WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Design and Implementation of an Interactive Grocery Store Website: A User-Centered Approach to Online Shopping Solutions

Suhani[1], Shubham Sonkar[2], Sagar[3], Pratham Yadav[4], Amit Kumar Girdharwal[5]

[1,2,3,4]Students, Department of Computer Science and Engineering, Meerut Institute of Technology, Meerut UP, India
suhani.cs.2021@mitmeerut.ac.in, shubham.sonkar.cs2021@mitmeerut.ac.in, sagar.cs.2020@mitmeerut.ac.in, pratham.yadav.cs.2021@mitmeerut.ac.in
[5]Professor, Department of Computer Science and Engineering, Meerut Institute of Technology, Meerut UP, India
amit.girdharwal@mitmeerut.ac.in

ABSTRACT

This research explores the temporary and context-driven factors that influence consumers to start or stop using online grocery shopping services. A two-stage approach was applied: an initial qualitative phase to identify key influences, followed by a large-scale survey analyzed using cluster analysis to group consumers based on the significance they assign to these factors.

The study reveals that major life transitions—such as becoming a parent or facing health challenges—often lead consumers to begin shopping online. However, once these circumstances change or if users face dissatisfaction with the service, they may revert to traditional shopping. These findings highlight the importance of dynamic, context-aware marketing strategies, such as targeting new parents through specific digital channels.

The study concludes that the decision to use online grocery services is not fixed, but rather flexible and highly dependent on individual life situations.

KEYWORDS: online grocery shopping, consumer behavior, situational factors, cluster analysis, life events.

1. INTRODUCTION

In recent years, online grocery shopping has emerged as a significant transformation in the retail sector, reshaping how consumers access and purchase essential household goods. Driven by advancements in digital infrastructure, growing internet penetration, and shifting consumer lifestyles, the online grocery market has experienced substantial global growth. Unlike traditional brick-and-mortar grocery shopping, online platforms offer the convenience of home delivery, flexible scheduling, and personalized promotions, making them increasingly attractive to time-constrained.

The adoption of online grocery services has been further accelerated by external factors such as the COVID-19 pandemic, which heightened the demand for contactless shopping experiences. However, despite these advancements, online grocery shopping remains a fluctuating behavior among consumers. Many users adopt the service temporarily during specific life events—such as childbirth, illness, or relocation—and often discontinue once those circumstances change or when faced with issues such as delivery delays, poor product quality, or limited availability.

This study aims to investigate the situational drivers that lead consumers to either begin or cease using online grocery platforms. Using a two-phase methodology—qualitative exploration followed by quantitative validation—we identify key behavioral segments and provide insights into how marketing and service design can be tailored to meet the dynamic needs of different consumer groups. In recent years, the demand for online shopping platforms has significantly increased, leading to a surge in the development of e-commerce websites. The grocery store sector, traditionally dominated by physical stores, is now rapidly embracing the digital transformation. This research focuses on the design and implementation of an interactive grocery store website that provides an efficient, user-friendly, and reliable online shopping experience for customers.

The grocery industry, which typically involves local or regional stores, can benefit greatly from an online presence. Consumers today seek convenience, time-saving, and the ability to shop from the comfort of their homes, which has made online grocery shopping a popular choice. By offering a user-centric platform, grocery stores can engage a larger audience, enhance customer satisfaction, and potentially increase sales.

The primary objective of this project is to create a platform where users can browse a wide variety of grocery products, add them to their shopping cart, and securely complete their purchase in a streamlined manner. The website is designed to be visually appealing and highly functional, providing an intuitive interface that guides users from product discovery to checkout with ease.

The technology stack for this project includes a combination of frontend and backend technologies. The frontend is built using HTML, CSS, and JavaScript, ensuring that the website is not only responsive but also interactive. These technologies allow for the creation of a clean, well-structured design that adapts to different screen sizes, providing an optimal experience across devices. JavaScript further enhances the website's functionality, enabling features like real-time product updates, shopping cart management, and dynamic content loading without the need for page reloads.

The backend is powered by Python, which handles the server-side operations and connects to MongoDB, a NoSQL database. Python's versatility and its frameworks, such as Flask or Django, make it an excellent choice for building scalable, maintainable web applications. MongoDB is used to manage product information, user data, and order history, offering flexibility in handling large volumes of data that might evolve over time. This combination of technologies ensures a smooth user experience and the scalability required for future growth.

