeavors to promote better health outcomes and empower individuals to proactively manage their well-being in a more cohesive and informed manner.

Chapter 4

Data Designs

4.1. ER Diagram

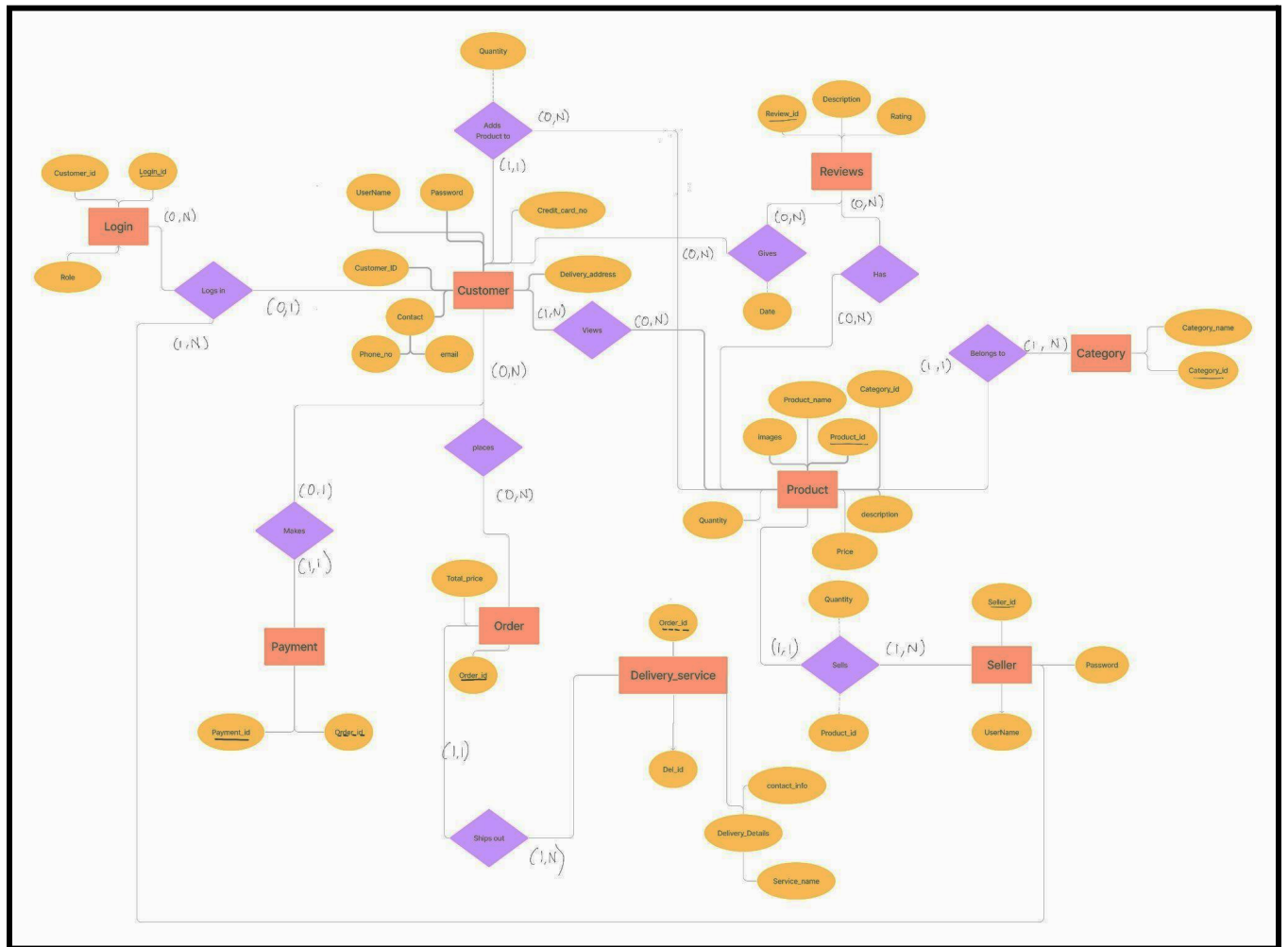


Fig.3. ER Daiagram

4.2. Schema Diagram

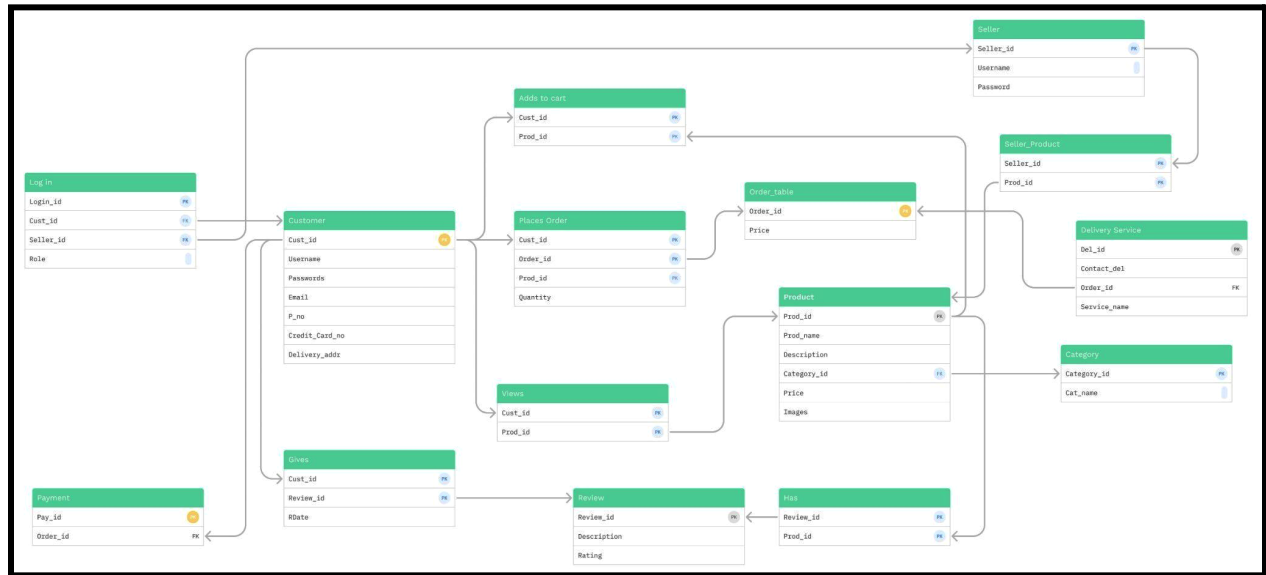


Fig.4. Reduced ER Schema

4.3. Reduction Schema

LOGIN(Login_id, Cust_id, Role)
 CUSTOMER(Cust_id, Username, Password, P_no, Email, Credit_card_no, Delivery_addr);
 CATEGORY(Cat_id, Cat_name);
 REVIEW(Review_id, Description, Rating);
 PRODUCT(P_id, P_name, Cat_id, Price, Description, Image1, Quantity);
 GIVES(Cust_id, Review_id, RDate);
 SELLER(Seller_id, Username, Password);
 SELLS(Seller_id, P_id);
 HAS(Review_id, P_id);
 PLACES_ORDER(Cust_id, Order_id, P_id)
 ADDS TO CART(Cust_id, P_id, Quantity);
 ORDER_TABLE(Order_id, Price);
 PAYMENT(Pay_id, Order_id);
 DELIVERY_SERVICE(Del_id, Contact_del, Order_id, Service_name);

4.4. Normalization (with Justification)

Customer Table

Attributes: Cust_id (A), Username (B), Password (C), P_no (D), Email (E), Credit_Card_no (F), Delivery_addr (G)

Functional Dependencies:

A → B, C, D, E, F, G

B → A, C, D, E, F, G

E → A, B, C, D, F, G

Candidate Keys: A, B, E

Primary Key: A

The table is in 1NF as it has atomic values in each cell. It is in 2NF as there are no partial dependencies. It is in 3NF as there are no transitive dependencies. It is also in BCNF as every determinant is a candidate key.

Category Table

Attributes: Cat_id (H), Cat_name (I)

Functional Dependencies:

H → I

I → H

Candidate Keys: H, I

Primary Key: H

The table is in 1NF as it has atomic values in each cell. It is in 2NF as there are no partial dependencies. It is in 3NF as there are no transitive dependencies. It is also in BCNF as every determinant is a candidate key.

Review Table

Attributes: Review_id (J), Description (K), Rating (L)

Functional Dependencies:

J → K, L

Candidate Key/Primary Key: J

The table is in 1NF as it has atomic values in each cell. It is in 2NF as there are no partial dependencies. It is in 3NF as there are no transitive dependencies. It is also in BCNF as every determinant is a candidate key.

Product Table

