CONSUMER BEHAVIOURAL ANALYSIS USING ADVANCED CLUSTERING TECHNIQUES

Supriya Thati[1], Kaja Masthan[2], Pravalika Konduru[3], Aishwarya Jakkidi[4] , Bhanu Prasad Karvanga[5] , Lasya Priya K[6]
[1] *Student, Sphoorthy Engineering College, Telangana, India*
[2] *Assistant Professor, Sphoorthy Engineering College, Telangana, India*
[3] *Student, Sphoorthy Engineering College, Telangana, India*
[4] *Student, Sphoorthy Engineering College, Telangana, India*
[5] *Student, Sphoorthy Engineering College, Telangana, India*
[6] *Student, Sphoorthy Engineering College, Telangana, India*

ABSTRACT

The rapid proliferation of digital transactions has generated vast volumes of customer data, creating an urgent need for intelligent and scalable segmentation tools that convert raw information into actionable business intelligence. This paper presents a comprehensive, web-based Customer Behavioural Analysis platform that integrates three complementary unsupervised machine learning algorithms-K-Means, MeanShift, and DBSCAN within a unified, user-friendly interface. The system is built using the Streamlit framework and supports secure multi-user authentication backed by MongoDB, complete activity logging, bulk dataset processing, real-time individual customer profiling, interactive visualizations, personalized discount recommendations, and automated email reporting.

The proposed platform adopts the Recency-Frequency-Monetary (RFM) behavioural scoring model as its analytical foundation, further enriched by Annual Income as an additional feature. A weighted composite scoring scheme is applied: Recency (weight 0.2, inverted), Frequency (weight 0.4), and Monetary (weight 0.4) form the RFM Score, which is then weighted at 0.8, while Annual Income contributes at 0.2. The resulting features are standardized using StandardScaler before being fed into the clustering models. K-Means (K=5) provides fixed-group segmentation for structured strategic planning; MeanShift automatically discovers the optimal number of clusters without manual parameter specification, achieving a Silhouette Score of 0.534 and Davies-Bouldin Index of 0.584; and DBSCAN excels at detecting outliers and irregular behavioural patterns, identifying 6 noise points and 6 meaningful clusters.

A central launcher application orchestrates the three independent dashboards on dedicated local ports, enabling business owners to compare and select the most appropriate segmentation approach for their data. Model quality is evaluated using Silhouette Score (K-Means: 0.564, MeanShift: 0.534) and Davies-Bouldin Index (K-Means: 0.557, MeanShift: 0.584), confirming well-separated, interpretable customer segments. All ten functional test cases passed without exceptions. The system successfully bridges advanced machine learning theory with practical business needs, making enterprise-grade customer analytics accessible to non-technical stakeholders in retail, e-commerce, banking, and service sectors.

Keywords: *Customer Segmentation, K-Means Clustering, MeanShift, DBSCAN, RFM Analysis, Unsupervised Machine Learning, Streamlit, Business Intelligence, Behavioural Analytics, MongoDB*

1. INTRODUCTION

The rapid growth of digital transactions has generated vast amounts of customer data, making it essential for businesses to understand consumer behaviour for effective decision-making. Traditional rule-based and manual segmentation methods are time-consuming, subjective, and fail to reveal complex behavioural patterns hidden in large datasets. Machine learning-based unsupervised clustering techniques have emerged as a powerful alternative, enabling automatic discovery of natural customer groupings without requiring labelled data.

This project, titled *Consumer Behavioural Analysis using Advanced Clustering Techniques*, presents a comprehensive web-based customer segmentation dashboard that integrates three powerful unsupervised algorithms—K-Means, MeanShift, and DBSCAN—into a unified, production-ready platform. Built using Streamlit, the system supports

secure MongoDB-backed authentication, bulk dataset processing, individual customer profiling, interactive visualizations, personalized discount recommendations, and automated email delivery of insights [1]. By combining RFM (Recency, Frequency, Monetary) analysis with Annual Income as an enhanced feature set, the system delivers actionable customer segments that drive targeted marketing strategies.

MeanShift stands out for its ability to automatically determine the optimal number of clusters without manual specification, offering greater flexibility in real-world scenarios where the ideal group count is unknown [2].

1.1 Features and Applications

The platform delivers multi-algorithm support through three independent yet integrated modules—Fixed Groups (K-Means), Automatic Grouping (MeanShift), and Outlier Detection (DBSCAN)—accessible via a single launcher. Key capabilities include secure MongoDB-based authentication with activity logging, bulk CSV analysis with automated clustering and visualization, real-time individual customer profiling with spending score calculation, personalized discount recommendations, one-click email report delivery, and pre-trained model persistence for fast inference.

The system is applicable across multiple industries: in Retail and E-commerce for targeted promotions, in Banking and Finance for premium client identification, in Telecommunications for customized subscriber plans, and in Hospitality for personalizing loyalty programs based on spending behaviour [3].