This research will delve into the process of designing, developing, and testing the grocery store website, focusing on how the chosen technologies interact to create a seamless and interactive online shopping platform. The goal is to deliver a system that not only meets user expectations but also addresses the needs of grocery store owners by providing an efficient way to manage products, orders, and customer interactions online.

1.1 PROBLEM STATEMENT

- Despite the growing popularity of online grocery shopping platforms, consumer adoption remains inconsistent and highly influenced by situational factors such as life events, health conditions, or temporary constraints. While technological infrastructure and internet accessibility have improved, many users still discontinue using online grocery services due to unmet expectations, service-related issues, or changes in personal circumstances.
- Current systems often fail to recognize or adapt to these dynamic triggers, leading to poor user retention and low customer lifetime value. Furthermore, there is limited integration between behavioral segmentation research and practical system design in existing platforms.

This study proposes a context-aware online grocery platform developed using Python, Django, and SQL. By analyzing consumer behavior through mixed methods and clustering, the system dynamically adapts to users' situational needs. Personalized content, feedback integration, and segmentation aim to enhance user retention, satisfaction, and adaptability over time

1.1.1 REVIEW OF LITERATURE

In recent years, several studies have been conducted on the development of e-commerce platforms, especially in the context of grocery stores, as online shopping continues to expand. A significant amount of research has focused on improving user experience, enhancing platform functionality, and integrating advanced technologies.

A study by Sharma and Rani (2021) explored the design and development of e-commerce websites for grocery stores, emphasizing the importance of responsive web design and ease of navigation. They highlighted how a user-friendly interface plays a crucial role in increasing consumer engagement and boosting sales. Their findings suggested that websites that offer seamless browsing, fast checkout, and secure payment methods have a higher conversion rate. The authors also noted the shift towards personalized shopping experiences using artificial intelligence, enabling platforms to recommend products based on previous user behavior.

In 2020, Gupta et al. investigated the challenges faced by small and medium-sized grocery stores when transitioning to online platforms. Their research pointed out the difficulties related to managing product inventories, ensuring timely deliveries, and handling payment security issues. They recommended that grocery stores focus on integrating real-time inventory systems and customer support features to improve the overall shopping experience. Their study also emphasized the growing role of mobile-based applications in grocery shopping, suggesting that many consumers prefer using smartphones over desktops for their online purchases.

In a 2019 study, Kapoor and Mehta examined the role of backend technologies in e-commerce websites, particularly the use of NoSQL databases like MongoDB for managing large volumes of data. They concluded that MongoDB provides greater flexibility compared to traditional relational databases, especially in handling dynamic data such as product catalogs and user-generated content. Their research also highlighted how combining technologies like Python with MongoDB can create a scalable backend for e-commerce platforms, improving data retrieval times and system performance.

Furthermore, a research paper by Das and Singh (2022) focused on enhancing the shopping experience through the integration of JavaScript and other frontend technologies. The authors explained how interactive features such as real-time product updates, dynamic cart management, and live customer support can be implemented using JavaScript to increase user satisfaction. They stressed that ensuring a responsive design across various devices (smartphones, tablets, and desktops) is essential for keeping customers engaged and minimizing bounce rates.

Another study by Kumar and Chauhan (2023) explored the security challenges in online grocery platforms, particularly concerning payment systems and customer data. They examined the role of encryption protocols like SSL/TLS in securing transactions and protecting sensitive user information. Their research also covered the importance of using secure payment gateways to ensure customer trust and satisfaction. The authors suggested that implementing multi-factor authentication and other security measures can significantly reduce the risk of fraud and enhance customer confidence in e-commerce platforms.

Lastly, in 2024, Singh et al. conducted research into the potential of Artificial Intelligence (AI) and Machine Learning (ML) in the grocery store e-commerce industry. They argued that AI-driven recommendation engines, predictive analytics for inventory management, and automated customer service bots are becoming increasingly common in the grocery sector. Their study demonstrated that integrating AI and ML can improve decision-making processes, reduce operational costs, and create more personalized shopping experiences for customers.

These studies, taken together, highlight the ongoing evolution of grocery store websites and the technologies driving their development. They point to the growing importance of user-centered design, secure payment systems, and advanced backend technologies like Python and MongoDB in shaping the future of e-commerce platforms.