Attributes: P_id(M), P_name(N), Cat_id(H), Price(O), Description(P), Image1(Q), Quantity (R)

Functional Dependencies:

$M \rightarrow N, H, O, P, Q, R$

Candidate Key/Primary Key: M

The table is in 1NF as it has atomic values in each cell. It is in 2NF as there are no partial dependencies. It is in 3NF as there are no transitive dependencies. It is also in BCNF as every determinant is a candidate key.

Seller Table

Attributes: Seller_id (R), Username (S), Password (T)

Functional Dependencies:

$R \rightarrow S, T$

Candidate Keys: R, S

Primary Key: R

The table is in 1NF as it has atomic values in each cell. It is in 2NF as there are no partial dependencies. It is in 3NF as there are no transitive dependencies. It is also in BCNF as every determinant is a candidate key.

Sells Table

Attributes: Seller_id (R), P_id (M)

Functional Dependencies:

None

Candidate Keys: R, M

Primary Key: R, M

The table is in 1NF as it has atomic values in each cell. It is in 2NF as there are no partial dependencies. It is in 3NF as there are no transitive dependencies. It is also in BCNF as every determinant is a candidate key.

Login Table

Attributes: Login_id (U), Cust_id (A), Seller_id (R), Role (V)

Functional Dependencies:

$U \rightarrow A, R, V$

Candidate Key/Primary Key: U

The table is in 1NF as it has atomic values in each cell. It is in 2NF as there are no partial dependencies. It is in 3NF as there are no transitive dependencies. It is also in BCNF as every determinant is a candidate key.

Gives Table

Attributes: Cust_id (A), Review_id (J), RDate (W)

Functional Dependencies:

$A, J \rightarrow W$

Candidate Keys: A, J

Primary Keys: A, J

The table is in 1NF as it has atomic values in each cell. It is in 2NF as there are no partial dependencies. It is in 3NF as there are no transitive dependencies. It is also in BCNF as every determinant is a candidate key.

Has Table

Attributes: P_id (M), Review_id (J)

Functional Dependencies:

None

Candidate Keys: M, J

Primary Keys: M, J

The table is in 1NF as it has atomic values in each cell. It is in 2NF as there are no partial dependencies. It is in 3NF as there are no transitive dependencies. It is also in BCNF as every determinant is a candidate key.

