



TECHMART: A Data-Driven E-Commerce Platform Using Django for Customer Repurchase Prediction Assistance

Ayush Yadav¹, Manoj Kumar Chauhan², Arman Ansari³, Saif Ansari⁴, Arpita Singh⁵

^{1,2,3,4}Scholar, Department of Computer Science & Engineering, Goel Institute of Technology & Management, Lucknow, Uttar Pradesh, India

⁵Assistant Professor, Department of Computer Science & Engineering, Goel Institute of Technology & Management, Lucknow, Uttar Pradesh, India

KEYWORD

Django, Python, E-commerce, Customer Retention, RFM Model, Repurchase Prediction.

ABSTRACT

E-commerce platforms today are no longer just digital marketplaces—they are intelligent systems that continuously learn from user behavior to improve customer experience and drive long-term engagement. In this context, data-driven strategies have become essential for understanding customer preferences, predicting future actions, and ultimately increasing customer retention. This paper presents TechMart, a Django-based e-commerce system designed to analyse user behaviour and predict repurchase probability. The system utilizes Python for backend processing and integrates behavioural datasets including purchase frequency, recency, and monetary value (RFM model). Graphical analysis is performed to identify patterns in customer behaviour. The platform demonstrates how combining web development with data analytics can significantly improve business decision-making and user experience.

I. INTRODUCTION

E-commerce platforms play a crucial role in the modern digital economy by providing users with convenient access to products and services. With the rapid growth of online shopping, it has become essential for platforms to not only provide basic functionalities but also analyse user behaviour to improve engagement and retention. TechMart is designed as a **data-driven e-commerce platform** using Django and Python. Unlike traditional systems, it integrates behavioural analysis techniques to predict customer repurchase probability. The system focuses on scalability, security, and user experience while incorporating analytical insights for better decision-making.

1.1 Project Description

TechMart is a web-based e-commerce application that allows users to:

- Register and log in securely
- Browse and search products
- Add items to cart
- Place orders

The system also includes:

Corresponding Author: Ayush Yadav, Scholar, Department of Computer Science & Engineering, Goel Institute of Technology & Management, Lucknow, Uttar Pradesh, India

Email: ayush.y2966@gmail.com

- Admin panel for product and order management
- Behavioural tracking module for analytics

The application follows Django's **Model-Template-View (MTV)** architecture and uses SQLite for data storage.

1.2 Motivation

- To gain practical experience in Django-based full-stack development
- To understand how user behaviour can improve business decisions
- To implement data analytics within a real-world application

II. RELATED WORK

Paper [1] discusses the use of Django for developing scalable web applications, highlighting its built-in security and ORM capabilities.

Paper [2] explores customer retention techniques using RFM analysis in e-commerce systems.

Paper [3] focuses on behavioural analytics and how user interaction data can improve recommendation systems.

Paper [4] explains secure authentication mechanisms in web applications using Django's authentication framework.

Paper [5] studies data-driven decision-making in online shopping platforms.

III. METHODOLOGY

A. Overall Architecture

TechMart follows a three-tier architecture:

- Presentation Layer (Templates/UI)
- Application Layer (Django Views)
- Data Layer (Database)

B. Frontend Design

- Developed using HTML, CSS, JavaScript
- Django Templates for dynamic rendering
- Responsive design for better UX

C. Backend Implementation

- Django framework handles business logic
- Django ORM for database operations
- Secure authentication using Django auth system

D. Database Management

- Stores users, products, orders, and behavioural data
- Efficient querying using ORM

E. Behavioural Analysis Module

This is the **core innovation** of TechMart.

It tracks:

- Purchase frequency
- Product views
- Cart activity
- Last purchase time

F. Repurchase Probability Model

$$P(R) = w_1R + w_2F + w_3M$$

- R = Recency
- F = Frequency
- M = Monetary value

H. Implementation Flow

User accesses TechMart platform System loads frontend interface User interacts (browse, cart, purchase) Backend processes requests Behavioural data is stored Probability score is calculated Personalized insights are generated.