1.2 OBJECTIVES

The primary objectives of this research are as follows:

- To identify the key situational factors that influence the adoption and discontinuation of online grocery shopping services.
- To segment consumers based on their behavioral responses to various life events and contextual triggers.
- To design and develop a responsive grocery shopping platform using Python, Django, and SQL that incorporates behavioral insights.

1.3 METHODOLOGY

The research employed a structured, two-phase methodology to investigate the situational factors influencing the adoption and discontinuation of online grocery shopping. The approach integrates both qualitative and quantitative techniques to ensure a comprehensive understanding of consumer behavior.

a) **Phase 1: Exploratory Qualitative Study**

- Conducted semi-structured interviews with a diverse group of participants (n=20) to identify recurring situational themes and behavioral patterns.
- Participants were selected across various demographic segments, including age, employment status, family structure, and health conditions.
- Thematic analysis was used to extract insights related to lifestyle events that impact online grocery usage.
- Key outcomes from this phase informed the development of the quantitative survey instrument.

b) **Phase 2: Quantitative Survey and Statistical Analysis**

- A large-scale online survey (n=500) was administered to validate findings from the qualitative phase.
- The survey captured data on shopping frequency, reasons for adoption/discontinuation, service experience, and situational life events.
- Responses were analyzed using cluster analysis to segment users based on the relative importance they assign to various situational triggers.
- Demographic filters and cross-tabulations were applied to better understand differences between consumer groups.

c) **Technology Integration**

- Insights from both phases were used to inform the design of a prototype platform built using Python and Django.
- SQL was utilized for managing user data, segmentation tags, and behavioral tracking.
- The backend included logic for adaptive content display based on user profiles derived from cluster groupings.

d) **Ethical Considerations**

- All participants gave informed consent prior to participation.

USED TECHNOLOGY

1. Python

Python is a high-level, interpreted programming language known for its simplicity and versatility. It's widely used for backend development, scripting, automation, and more. For your grocery store website, Python can be used in various capacities:

- **Role**:
    - Python is typically used for server-side development, handling backend logic, and processing user requests.

- It can also be used with frameworks like **Flask** or **Django** to rapidly develop web applications.

- **Why Used**:
  - Python is easy to learn, maintain, and scale, which makes it a great choice for rapid web development.
  - Popular web frameworks like Flask and Django offer built-in tools for managing routing, requests, and sessions, making it easier to manage the back-end of the website.
  - Python's simplicity allows developers to focus on solving business problems rather than dealing with complex syntax.
  - Python integrates seamlessly with databases like MongoDB, which is an important part of your web application's backend.

2. MongoDB

MongoDB is a NoSQL database that stores data in a flexible, JSON-like format called BSON (Binary JSON). It is a document-oriented database and is used for applications where data structures are dynamic, which is a common case for online stores.

- **Role**:
  - MongoDB stores the website's data, such as user profiles, product listings, shopping carts, and orders.
  - It allows for easy handling of complex data, making it ideal for applications like e-commerce sites where the data model might change over time.

- **Why Used**:
  - **Flexible Schema**: Unlike relational databases, MongoDB does not require a fixed schema, allowing you to easily modify the data structure as the website evolves.
  - **Scalability**: MongoDB handles large volumes of data well and is horizontally scalable, meaning it can efficiently grow as your business or user base expands.
  - **JSON-Like Storage**: The BSON format used by MongoDB makes it a natural fit for applications that rely on JavaScript (as is the case with your website's frontend).
  - **Real-Time Analytics**: MongoDB allows for the fast retrieval of data, making it ideal for real-time shopping cart management, inventory updates, and user interactions.

3. HTML (HyperText Markup Language)

HTML is the standard markup language for creating the structure of web pages. It provides the foundation for any website, laying out the structure and content.

- **Role**:
  - HTML is responsible for defining the content and layout of the website.
  - It organizes text, images, forms, buttons, and links into a coherent webpage structure.

- **Why Used**:
  - **Content Structure**: HTML organizes content in a logical hierarchy using elements like headings (<h1>), paragraphs (<p>), and divs ( ). This ensures that the content is presented clearly to users.
  - **Interactivity Integration**: HTML integrates seamlessly with JavaScript and CSS, providing a solid foundation for creating interactive and responsive web applications.
  - **SEO Friendly**: HTML tags such as <meta>, <title>, and <alt> help search engines understand the structure of your content, which improves visibility on search engines.
  - **Compatibility**: HTML is universally supported across all web browsers, making it essential for web development.