AddsToCart Table

Attributes: Cust_id (A), P_id (M), Quantity (X)

Functional Dependencies:

A, M \rightarrow X

Candidate Keys: A, M

Primary Keys: A, M

The table is in 1NF as it has atomic values in each cell. It is in 2NF as there are no partial dependencies. It is in 3NF as there are no transitive dependencies. It is also in BCNF as every determinant is a candidate key.

OrderTable

Attributes: Order_id (Y), Price (O)

Functional Dependencies:

Y \rightarrow O

Candidate Key/Primary Key: Y

The table is in 1NF as it has atomic values in each cell. It is in 2NF as there are no partial dependencies. It is in 3NF as there are no transitive dependencies. It is also in BCNF as every determinant is a candidate key.

PlacesOrder Table

Attributes: Cust_id (A), Order_id (Y), P_id (M), Quantity (X), Price (O)

Functional Dependencies:

A, Y, M \rightarrow X, O

Candidate Keys: A, Y, M

Primary Keys: A, Y, M

The table is in 1NF as it has atomic values in each cell. It is in 2NF as there are no partial dependencies. It is in 3NF as there are no transitive dependencies. It is also in BCNF as every determinant is a candidate key.

Payment Table

Attributes: Pay_id (Z), Order_id (Y)

Functional Dependencies:

$Z \rightarrow Y$

Candidate Key/Primary Key: Z

The table is in 1NF as it has atomic values in each cell. It is in 2NF as there are no partial dependencies. It is in 3NF as there are no transitive dependencies. It is also in BCNF as every determinant is a candidate key.

DeliveryService Table

Attributes: Del_id (AA), Order_id (Y), Service_name (BB), Contact_del (CC)

Functional Dependencies:

$AA \rightarrow Y, BB, CC$

Candidate Key/Primary Key: AA

The table is in 1NF as it has atomic values in each cell. It is in 2NF as there are no partial dependencies. It is in 3NF as there are no transitive dependencies. It is also in BCNF as every determinant is a candidate key.

Chapter 5

Methodology

5.1. Initial Stage

- Requirements Gathering and Analysis

The first step in our project methodology is to gather requirements and analyze them to determine the scope and objectives of the e-commerce platform. This includes interviewing customers and paying attention to their feedback defining user requirements for ease of use. Our goal during this phase is to identify the basic features and functionalities that the platform should have.

- Database Design and Implementation

Once we have identified the requirements and objectives, we move on to the design and implementation of the inventory database. This phase includes designing the database schema, tables, and relationships. We also identified the data types and constraints for each table column. After designing the database, we implemented it using PostgreSQL and database management tools. Necessary condition checks have also been used to secure the database from any insecurities and inconsistencies.

- Frontend design and implementation of the database

After the design and implementation of the database, we proceeded to design and implement the front end of the platform using NextJs. This involved designing the user interface for various pages such as the landing page, product page, and the checkout page. We also created routes for the registration, login, and seller pages. The front end was then integrated with a local PostgreSQL server using the pool library.

Our primary goal during this phase was to create a user-friendly interface that allows users to interact with the system effortlessly. Initially, users are directed to the registration page where they can register as either a customer or a seller. This information is stored in the **Customer** and **Seller** tables respectively. After submitting all the required details, the user is redirected to the login page. The **Login** table is used here for an extra step of validation at the time of login.

Upon successful login, users are redirected to a landing page that provides an outline of the product catalog on offer. The **Product** table is used here to display all the products along with their details such as price and description. Users can navigate the application as per their requirements and choose any product they wish to purchase.

Upon clicking on a product, a detailed description of the product is displayed along with reviews given by other users. This information is fetched from the **Product** and **Review** tables. Users also have the functionality to add reviews, which are stored in the **Gives** and **Has** tables.

Users can add products to their cart, and upon clicking the cart button, they will be able to see all the products in their cart. This information is stored in the **AddsToCart** table. Users then have the option to either checkout or continue shopping. After clicking on checkout, users can see their order summary with all the product details and the final price of the product. This information is fetched from the **OrderTable** and **PlacesOrder** tables. The product quantity gets updated real-time from all the tables.

After a successful order has been placed, users will be displayed with an order confirmation with the details of their credit card number and delivery address along with the delivery service name and their contact details. This information is fetched from the **Customer**, **Payment**, and **DeliveryService** tables.

We have implemented a feature that allows users to view their profiles, which includes the option to see all their previous order history and reviews. This feature retrieves data from the **Customer**, **PlacesOrder**, **OrderTable**, **Product**, **Review**, **Gives**, and **Has** tables. It displays all the details of the products the customer has purchased from the **Product** table and the reviews they have made from the **Review** table.

We have also added a seller router. Upon successful login, the seller is redirected to a form to add product details. This information is stored in the **Product** and **Sells** tables. A trigger checks if the **Product** table needs to be updated or if a new product needs to be inserted. This methodology, which is based on the **Seller**, **Login**, and **Sells** tables, ensures a smooth and user-friendly experience for our users.

- Functional modules:

The functional modules included in the project are listed below:

- Insert Module:

This module provides the functionality of collecting the required data from the designed interface and transmitting it to the appropriate table present in the database designed for this project. If the provided data does not satisfy the given constraints, it must refrain from storing it into the database.

- Update Module:

This module again has the functionality of collecting the data from the designed interface, but it updates the already existing tuple that matches the provided primary key of the tuple to be updated, by replacing the existing attribute values with the newly collected data. Again, if the newly provided data does not satisfy the given constraints, it must refrain from updating the corresponding tuple.

