



DMART

ON

Submitted in partial fulfillment of the requirements
of the degree of

**Bachelor of Engineering
(Information Technology)**

By

Miss. Anuprita Mhapankar - Roll No (28)

Under the guidance of

Prof. Dipti Karani



Department of Information Technology

**VIVEKANAND EDUCATION SOCIETY'S INSTITUTE OF TECHNOLOGY, Chembur, Mumbai
400074**

(An Autonomous Institute, Affiliated to University of Mumbai)



Vivekanand Education Society's Institute of Technology

(Autonomous Institute Affiliated to University of Mumbai, Approved by AICTE & Recognised by Govt. of Maharashtra)
NAAC accredited with 'A' grade

April 2024

Certificate

This is to certify that project entitled

"DMART"

Group Members Names

Miss. Anuprita Mhapankar - Roll No (28)

In fulfillment of degree of BE. (Sem.VI) in Information Technology for Project is approved.

**Prof. Dipti Karani
Project Mentor**

External Examiner

**Dr.(Mrs.)Shalu
Chopra
H.O.D**

**Dr.(Mrs.)J.M.Nair
Principal**

Date:08 /04 /2025
Place: VESIT, Chembur

Declaration

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

(Signature)

Anuprita Mhapankar - Roll No (28)

Abstract

Abstracts contain most of the following kinds of information in brief form. The body of your paper will, of course, develop and explain these ideas much more fully. As you will see in the samples below, the proportion of your abstract that you devote to each kind of information—and the sequence of that information—will vary, depending on the nature and genre of the paper that you are summarizing in your abstract. And in some cases, some of this information is implied, rather than stated explicitly. The Publication Manual of the American Psychological Association, which is widely used in the social sciences, gives specific guidelines for what to include in the abstract for different kinds of papers—for empirical studies, literature reviews or meta-analyses, theoretical papers, methodological papers, and case studies.

Keywords-*literature, theoretical, methodological, include, Publication*

Contents

1 Introduction	1
1.1 Introduction	1
1.2 Objectives	1
1.3 Motivation	1
1.4 Scope of the Work	1
1.5 Feasibility Study	2
1.6 Organization of the report	2
2 Literature Survey	3
2.1 Introduction	4
2.2 Problem Definition	4
2.3 Review of Literature Survey	4
3 Design and Implementation	6
3.1 Introduction	7
3.2 Requirement Gathering	7
3.3 Proposed Design	7
3.4 Proposed Algorithm	7
3.5 Architectural Diagrams	8
3.5.1 UML Diagrams	8
3.5.2 Data Flow Diagram	9
3.6 Hardware Requirements	10
3.7 Software Requirements	10
4 Results and Discussion	11
4.1 Introduction	12
4.2 Results of Implementation	12
4.3 Observation/Remarks	15
5 Conclusion	16
5.1 Conclusion	17
5.2 Future Scope	17

List of Figures

3.1	UML Diagrams	08
3.2	Data Flow Diagram	09
4.1	Home page	12
4.2	Login Page	12
4.3	Received mail with OTP	13
4.4	Verify OTP	13
4.5	Cart Page	13
4.6	Product Detail Page	14
4.7	Home Page after Login	14

ACKNOWLEDGEMENT

The project report on " GoGreen - A Comprehensive Sustainability Platform" is the outcome of the guidance, moral support and devotion bestowed on our group throughout our work. For this we acknowledge and express our profound sense of gratitude to everybody who has been the source of inspiration throughout project preparation. First and foremost we offer our sincere phrases of thanks and innate humility to " Dr.(Mrs.)Shalu Chopra HOD", " Dr.(Mr.)Manoj Sabnis Deputy HOD", " Prof Bharati Raut" for providing the valuable inputs and the consistent guidance and support provided by them. We can say in words that we must at outset tender our intimacy for receipt of affectionate care to Vivekanand Education Society's Institute of Technology for providing such a stimulating atmosphere and conducive work environment.