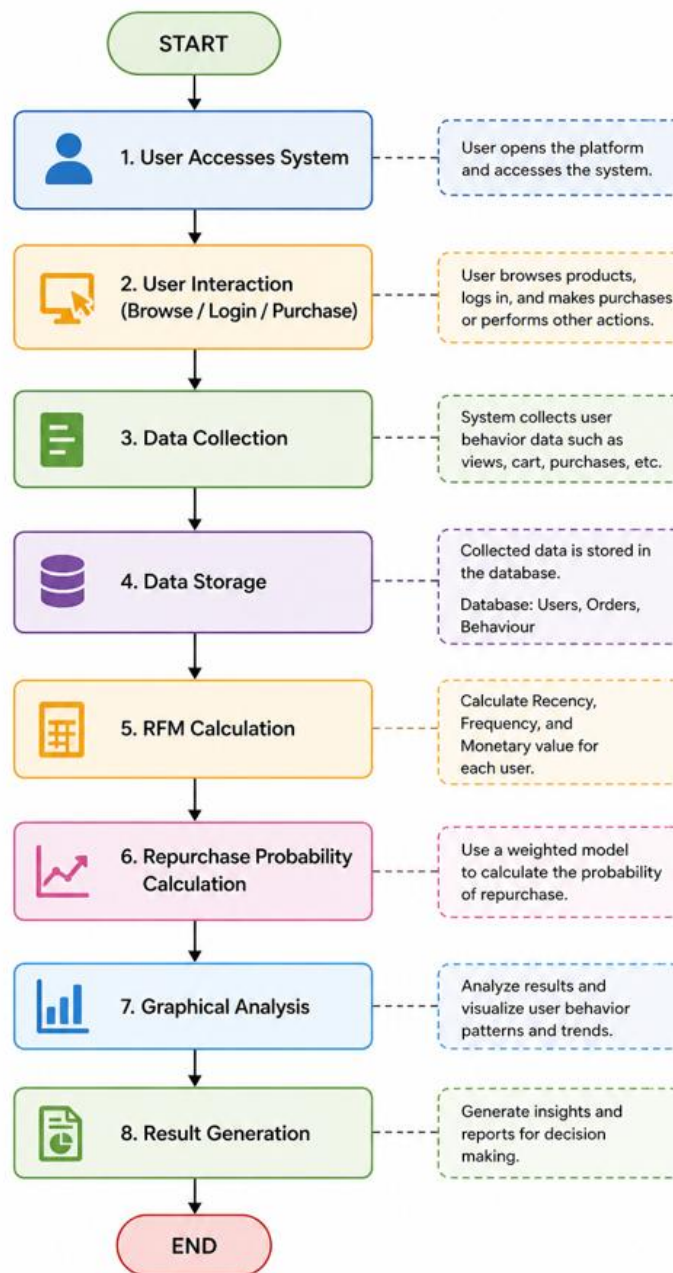


Fig 1: Flow Chart of a Methodology

IV. System Design and Implementation Framework

This section presents the overall system design, architectural framework, and implementation strategy adopted for the TechMart e-commerce platform. The system is developed using the Django framework and follows a modular, scalable, and data-driven design approach.

A. System Architecture and Design Approach

TechMart is built using Django's Model-Template-View (MTV) architecture, which ensures clear separation of concerns and efficient management of application components. Model Layer: Handles database schema and data operations using Django ORM Template Layer: Responsible for rendering dynamic user interfaces View Layer: Processes business logic and handles client requests The system follows a client-server architecture, where the

client (web browser) communicates with the Django server through HTTP requests. The server processes these requests, interacts with the database, and returns appropriate responses.

Key architectural characteristics:

- Modular and reusable components
- Scalable backend structure
- Secure request handling

Integration-ready for analytics modules

B. System Workflow

The workflow of TechMart is designed to ensure smooth interaction between users and system components.