- **Delete Module:**
The delete counterpart is loaded with the ability to delete a single or multiple record from the table. It searches for the tuple, in the query specified table, based on the provided value for an attribute. Admin can delete in the interface, based on which delete module searches for the record corresponding to that provided attribute value and deletes the record.
- **Retrieve Module:**
The retrieve module has a basic functionality of accessing the entire specified table from the database and displays it.
- **Trigger Module:**
Trigger in database is set of statements that are executed after an event occurs on the specified table. This is useful for logs wherein every change in the database can be logged which helps keep a track of all changes/transactions on the database.

5.2. Trigger Creation:

1. Update in Product Table:

```
CREATE OR REPLACE FUNCTION product_insert_update() RETURNS TRIGGER AS $$
BEGIN
  IF EXISTS (SELECT 1 FROM Product WHERE P_id = NEW.P_id) THEN
    UPDATE Product SET Quantity = Quantity + NEW.Quantity WHERE P_id = NEW.P_id;
    RETURN NULL;
  ELSE
    RETURN NEW;
  END IF;
END;
$$ LANGUAGE plpgsql;
```

2. Update on Add to Cart:

```
CREATE TRIGGER product_insert_update BEFORE INSERT ON Product
FOR EACH ROW EXECUTE FUNCTION product_insert_update();
```

Cart quantity:

```
BEGIN
  DECLARE
    product_quantity INT;
  BEGIN
    SELECT Quantity INTO product_quantity FROM Product WHERE P_id = NEW.P_id;
    IF product_quantity <= 1 THEN
      RAISE EXCEPTION 'Cannot add product to cart. Product quantity should be more than 1.';
    END IF;
    RETURN NEW;
  END;
END;
```

3. Insert in Customer table for Username:

```
CREATE OR REPLACE FUNCTION check_customer_exists() RETURNS TRIGGER AS $$
BEGIN
  IF EXISTS (SELECT 1 FROM Customer WHERE Username = NEW.Username) THEN
    RAISE EXCEPTION 'Customer with username % already exists', NEW.Username;
  END IF;
  RETURN NEW;
END;
$$ LANGUAGE plpgsql;

CREATE TRIGGER customer_exists_before_insert_trigger
BEFORE INSERT ON Customer
FOR EACH ROW EXECUTE PROCEDURE check_customer_exists();
```

4. Insert in Seller table for Username:

```
CREATE OR REPLACE FUNCTION check_seller_exists() RETURNS TRIGGER AS $$
BEGIN
  IF EXISTS (SELECT 1 FROM Seller WHERE Username = NEW.Username) THEN
    RAISE EXCEPTION 'Seller with username % already exists', NEW.Username;
  END IF;
  RETURN NEW;
END;
$$ LANGUAGE plpgsql;

CREATE TRIGGER seller_exists_before_insert_trigger
BEFORE INSERT ON Seller
FOR EACH ROW EXECUTE PROCEDURE check_seller_exists();
```

5.3. Development

- Environment Setup: Install Node.js, Next.js, Tailwind CSS, and PostgreSQL for development.
- Backend API Creation: Utilize PostgreSQL for database setup and use Next.js for creating backend API endpoints.
- Frontend Development: Develop the frontend application using Next.js, Tailwind CSS, and JavaScript.
- Integration: Use Next.js like pool libraries to integrate frontend and backend functionalities.