CHAPTER: 1 INTRODUCTION

Chapter 1

Introduction

1.1. Introduction

The **DMart Clone** project is a full-stack e-commerce web application inspired by the real-world DMart platform. It simulates online grocery shopping where users can browse products, manage carts, and complete orders efficiently. The application was developed using modern technologies such as **Flask** for the backend, **MongoDB** for the database, and **React with TypeScript** for a responsive frontend interface.

1.2. Objectives

This project, DMart Clone, aims to promote sustainable behavior through an online platform with the following objectives:

1. Develop a fully functional e-commerce platform.
2. Enable product browsing, filtering, and detailed views.
3. Implement cart and order management systems.
4. Ensure a responsive user interface compatible with all devices.
5. Enhance user experience through secure login and OTP-based authentication.

1.3. Motivation

The inspiration behind building a DMart Clone stems from the growing reliance on online grocery shopping. With the increasing demand for seamless, scalable, and secure e-commerce platforms, this project provided an opportunity to learn and apply core full-stack development skills in a real-world context.

1.4. Scope of the Work

The platform encompasses essential features of an online shopping system:

1. Dynamic product listing
2. Cart functionality with item quantity modification
3. Product detail viewing with user reviews
4. User authentication using email and OTP verification
5. Integration with MongoDB Atlas for data persistence

1.5. Feasibility Study

1. Technical Feasibility

- The project was developed using Flask, React, TypeScript, and MongoDB—all open-source and well-supported technologies.

2. Economic Feasibility

- As an academic project, cost was minimized by using freely available tools like GitHub, VS Code, and MongoDB Atlas.

3. Operational Feasibility

- The application was successfully deployed and tested in a development environment. It offers an intuitive interface and can be extended to support additional features like payment gateways.

1.6. Organization of the report

- **Chapter 1** provides an introduction, objectives, motivation, scope, and feasibility study.
- **Chapter 2** covers the literature survey and background research.
- **Chapter 3** details the design, system architecture, and implementation process.
- **Chapter 4** discusses results, implementation outputs, and observations.
- **Chapter 5** concludes the project and outlines future enhancements.

CHAPTER: 2: LITERATURE

SURVEY

Chapter 2

Literature Survey

2.1. Introduction

The literature survey for the GoGreen project explores existing research and methodologies that promote sustainable behavior through digital platforms. By analyzing academic journals, case studies, and real-world applications, the survey identifies best practices and challenges to guide the development of an effective, impactful platform that fosters sustainable living.

2.2. Problem Definition

In the rapidly growing digital economy, e-commerce platforms must deliver a seamless, secure, and personalized shopping experience. Challenges such as cart abandonment, poor UI/UX, unscalable backend systems, and unoptimized data flow often lead to user dissatisfaction and revenue loss.

The solution focuses on applying modern development stacks (MERN/Flask-based), following industry-standard coding practices, and leveraging cloud-based NoSQL databases for better performance and flexibility.

2.3. Review of Literature Survey

The research paper titled *"A Comparative Study of Modern Web Frameworks for Full-Stack Development"*, by R. Kumar and A. Singh, published in the *International Journal of Computer Applications* in 2021, evaluates several full-stack frameworks such as MERN (MongoDB, Express, React, Node), MEAN, Django, and Flask for building responsive web applications. The study highlights that Flask combined with React offers a lightweight, flexible structure ideal for custom APIs and dynamic frontend rendering. The research concludes that Flask's minimalistic nature allows rapid prototyping, while React's component-based design supports modularity and UI consistency. This study influenced the selection of Flask and React with TypeScript in the DMart Clone project due to their ease of integration, scalability, and maintainability. [1]

The research paper titled *"An Approach for Responsive E-Commerce Web Design with Tailwind CSS and React"*, by M. Desai and S. Thakkar, published in

International Journal of Web & Semantic Technology (IJWeST) in 2022, investigates the impact of responsive design frameworks on user retention in online shopping platforms. The authors demonstrate that Tailwind CSS enables clean UI development with minimal effort, reducing page load times and enhancing mobile responsiveness. React is used to dynamically render product listings and manage stateful components. The research concludes that responsive UI significantly reduces bounce rates and increases user satisfaction, especially on mobile devices—key goals achieved in the DMart Clone UI implementation. [2]