User Interaction Phase

Users access the platform through a web browser Browse products, search items, and view details

1. Authentication Phase
 - Users register or log in using secure credentials
 - Django authentication system validates user identity
2. Transaction Phase
 - Users add products to cart
 - Proceed to checkout and confirm orders
3. Data Processing Phase
 - System stores transaction data
 - Behavioural data is recorded (views, purchases, activity)
4. Analytics Phase
 - RFM model calculates repurchase probability
 - Results used for insights and future recommendations

C. Module Design

The system is divided into multiple functional modules:

1. User Management Module
 - Handles registration and login
 - Manages user sessions
 - Stores user profile data
2. Product Management Module
 - Displays product listings
 - Supports filtering and searching
 - Managed by admin panel
3. Cart and Order Module
 - Handles cart operations
 - Processes orders and transactions
 - Stores order history
4. Behavioural Analysis Module
 - Tracks user activity
 - Calculates RFM values
 - Generates repurchase probability
5. Admin Module
 - Manage products and inventory
 - Monitor user activity
 - View analytics and reports

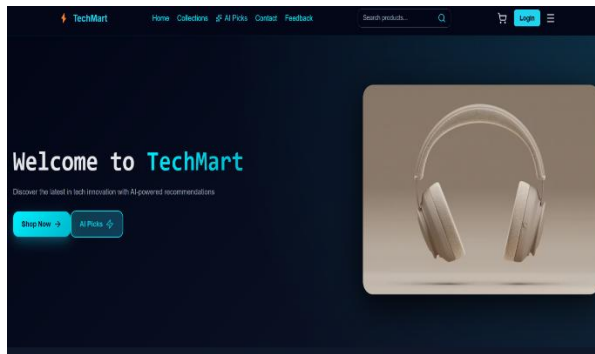


Fig.2.Home Page

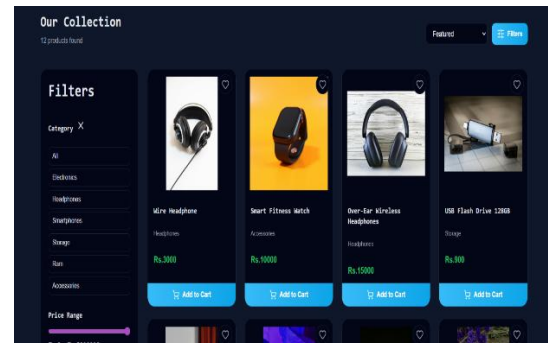


Fig.3.Product Page

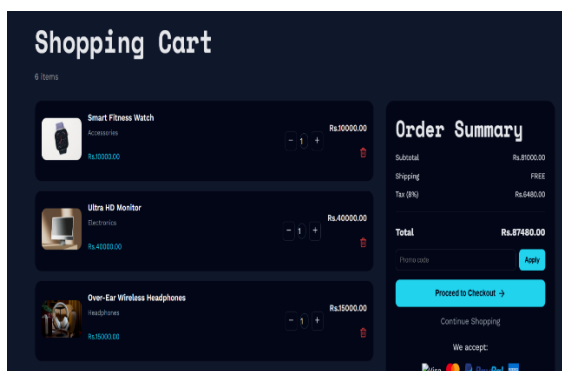


Fig.4.Cart Page

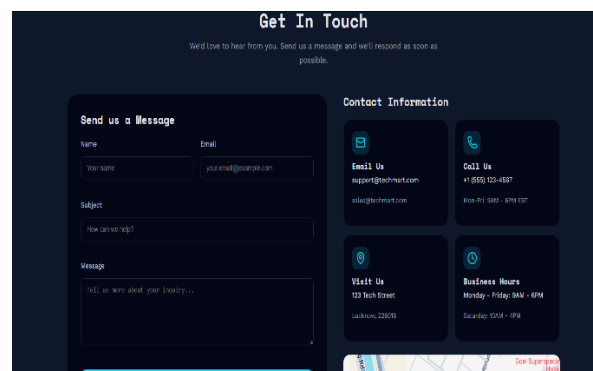


Fig.5.Contact Page

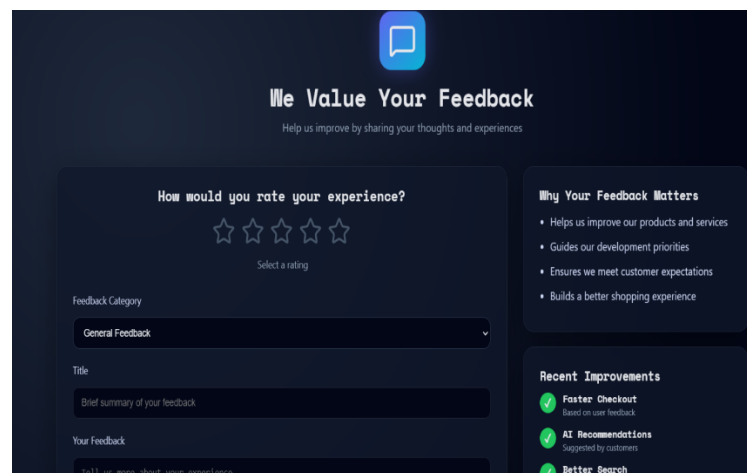


Fig.6.Feedback Page

V. Results and Discussion

The implementation and evaluation of the TechMart e-commerce platform demonstrate that the system effectively combines standard online shopping functionalities with data-driven behavioral analysis. The application was tested under normal operating conditions, including user registration, authentication, product browsing, cart management, and order processing. The results confirm that the system performs reliably, providing a smooth and secure user experience. Django's built-in authentication system ensures secure access control, while the admin panel allows efficient management of products, users, and orders. A key contribution of TechMart is the integration of behavioural analytics using the RFM (Recency, Frequency, Monetary) model to estimate repurchase

probability. The analysis shows that users who have made recent purchases are more likely to return, indicating that recency is a strong predictor of customer retention. Similarly, users with higher purchase frequency demonstrate increased engagement and a greater likelihood of repeat transactions. Monetary analysis further reveals that customers with higher total spending tend to exhibit stronger loyalty, contributing significantly to long-term business value.

VI. Conclusion