5.4. Block Diagram

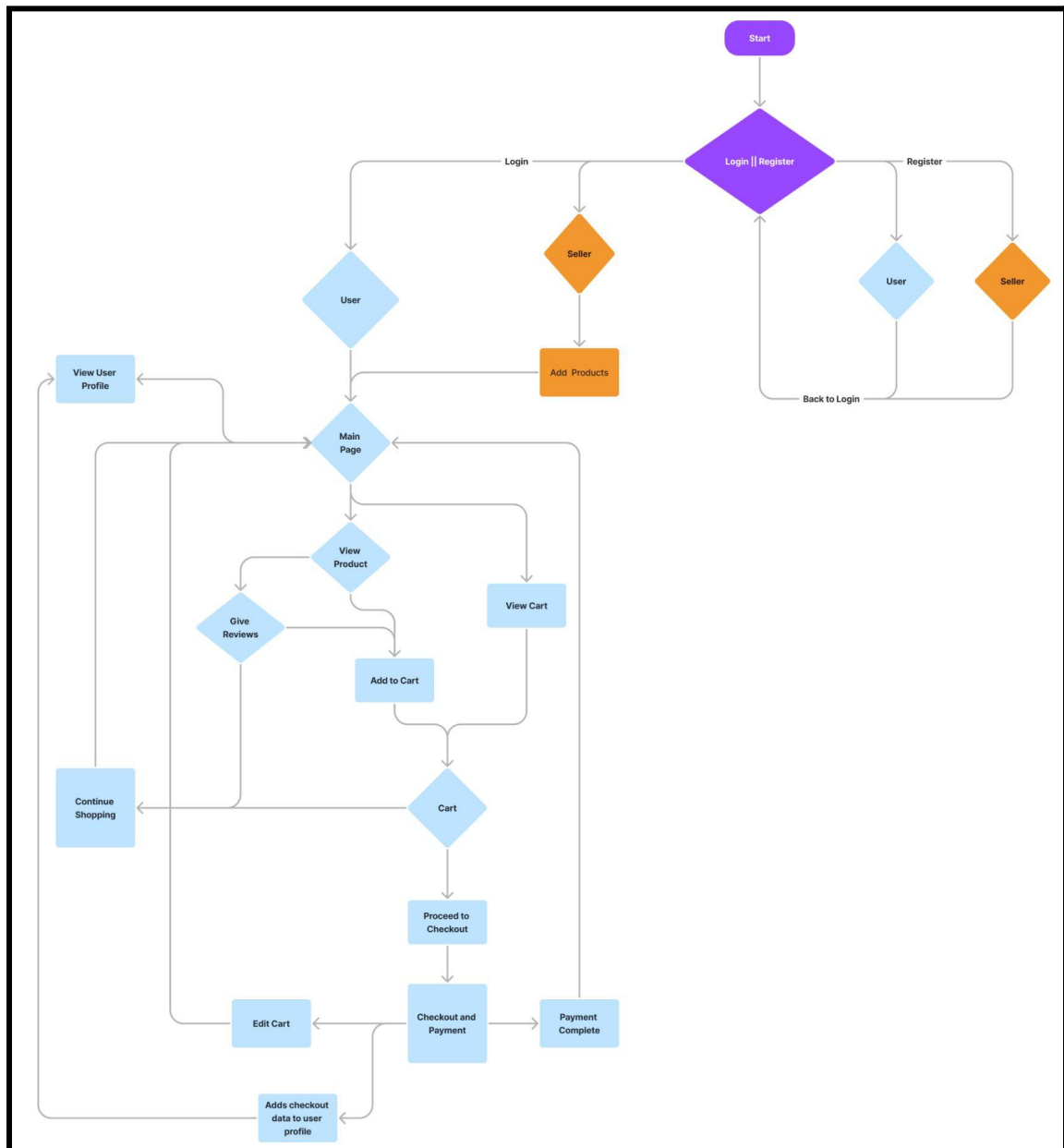


Fig.5. Block Diagram of Methodology

5.5. Testing and Maintenance

- Once the data has been added, we conducted system testing and quality assurance to ensure that the platform is functioning correctly. We tested each functionality, feature, and use case to ensure that it meets the user requirements and objectives. We also

identified and fixed all the bugs or issues that we found during the testing. This includes validating all the SQL queries through the pgadmin4 tool.

- Proper data validity has also been provided at each step of the process for an interactive experience for the users and ease of use.
- Finally, we will provide ongoing maintenance and upgrades to the platform to ensure that it continues to meet the needs of the end-users and its smooth functioning. This includes monitoring performance, identifying and fixing any issues or bugs, and implementing new features or functionalities as needed.

5.6. Technologies Used

- JavaScript: A scripting language for interactivity.
- Next.js: React framework for server-rendered applications.
- Tailwind CSS: Utility-first CSS framework for styling.
- PostgreSQL: Open-source relational database management system.
- Figma: Collaborative interface design tool.

Chapter 6

6.1. Results

The platform provides a detailed specification of the requirements for an e-commerce database management system. This system will be able to manage a database of products, customers, and orders for an e-commerce website. The system will provide functionality for adding products, managing customer accounts, processing orders, and keeping track of customer details. The platform offers a comprehensive blueprint for an e-commerce database management system. This system is designed to manage a wide range of data, thereby serving as the backbone of an e-commerce website.

It allows for the addition and management of products, each with its unique details such as name, category, price, description, image, and quantity. It also keeps track of customer details, including their username, password, phone number, email, credit card number, and delivery address.

The system works in tandem to process orders, keeping a record of each order's details and quantity. An extra layer of security is added, validating user credentials during login. It simplifies the purchase process by providing a user-friendly interface. Customers can add products to their cart and view them later, enhancing the shopping experience.

One of the features of this system is its ability to manage reviews. It handles the storage and retrieval of product reviews, contributing to a more informed purchase decision for customers. In summary, this platform streamlines the e-commerce process, from product addition to order placement, while also keeping track of inventory and customer reviews. This comprehensive approach benefits all stakeholders involved in the application, from customers to sellers, and it simplifies the purchase process and helps keep track of inventory along with their reviews which is a bonus for all stakeholders in the application.

Health Hub

User Profile Sign In Register Cart

Register

Username:

Password:

Role:

Customer

Name:

Phone Number:

Email:

Credit Card Number:

Delivery Address:

Register

Fig.6. Registration Page

Login

Username:

Password:

Role:

Customer

Log In

Forgot Password?

Fig.7. Login Page

Your Health and Wellness Partner

Health Hub

Your one-stop shop for health and wellness products

Our Products

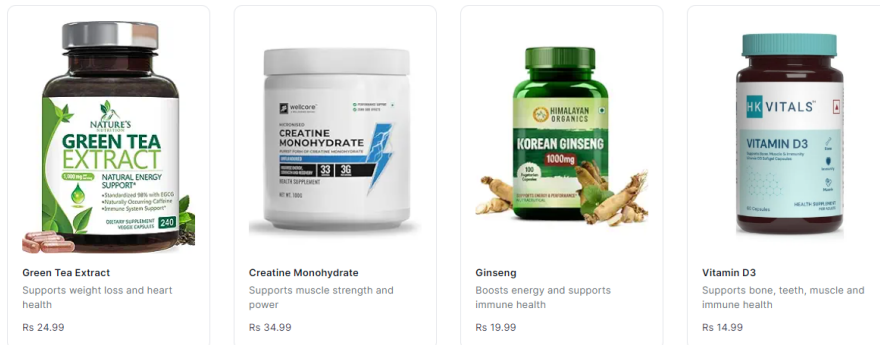


Fig.8. Landing Page



Green Tea Extract

Price: Rs. 24.99

2 reviews

Reviews

Excellent quality, noticed improvements within a week.