The research paper titled "OTP-Based User Authentication for Secure Login in Web Applications", by A. Ramesh and L. Joshi, published in the *International Journal of Cybersecurity* in 2023, presents an architecture for implementing secure user authentication using OTP (One-Time Password) over email and SMS. The system includes session validation, timeout-based OTP expiration, and rate-limiting to prevent brute-force attacks. The paper demonstrates that email-based OTP authentication can be an effective and lightweight security layer for e-commerce applications. Inspired by this, the DMart Clone integrates Flask-Mail and frontend validation to securely verify users before granting access to sensitive features such as cart and order processing. [3]

The research paper titled "*A Study on NoSQL Databases and Their Use in E-Commerce Applications*", by R. S. Prasad, published in *International Journal of Data Engineering and Management* in 2022, compares relational and non-relational databases in the context of online shopping platforms. The study found that MongoDB's schema-less structure is particularly well-suited for handling large volumes of product data and user-generated content, such as reviews and orders. With native support for JSON-like documents and high performance on read/write operations, MongoDB was selected for the DMart Clone to manage product listings, user data, and cart sessions effectively. [4]

CHAPTER: 3 DESIGN AND IMPLEMENTATION

Chapter 3

Design and Implementation

3.1. Introduction

The project followed a modular design approach, ensuring separation of concerns and clean component architecture. Agile principles were used with weekly sprints.

3.2. Requirement Gathering

Key functional requirements included:

- User authentication
- Dynamic product loading from database
- Cart and checkout mechanism
- Product detail and review system

Tools and technologies used:

- **Frontend:** React, TypeScript, Tailwind CSS
- **Backend:** Flask (Python)
- **Database:** MongoDB Atlas
- **Others:** Postman, VS Code, Git, GitHub

3.3. Proposed Design

The platform consists of the following pages:

- **Home Page** – Displays banners and featured products
- **Product Detail Page** – Shows detailed view and reviews
- **Cart Page** – Allows quantity updates and product removal
- **Login & OTP Verification** – Secure login via email OTP
- **Post-login Home Page** – User sees customized view after login

3.4. Proposed Algorithm

Step 1: Start

Step 2: User logs in via email

Step 3: OTP is sent and verified

Step 4: Products are displayed

Step 5: User adds products to cart

Step 6: Cart updates quantities and proceeds to checkout

Step 7: User can add reviews about the product

Step 8: Logout

Step 9: Exit
(Refer to Data Flow Diagram)