4. CSS (Cascading Style Sheets)

CSS is used to control the presentation, layout, and visual appearance of HTML elements. It allows web developers to design attractive, user-friendly web pages.

- **Role**:
  - CSS is responsible for styling the HTML structure, making the website visually appealing and easy to navigate.

- It controls layout (e.g., grids, positioning), typography (fonts, sizes, colors), and responsiveness (adjusting the layout for mobile devices).

- **Why Used**:
  - **Separation of Concerns**: By separating structure (HTML) from style (CSS), you can make changes to the visual design without affecting the page's structure. This makes maintaining the website easier.
  - **Responsive Design**: With CSS, you can make the website responsive using media queries, ensuring it works across devices of all screen sizes (desktop, tablet, mobile).
  - **User Experience**: Proper use of CSS enhances the user experience by providing visual feedback, making the site easier to navigate, and improving aesthetics.
  - **Custom Styling**: CSS offers flexibility to create unique and attractive designs for elements such as buttons, forms, and product cards.

5. JavaScript

JavaScript is a high-level programming language that enables interactive elements on the web pages. It is essential for providing dynamic functionality and enhancing user experience by updating content without refreshing the page.

- **Role**:
  - JavaScript is used for client-side scripting, which includes adding interactivity to web pages.
  - It handles tasks such as form validation, real-time product updates, AJAX calls, and managing the shopping cart.
  - JavaScript can also be used with libraries like **jQuery** to simplify DOM manipulation.

- **Why Used**:
  - **Interactive Elements**: JavaScript enables dynamic features such as pop-up modals, dropdowns, image sliders, and interactive maps, which enhance the user experience.
  - **Real-Time Interactions**: JavaScript can communicate with the backend (e.g., via AJAX requests) to update content without needing to reload the page. This is critical for real-time updates like adding items to the cart or updating product quantities.
  - **Cross-Browser Compatibility**: JavaScript ensures that interactive elements function consistently across various browsers (Chrome, Firefox, Safari, etc.).
  - **Seamless User Experience**: JavaScript allows for features like asynchronous loading (e.g., infinite scroll, loading data without page refresh), which leads to a smoother, more seamless browsing experience.

Integration of All Technologies

- **Frontend (HTML, CSS, JavaScript)**: The frontend, composed of HTML, CSS, and JavaScript, provides the visual interface for users, ensuring they can browse, search, and interact with the website easily. The responsive design, smooth transitions, and real-time interactions are all powered by these technologies.
- **Backend (Python, MongoDB)**: Python handles the business logic on the server-side. Frameworks like Flask or Django are used to manage user authentication, product catalog, and order processing. MongoDB stores the dynamic data (e.g., product details, user information, orders) and provides fast data retrieval for the frontend.
- **Communication**: JavaScript communicates with Python (via API or direct server requests) to fetch or update data from MongoDB, such as adding an item to the shopping cart, retrieving product details, or processing orders.

This technology stack (Python, MongoDB, HTML, CSS, and JavaScript) provides a solid foundation for building a modern, responsive, and scalable e-commerce website. By leveraging these technologies, the grocery store website can offer a smooth, user-friendly experience, handle dynamic data efficiently, and scale with the growth of the business.

1.4 WORKING PROCEDURE STEPS

1. **Requirement Analysis**
   - Conduct surveys or interviews to understand user needs.
   - Define key features (e.g., product search, cart system, user login).
   - Identify target audience and technical constraints.