Rating: 5/5

AHbsdhasb

Rating: 8/5

Add a Review

Rating

0

Review

Add Review

Add to Cart

Fig.9. Product Page

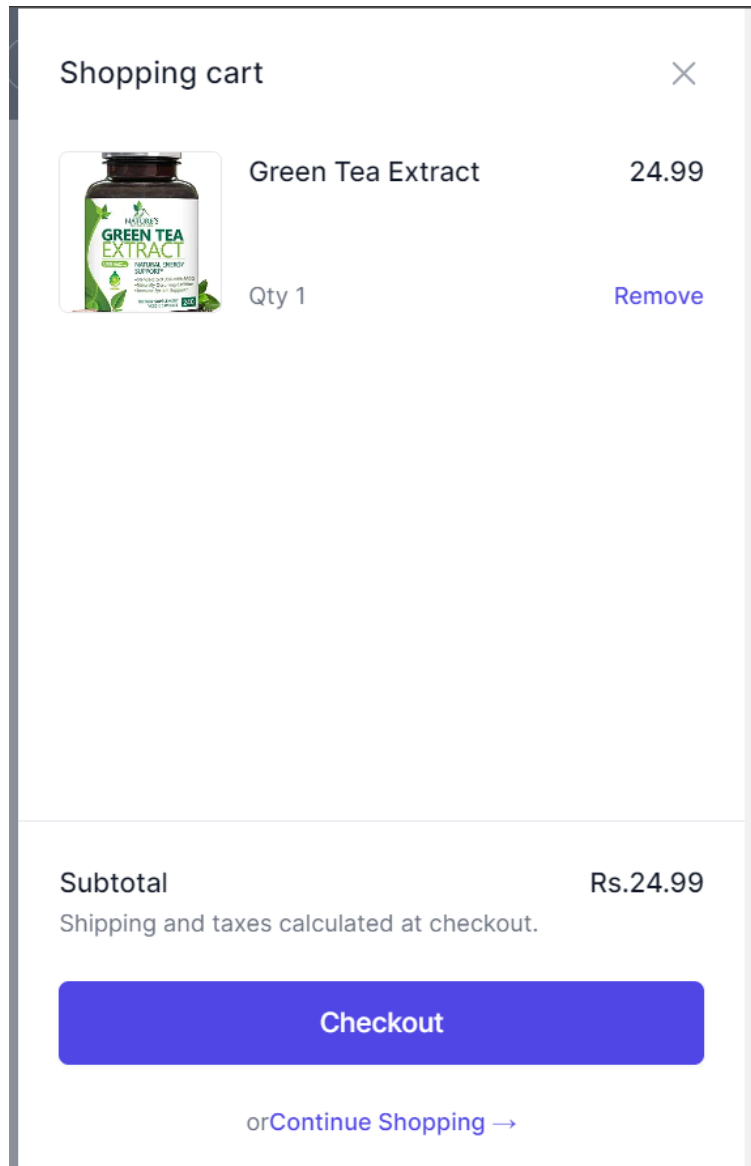


Fig.10. Cart

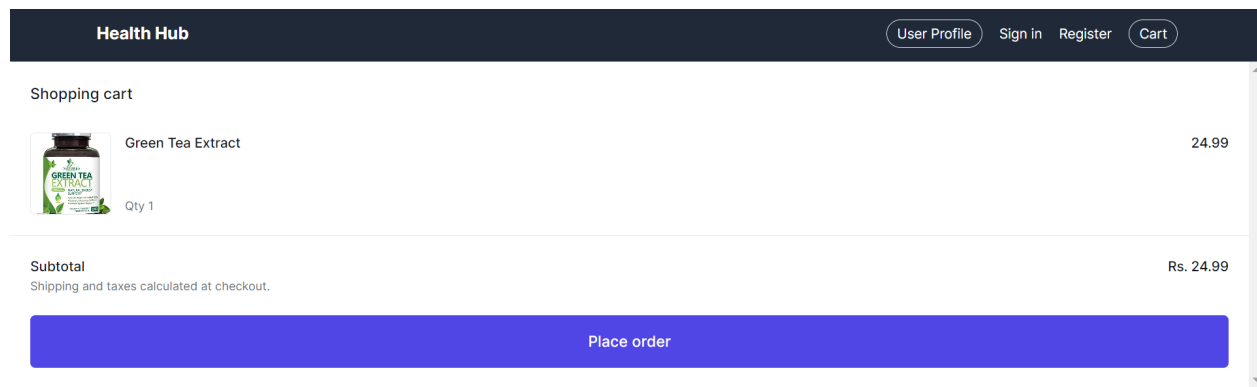