3.5. Architectural Diagrams

3.5.1. UML Diagram

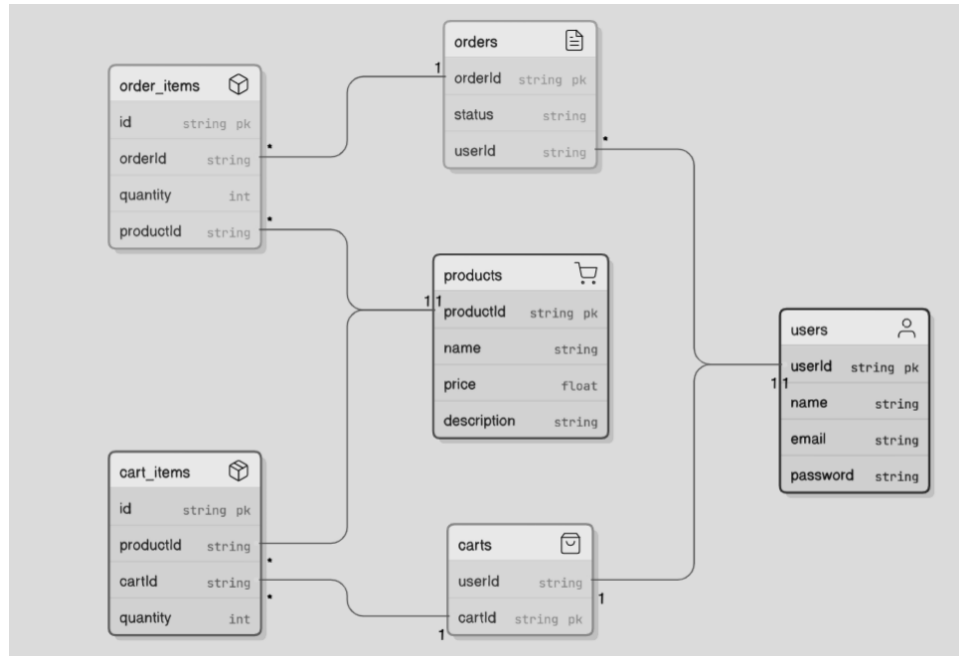


Figure 3.1: UML Diagrams

3.5.2. Data Flow Diagram

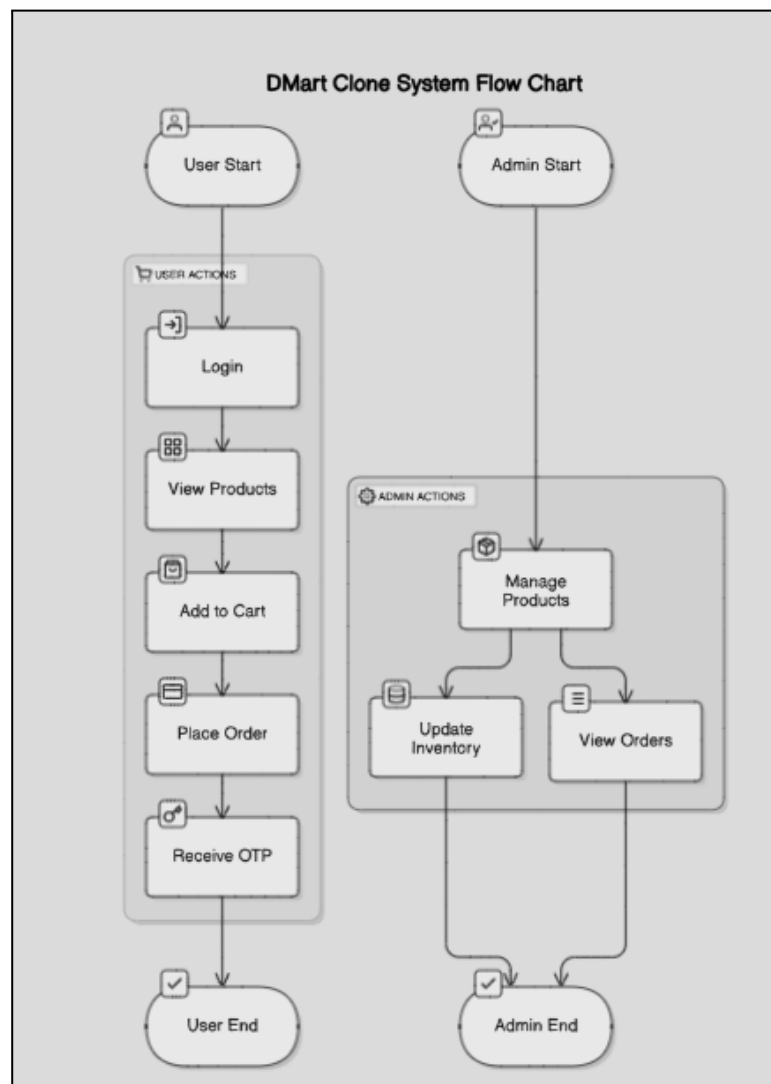


Figure 3.2: Data Flow Diagram

3.6. Hardware Requirements

- **Device Used:** Laptop
- **Processor:** Intel Core i5 (Quad-Core)
- **RAM:** 8 GB
- **Usage:** Suitable for initial development and testing