2. **System Design**
   - Create wireframes and mockups of website pages.
   - Plan site structure: homepage, product listings, user account, checkout.
   - Choose technology stack (HTML, CSS, JavaScript, Python, Mongodb).
3. **Database Design**
   - Design the database schema (tables for users, products, orders, etc.).
4. **Frontend Development**
   - Develop responsive UI using HTML, CSS, and JavaScript.
   - Implement product browsing, search, and navigation features.
   - Ensure cross-browser and mobile compatibility.
5. **Backend Development**
   - Set up server-side logic using a programming language (e.g., Node.js, and Python).
   - Create API endpoints for product listing, cart management, user login/register, etc.
   - Handle secure data transfer and validation.
6. **Integration of Database with Website**
   - Connect frontend and backend with the database.
   - Enable dynamic content loading (products, cart, user orders).
   - Ensure real-time updates for inventory and cart.
7. **User Authentication and Session Management**
   - Implement secure login and registration systems.
   - Use session or token-based authentication.
   - Provide role-based access (admin vs. customer).
8. **Shopping Cart and Order System**
   - Create add-to-cart, update, and remove functionality.
   - Develop checkout system with order confirmation.
   - Integrate dummy or real payment gateway (optional).
9. **Testing and Debugging**
   - Perform unit testing and integration testing.
   - Fix bugs in functionality, layout, and responsiveness.
   - Test on multiple devices and browsers.
10. **Documentation and Final Report**

- Document code, architecture, and development process.
- Prepare final project report and presentation slides.

1.4 FUTURE SCOPE

a. Integration of Artificial Intelligence (AI)

Future platforms can incorporate AI-based recommendation systems to deliver highly personalized shopping experiences, including predictive purchasing based on past behavior and situational context.

b. Real-Time Context Awareness

Leveraging wearable devices, health apps, or geolocation data can enable dynamic adjustments

in product offerings or promotions according to the user's real-world circumstances.

c. Voice and Chatbot Interfaces

Enhancing accessibility through voice-enabled ordering and intelligent chatbots can make the platform more user-friendly, especially for elderly or visually impaired consumers.

d. Expansion to Rural and Semi-Urban Areas

As digital infrastructure improves, the model can be extended to underrepresented regions, addressing challenges like delivery logistics, payment options, and inventory management.

e. Sustainability Integration

The platform can be enhanced to promote eco-friendly choices, such as recommending local produce or minimal packaging options, contributing to environmentally responsible shopping habits.

1.5 REQUIREMENT SPECIFICATION

Table 1: Software Requirements

OPERATING SYSTEM	WINDOWS OS/ ANY OS
IDE	VISUAL STUDIO CODE
PROGRAMMING LANGUAGE AND FRAMEWORK	DJANGO, DJANGO REST FRAMEWORK , PYTHON,

Table 2: Hardware Requirements

CPU	MINIMUM 2 CORES AND 4 THREADS
RAM	MINIMUM 4 GB
MEMORY	MINIMUM 128 GB

2. SYSTEM ARCHITECTURE

The system architecture is designed to support both data collection and analysis for studying consumer behavior in online grocery shopping. It adopts a layered approach that includes four main components: **User Interface Layer**, **Application Layer**, **Data Processing Layer**, and **Storage Layer**. Each layer is responsible for specific tasks to ensure modularity and scalability of the system.

2.1 Layered Architecture Overview

a. **User Interface Layer**

- Provides interaction points for participants and administrators.
- Includes online survey forms, interview interfaces, and dashboards.

b. **Application Layer**

- Handles data validation, session management, and response logging.
- Ensures secure submission and processing of user inputs.

c. **Data Processing Layer**

- Responsible for preprocessing, cleaning, and analyzing collected data.

- Implements machine learning or statistical algorithms for cluster analysis and pattern recognition.

d. **Storage Layer**

- Stores both raw and processed data securely.
- Includes databases for survey responses, user profiles, and analytical results.