Fig.11. Checkout Page

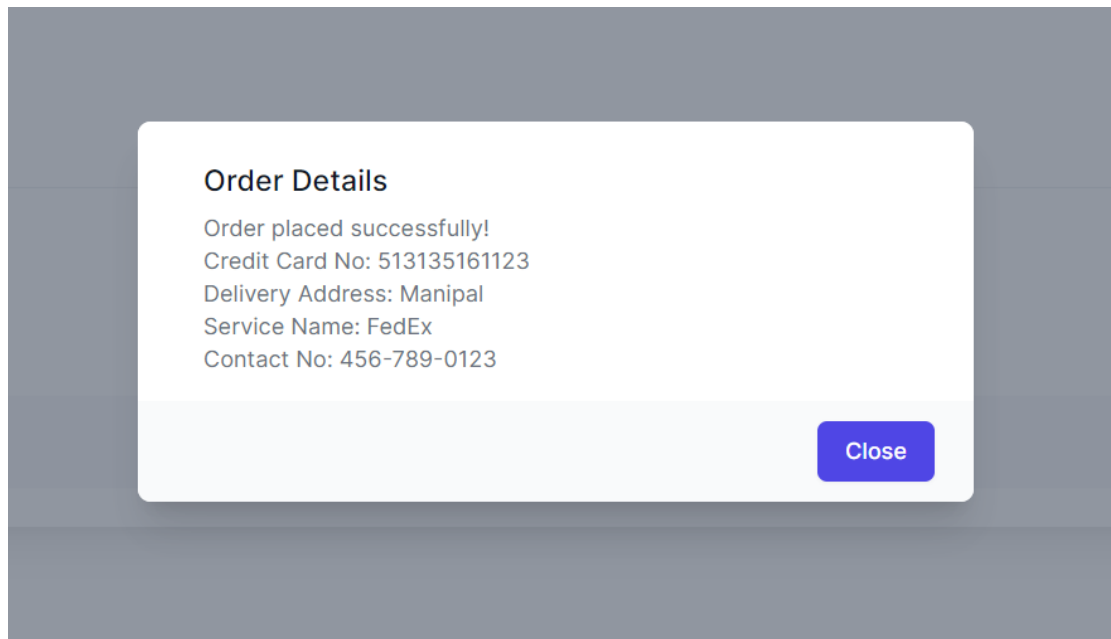


Fig.12. Delivery message Box

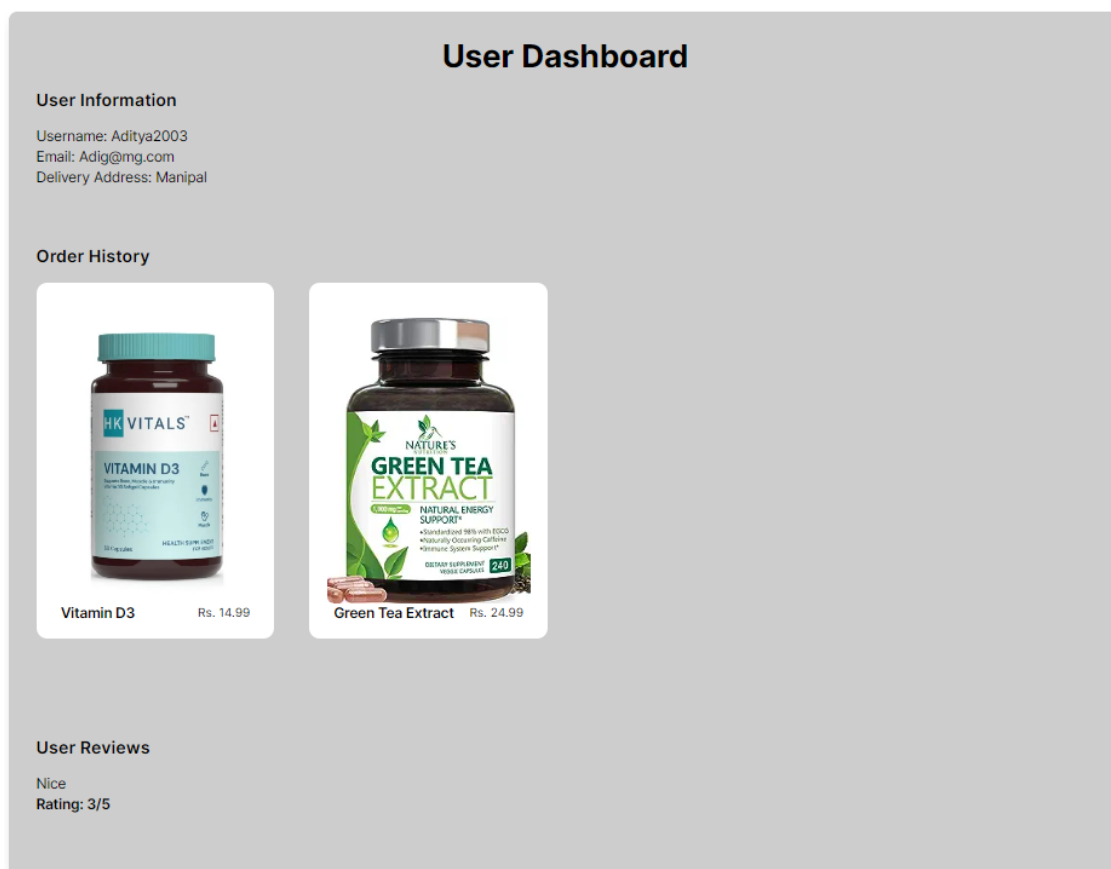
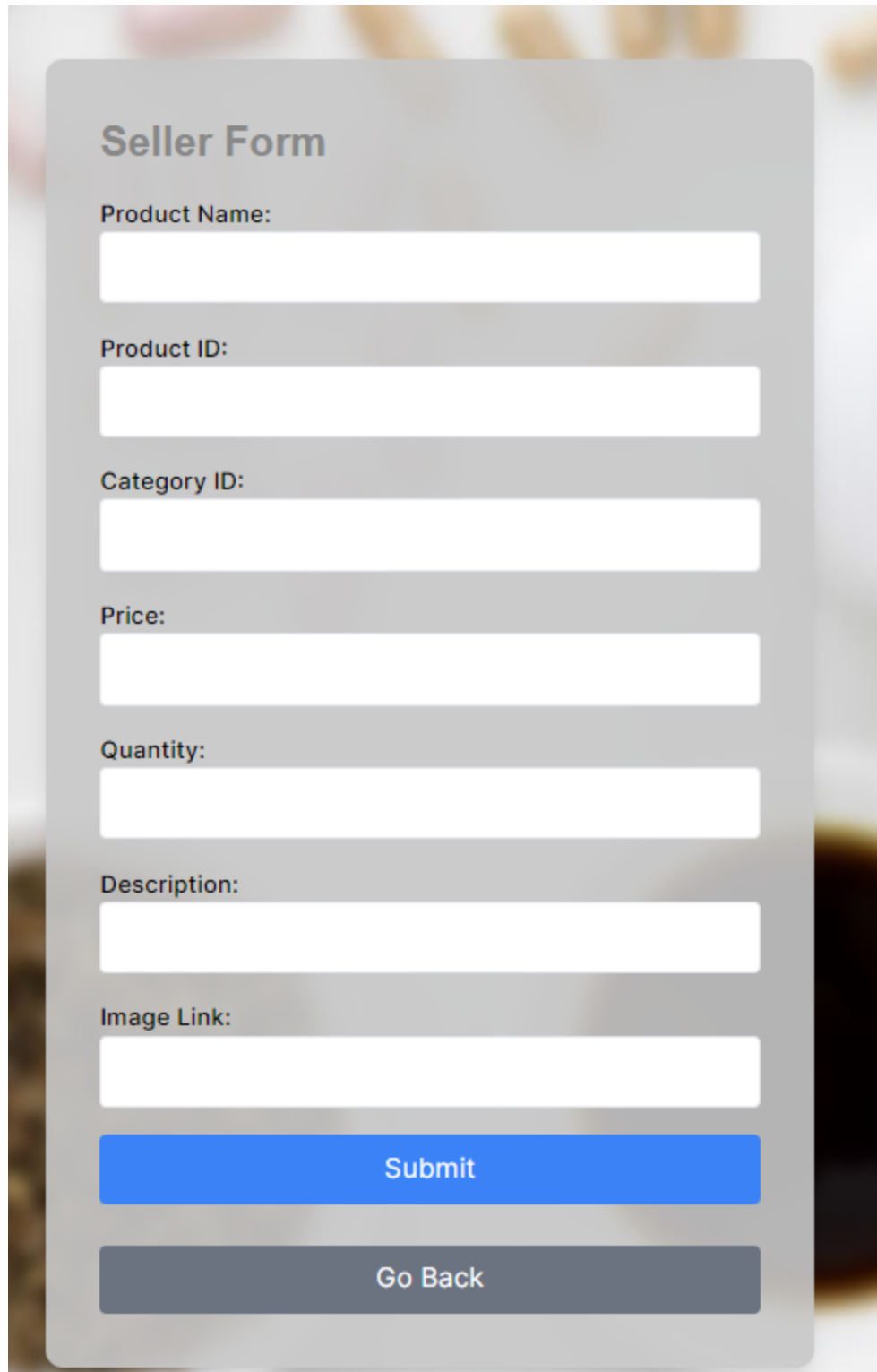