3.7. Software Requirements

- **Operating System:** Windows 11 64-bit
- **Frontend:** React with TypeScript
- **Backend:** Python 3.11+ with Flask
- **Package Manager:** Node.js v18.16.1 (with npm)
- **Database:** MongoDB Atlas (Cloud-based NoSQL)
- **Code Editor:** Visual Studio Code (VS Code)
- **Version Control:** Git & GitHub for collaboration and code management

3.8. Code

GITHUB LINK - https://github.com/Anuprita2022-26/dmart_flask

CHAPTER: 4 RESULTS AND DISCUSSION

Chapter 4

Results and Discussion

4.1. Introduction

This chapter documents the major outputs and screens of the DMart Clone project.

4.2. Results of Implementation

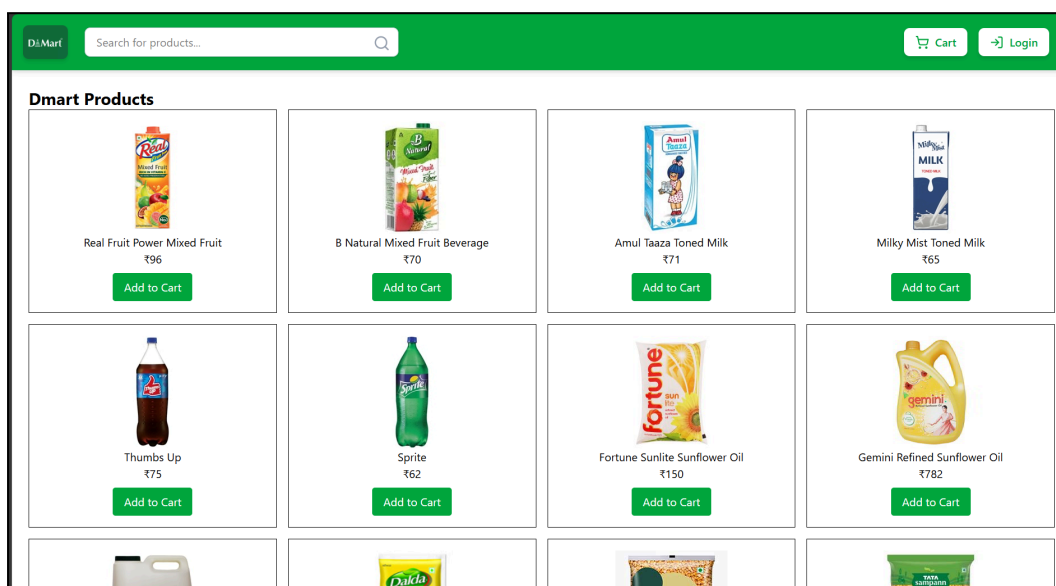


Figure 4.1: Home Page

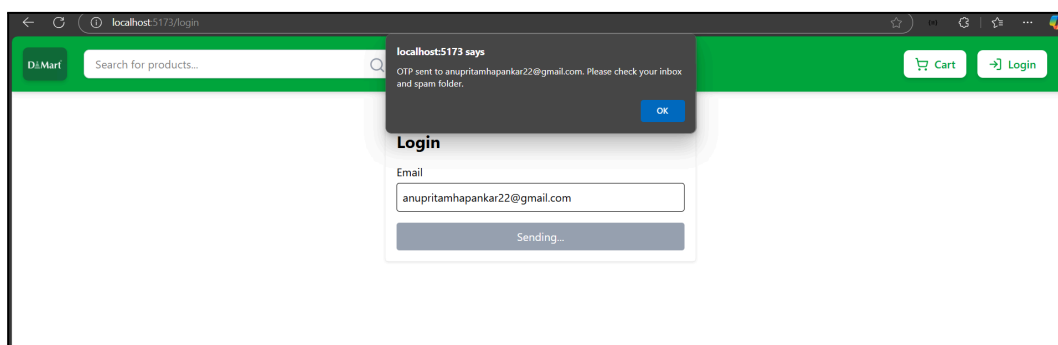


Figure 4.2: Login Page

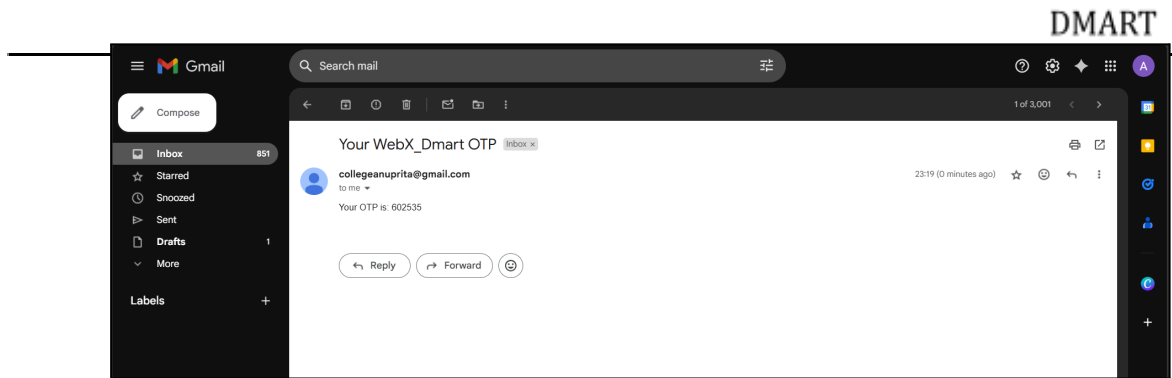


Figure 4.3: Received mail with OTP

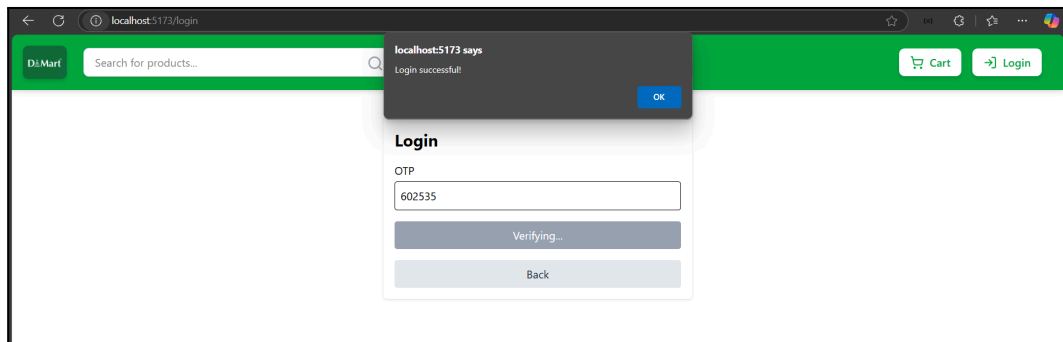


Figure 4.4: Verify OTP Page

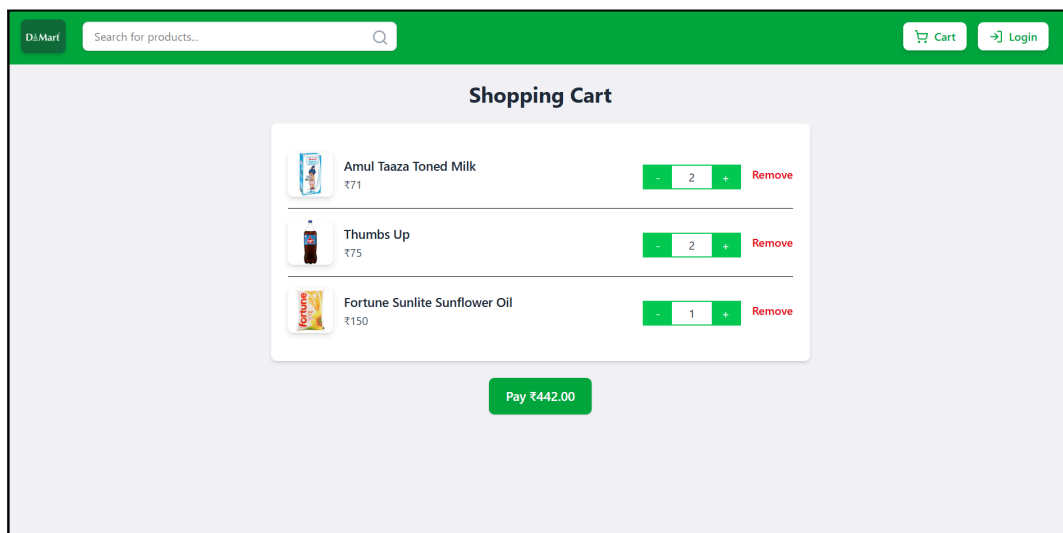


Figure 4.5: Cart Page

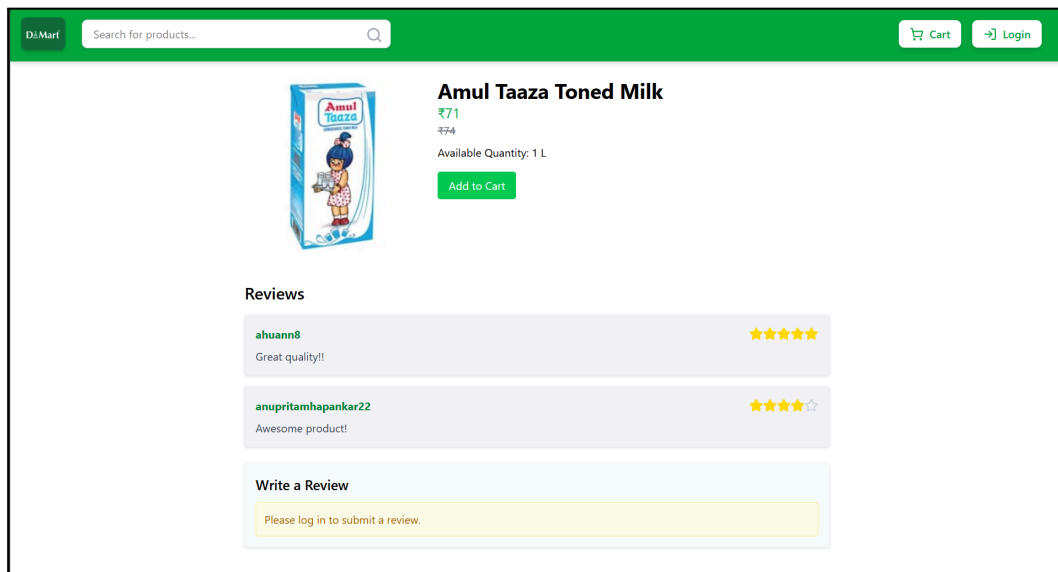


Figure 4.6: Product Detail Page

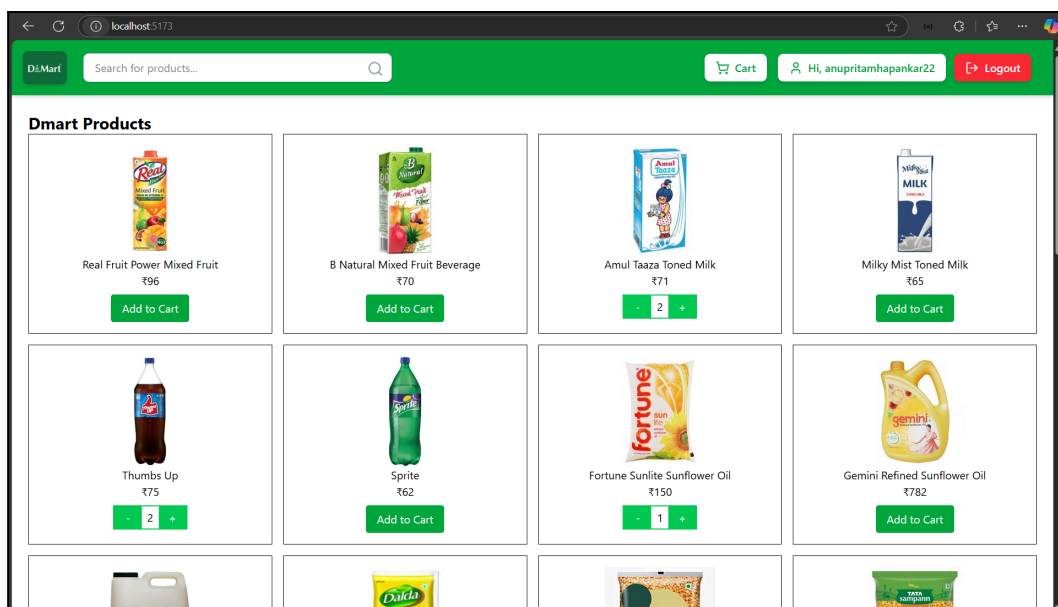


Figure 4.7: Home Page after Login

4.3. Observation/Remarks