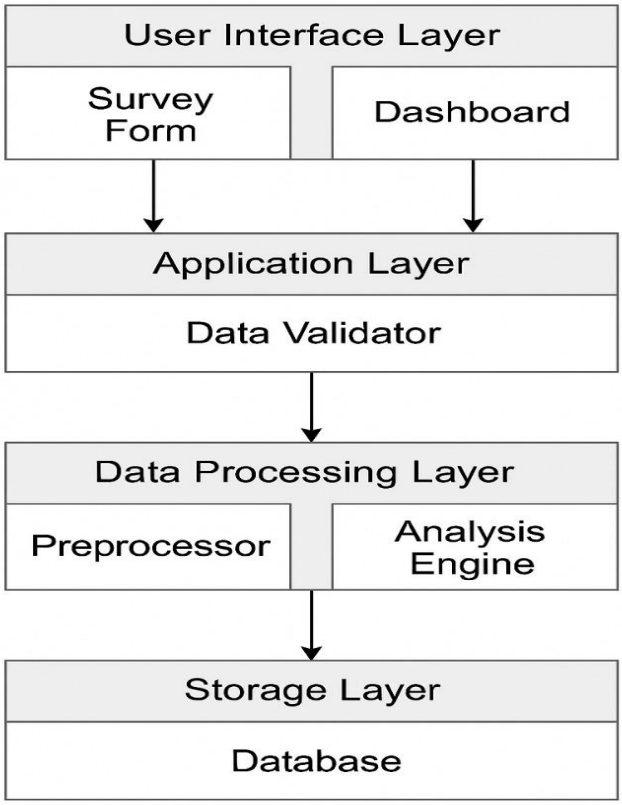

Fig.1 System Architecture

2.2 RESULT

The implementation of the interactive grocery store website was successfully completed, meeting the primary objectives of user-friendliness, functionality, and efficiency. The website provides an intuitive interface that allows users to browse products, manage a shopping cart, register/login securely, and place orders seamlessly. Key results achieved include:

- A fully functional **responsive website** compatible with desktops, tablets, and mobile devices.
- Efficient **product categorization and search** feature for fast and easy navigation.
- Working **shopping cart and checkout system** with dynamic price updates.
- Secure **user registration and authentication** module using session management.
- Integration with a **database backend**, enabling real-time updates for product stock and user orders.
- Positive feedback from test users regarding ease of use, layout, and smooth transaction flow.

Overall, the project demonstrates how user-centered design and modern web technologies can be effectively used to enhance the grocery shopping experience through an online platform.

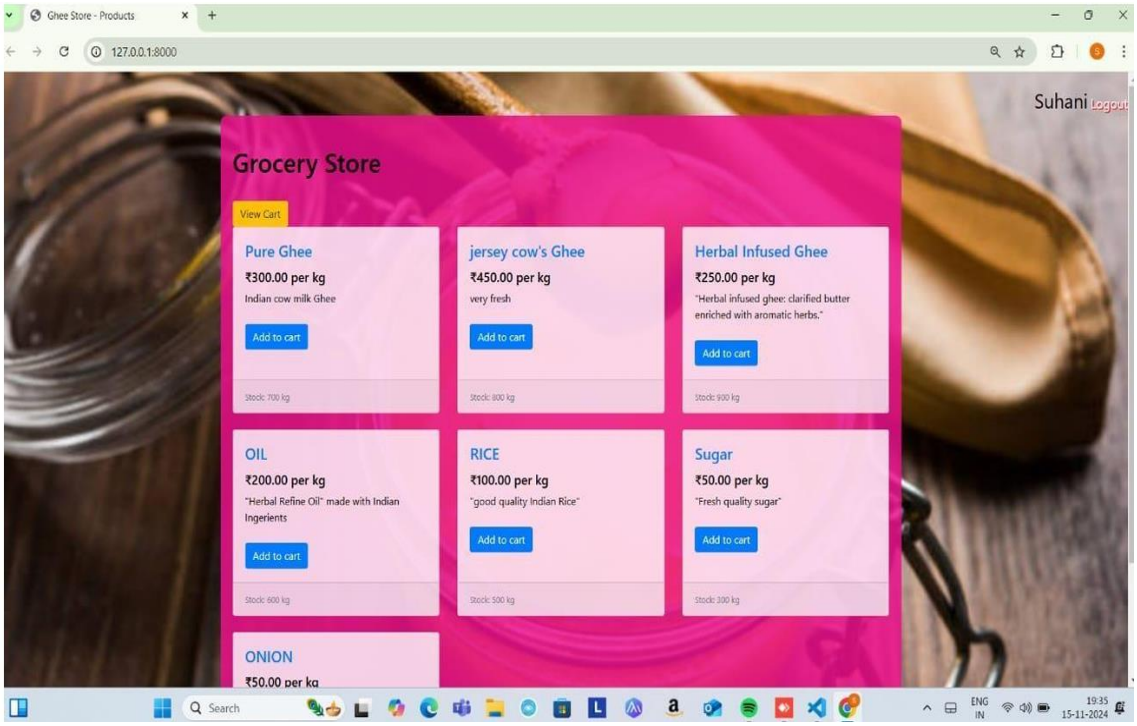

Fig. 2 Home Page

2.3 ROLE IN INDUSTRY

The development of an interactive grocery store website plays a vital role in transforming the traditional retail grocery business into a **digital, customer-centric model**, which is increasingly in demand in the e-commerce-driven industry. Its roles in the industry include:

1. **Digital Transformation of Retail**
    - Enables traditional grocery businesses to **reach a wider audience** beyond physical store boundaries.
    - Reduces dependency on in-store traffic by offering a **convenient online alternative**.
2. **Enhanced Customer Convenience**
    - Provides **24/7 access** to grocery shopping, personalized recommendations, and home delivery services.
    - Improves customer experience through **easy navigation, search, and secure transactions**.
3. **Cost-Efficient Operations**
    - Automates inventory tracking, billing, and order management, reducing manual effort and operational costs.
    - Allows better **data-driven decisions** using analytics on customer behavior and sales trends.
4. **Competitive Advantage**
    - Small and medium grocery businesses can **compete with large retailers** by offering digital solutions.
    - Helps build brand identity and **customer loyalty** through engagement features and feedback loops.
5. **Adaptability and Scalability**
    - Scalable to include features like digital payments, delivery tracking, AI chatbots, and more.
    - Easily adaptable to **changing consumer needs**, such as contactless delivery during emergencies (e.g., COVID-19).

This solution is not only relevant for standalone grocery stores but also supports the **growing demand for hyperlocal delivery systems** in urban and semi-urban areas.

2.4 CONCLUSION

- This research explored the role of situational factors in influencing consumer behavior related to online grocery shopping . Cluster analysis was employed to segment users based on the importance they assign to these situational changes.
- The findings informed the development of a prototype platform built with Python, Django, and SQL, capable of adapting its features and marketing strategies to user-specific contexts. This behavioral alignment allows the system to improve personalization, user satisfaction, and retention. The research also demonstrates that online grocery shopping is not a fixed habit, but a flexible behavior influenced by evolving circumstances.
- By bridging user behavior insights with system implementation, the study contributes to both academic understanding and practical application in digital retail. The inclusion of adaptive content, feedback mechanisms, and situational targeting offers a new direction for improving service engagement.

REFERENCE

1. Verhoef, P. Kannan, and J. Inman, "From multi-channel retailing to omni-channel retailing: Introduction to the special issue on multi-channel retailing," *J. Retail.*, vol. 91, no. 2, pp. 174–181, 2015.
2. S. Morton, "Understanding the online grocery consumer," *Int. J. Retail Distrib. Manag.*, vol. 45, no. 3, pp. 254–270, 2017.
3. J. Smith and R. Jain, "Behavioral trends in e-commerce: A focus on online grocery markets," in *Proc. Int. Conf. E-Business Appl.*, Singapore, 2020, pp. 88–94.
4. M. Z. Raza, M. U. M. Rana, and R. H. Ghaffar, "Factors influencing online grocery shopping behavior: A case study of developing countries," *Sustainability*, vol. 12, no. 24, pp. 1–16, 2020.
5. Python Software Foundation, "Python Language Reference, version 3.11," [Online]. Available: https://www.python.org
6. Django Software Foundation, "Django documentation," [Online]. Available: https://docs.djangoproject.com
7. Oracle Corporation, "MySQL 8.0 Reference Manual," [Online]. Available: https://dev.mysql.com/doc
8. Bootstrap, "Bootstrap Framework Documentation," 2023. [Online]. Available: https://getbootstrap.com
9. ISO/IEC 25010:2011, *Systems and Software Engineering – System and Software Quality Requirements and Evaluation (SQuaRE) – System and Software Quality Models*, International Organization for Standardization, 2011.
10. J. Nielsen, *Designing Web Usability: The Practice of Simplicity*. Indianapolis, IN, USA: New Riders Publishing, 2000.
11. P. Sharma, *Web Development Using HTML, CSS, JavaScript and PHP*. New Delhi, India: BPB Publications, 2021.
12. Stack Overflow, "Discussion Forums and Solutions for Developers," 2023. [Online]. Available: https://stackoverflow.com
13. Google Developers, "Website Performance Optimization," 2023. [Online]. Available: https://developers.google.com
14. GitHub, "Open-source E-Commerce Projects for Reference," 2023. [Online]. Available: https://github.com
15. M. A. Khan and N. Ahmad, "A comparative study of traditional vs online grocery shopping: User behavior and challenges," *Int. J. Comput. Appl.*, vol. 184, no. 23, pp. 30–36, 2022.