Fig.13. User Dashboard

The image shows a 'Seller Form' interface. It is a light gray rounded rectangle with a title 'Seller Form' at the top. Below the title are seven input fields, each with a label to its left: 'Product Name:', 'Product ID:', 'Category ID:', 'Price:', 'Quantity:', 'Description:', and 'Image Link:'. The first six fields are single-line text inputs, while the 'Description:' field is a multi-line text area. At the bottom of the form are two buttons: a blue 'Submit' button and a gray 'Go Back' button. The background of the entire image is a blurred photograph of coffee beans.

Seller Form

Product Name:

Product ID:

Category ID:

Price:

Quantity:

Description:

Image Link:

Submit

Go Back

Fig.14. Seller Route

Chapter 7

7.1. Conclusions and Future Works

Conclusion:

In summary, the e-commerce platform developed in this project has proven to be a powerful tool for simplifying the shopping experience. It effectively manages customer data and product data. The platform reduces manual work involved in management, making it less time-consuming and requiring minimal effort to use. The user-friendly interface is self-explanatory and caters to both registered and new users. The simplicity of the platform, coupled with door-to-door delivery, enhances the convenience factor. As e-commerce platforms continue to evolve and reach new heights, it's crucial to understand and simulate the specifics at a granular level.

Future Scope:

Looking ahead, there are several opportunities for further development and enhancement of this platform. One potential area of expansion is the integration of advanced analytics to provide insights into customer behavior and product performance. This could aid in inventory management and enable personalized marketing strategies. Additionally, the platform could benefit from the incorporation of more robust security measures to further protect user data. Implementation of the OTP mechanism, and two-step verification system could also be some far-sighted technological advances to this system. Lastly, exploring the integration of AI and machine learning algorithms for features like product recommendation and customer service could significantly enhance the user experience and operational efficiency of the platform.

7.2. References

- Yao Zheng "Privacy-Preserving Personal Health Record System Using Attribute-Based Encryption " Journal of Medical Systems, 2011.
- Yi Jin Lim, et al. "Factors influencing online shopping behavior: The mediating role of purchase intention." Journal of Retailing and Consumer Services, 2015.
- Y Liu "Personalized content recommendation in e-commerce." IEEE Transactions on Knowledge and Data Engineering, 2022.
- Martina Ziefle, et al. "Human-centered design of a distributed platform for health information exchange." Journal of Human-Centered Design of E-Health Technologies, 2019.

- <https://nextjs.org/docs>
- <https://www.postgresql.org/docs/current/>
- <https://v2.tailwindcss.com/docs>
- <https://www.w3schools.io/learn/nextjs-tutorial/>
- Ji, Z., Pi, H., Wei, W., Xiong, B., Woźniak, M., and Damasevicius, R. Recommendation based on review texts and social communities: a hybrid model. IEEE Access, 2019.