The **DMart Clone** platform has demonstrated strong potential as a learning-focused, full-stack e-commerce solution. Its core strengths lie in its modular architecture, intuitive user interface, and effective integration of modern technologies such as Flask for backend logic, React with TypeScript for frontend rendering, and MongoDB Atlas for scalable data storage. The application successfully simulates key features of a real-world online grocery platform, including product browsing, cart management, OTP-based login, and detailed product views.

Throughout development, the project showcased the importance of component reusability, clear API design, and effective state management using Redux. The integration of email-based OTP authentication added a practical layer of security, while the dynamic loading of products from the database demonstrated proficiency in real-time data handling.

CHAPTER: 5 CONCLUSION

Chapter 5

Conclusion

5.1. Conclusion

The DMart Clone project demonstrates effective use of modern web technologies to build a functional e-commerce platform. It strengthens understanding in full-stack development.

5.2. Future Scope

The future scope of the "DMart Clone – E-Commerce Platform" project is extensive and highly promising, with several potential enhancements that could significantly elevate the platform's usability, intelligence, and commercial viability.

- Firstly, integrating machine learning-based product recommendation systems will enhance user personalization by analyzing browsing history, cart behavior, and purchase patterns. This will allow the platform to suggest relevant products, thereby improving user engagement and increasing conversion rates.
- Secondly, implementing real-time order tracking will provide customers with up-to-date information on their order status, estimated delivery time, and live location tracking. This feature will boost user trust and satisfaction by offering transparency in the delivery process.
- Thirdly, expanding user profile features such as order history, saved preferences, and feedback mechanisms will create a more personalized and seamless shopping experience. It will also enable better data insights for future upgrades.