This paper presented *TechMart*, a Django-based e-commerce platform enhanced with data-driven techniques to analyse customer behaviour and predict repurchase probability. Unlike conventional e-commerce systems that primarily focus on transaction processing, TechMart integrates behavioural analytics using the RFM (Recency, Frequency, Monetary) model to generate meaningful insights about user engagement and purchasing patterns. The system successfully demonstrates how a web application developed using Python and Django can be extended beyond basic functionality to include analytical capabilities. By capturing user interactions such as product views, purchase history, and cart activity, the platform is able to estimate the likelihood of repeat purchases. This enables the identification of high-value customers and supports the development of targeted retention strategies.

VII. Future Work

Although TechMart achieves its primary objectives, several enhancements can be implemented to further improve the system's functionality, intelligence, and scalability. One of the most significant improvements involves integrating advanced machine learning algorithms such as Logistic Regression, Decision Trees, or Random Forest models to predict repurchase probability with higher accuracy. These models can be trained on historical user data to provide more precise and dynamic predictions compared to the current weighted scoring approach. Another potential enhancement is the implementation of a personalized recommendation system. By analysing user preferences and purchase history, the system can suggest relevant products using techniques such as collaborative filtering or content-based filtering, thereby improving user engagement and increasing sales.

References

- [1] Stuart Russell and Peter Norvig, *Artificial Intelligence: A Modern Approach*, 4th ed. Pearson, 2020.
- [2] Tom M. Mitchell, *Machine Learning*. New York, NY, USA: McGraw-Hill, 1997.
- [3] Ian Goodfellow, Yoshua Bengio, and Aaron Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
- [4] F. F. Reichheld, "The one number you need to grow," *Harvard Business Review*, vol. 81, no. 12, pp. 46–54, 2003.
- [5] W. J. Reinartz and V. Kumar, "On the profitability of long-life customers in a noncontractual setting: An empirical investigation," *Journal of Marketing*, vol. 64, no. 4, pp. 17–35, 2000.
- [6] Django Software Foundation, "Django Web Framework Documentation," 2024. [Online]. Available: <https://docs.djangoproject.com/>
- [7] Python Software Foundation, "Python Documentation," 2024. [Online]. Available: <https://docs.python.org/>
- [8] Customer Relationship Management and RFM-based analysis: J. Bult and T. Wansbeek, "Optimal selection for direct mail," *Marketing Science*, vol. 14, no. 4, pp. 378–394, 1995.