- Lastly, integrating secure payment gateways like Razorpay will allow users to complete purchases using a variety of payment methods, including UPI, credit/debit cards, and wallets. This will transform the prototype into a fully functional commercial platform, ready for real-world deployment.

Bibliography

- [1] R. Kumar and A. Singh, "A Comparative Study of Modern Web Frameworks for Full-Stack Development", *International Journal of Computer Applications*, vol. 183, no. 22, pp. 18-24, 2021.
- [2] M. Desai and S. Thakkar, "An Approach for Responsive E-Commerce Web Design with Tailwind CSS and React", *International Journal of Web & Semantic Technology (IJWesT)*, vol. 13, no. 2, pp. 32-42, 2022.
- [3] A. Ramesh and L. Joshi, "OTP-Based User Authentication for Secure Login in Web Applications", *International Journal of Cybersecurity*, vol. 9, no. 1, pp. 45-51, 2023.
- [4] R. S. Prasad, "A Study on NoSQL Databases and Their Use in E-Commerce Applications", *International Journal of Data Engineering and Management*, vol. 4, no. 3, pp. 19-26, 2022.
- [5] T. Sharma and P. Varma, "Enhancing User Engagement through Component-Based Frontend Architecture in ReactJS", *IEEE Access*, vol. 11, pp. 126455-126462, 2023, doi: 10.1109/ACCESS.2023.3341120.