1.2 Relevance to AI-ML

This project exemplifies the practical application of unsupervised machine learning in real-world business intelligence. It demonstrates core AI-ML concepts including data preprocessing (feature weighting and standardization), clustering algorithms, model evaluation using Silhouette Score and Davies-Bouldin Index, and deployment of ML models as interactive web applications. By comparing K-Means, MeanShift, and DBSCAN side-by-side, the system highlights the strengths and trade-offs of different clustering approaches, bridging academic theory with industry needs [4].

2. PROBLEM DEFINITION

2.1 Problem Statement

In today's competitive business environment, organizations generate enormous volumes of customer transactional data but often lack effective mechanisms to transform this raw information into meaningful customer segments. Traditional manual or rule-based segmentation approaches are time-consuming, subjective, and fail to capture complex behavioural patterns. Existing tools either require predefined cluster counts, overlook natural data groupings, or cannot reliably identify anomalous customers, resulting in missed marketing opportunities, inefficient resource allocation, and suboptimal customer engagement strategies.

2.2 Existing Solutions and Limitations

Several commercial and open-source solutions exist for customer segmentation, including basic CRM modules, spreadsheet-based RFM analysis, and standalone Python scripts using single clustering algorithms. Popular platforms such as Tableau, Power BI, and certain SaaS tools offer visualization capabilities but typically rely on K-Means with manual parameter tuning or provide limited algorithmic flexibility. Most solutions suffer from a lack of integrated multi-algorithm support, absence of secure multi-user authentication with activity logging, no seamless transition between bulk processing and individual customer profiling, and limited automation for personalized discount recommendations or email campaigns.

2.3 Proposed Solution

This project addresses these challenges by developing a secure, web-based customer segmentation dashboard that integrates three complementary unsupervised machine learning algorithms through a central launcher application. K-Means handles scenarios where the business requires a fixed number of segments; MeanShift dynamically discovers the optimal cluster count without prior specification; and DBSCAN effectively identifies noise points and unusual customer behaviours that do not fit into any regular group. The platform features MongoDB-backed authentication, pre-trained models, interactive visualizations, discount recommendations, email integration, and a complete activity history audit trail.

3. LITERATURE SURVEY

Customer behavioural analysis has evolved significantly since the introduction of the RFM framework by Hughes [5] in 1994. The major area relevant to this project is unsupervised machine learning applied to RFM-enhanced customer segmentation. Table 1 summarizes the chronological evolution of key contributions in this domain.

Table 1: Literature Survey — Evolution of Customer Segmentation Techniques

Year	Author(s)	Core Method	Key Contribution
1994	Hughes [5]	Classic RFM	Introduced RFM as the gold standard behavioural scoring framework
2008	Chan [6]	RFM + Value Pyramid	Proved behavioural segmentation superior to demographics
2018	Aryuni et al. [7]	K-Means vs K-Medoids	Confirmed K-Means superiority in speed and accuracy on RFM data
2018	Wei et al. [2]	MeanShift on weighted RFM	First demonstration of fully automatic cluster discovery without K
2020	Hosseini & Shalmani [8]	DBSCAN on transactional data	Excellent outlier detection and handling of irregular clusters
2021	Cheng & Chen [9]	RFM + Annual Income + weighted scoring	Showed Annual Income dramatically improves segment separation
2022	Gankidi et al. [10]	K-Means + Elbow + Streamlit	First web-based interactive prototype (single algorithm)
2025	Present Work	K-Means + MeanShift + DBSCAN + Weighted RFM+Income + Streamlit + MongoDB + Email	First full production-ready system with three algorithms, automatic bandwidth, AI naming, discount suggestions, individual & bulk analysis, and secure multi-user environment

Most prior works exhibit limitations including single-algorithm focus, manual parameter dependency, limited deployment readiness, absence of multi-feature integration, and lack of end-to-end business workflow including email automation and individual customer profiling [10]. The present work overcomes all these limitations by delivering the first full production-ready multi-algorithm platform.

4. SYSTEM DESIGN

4.1 System Architecture

The system follows a modular, three-layer architecture designed for flexibility, maintainability, and ease of deployment:

- **Presentation Layer**: Built using Streamlit, providing an interactive, responsive web interface with cards, forms, visualizations, and navigation.

- **Business Logic Layer**: Contains the core ML models (K-Means, MeanShift, DBSCAN), RFM scoring, feature weighting, discount recommendation engine, and Gemini API integration for intelligent cluster naming.
- **Data Layer**: MongoDB handles user authentication, password hashing (SHA-256), and activity logging. Pre-trained models and scalers are persisted as pickle files.

A central launcher app.py acts as the entry point, launching the three independent dashboards on dedicated ports: 8601 (K-Means), 8602 (MeanShift), and 8603 (DBSCAN). This design ensures algorithm isolation while sharing common authentication and logging components.

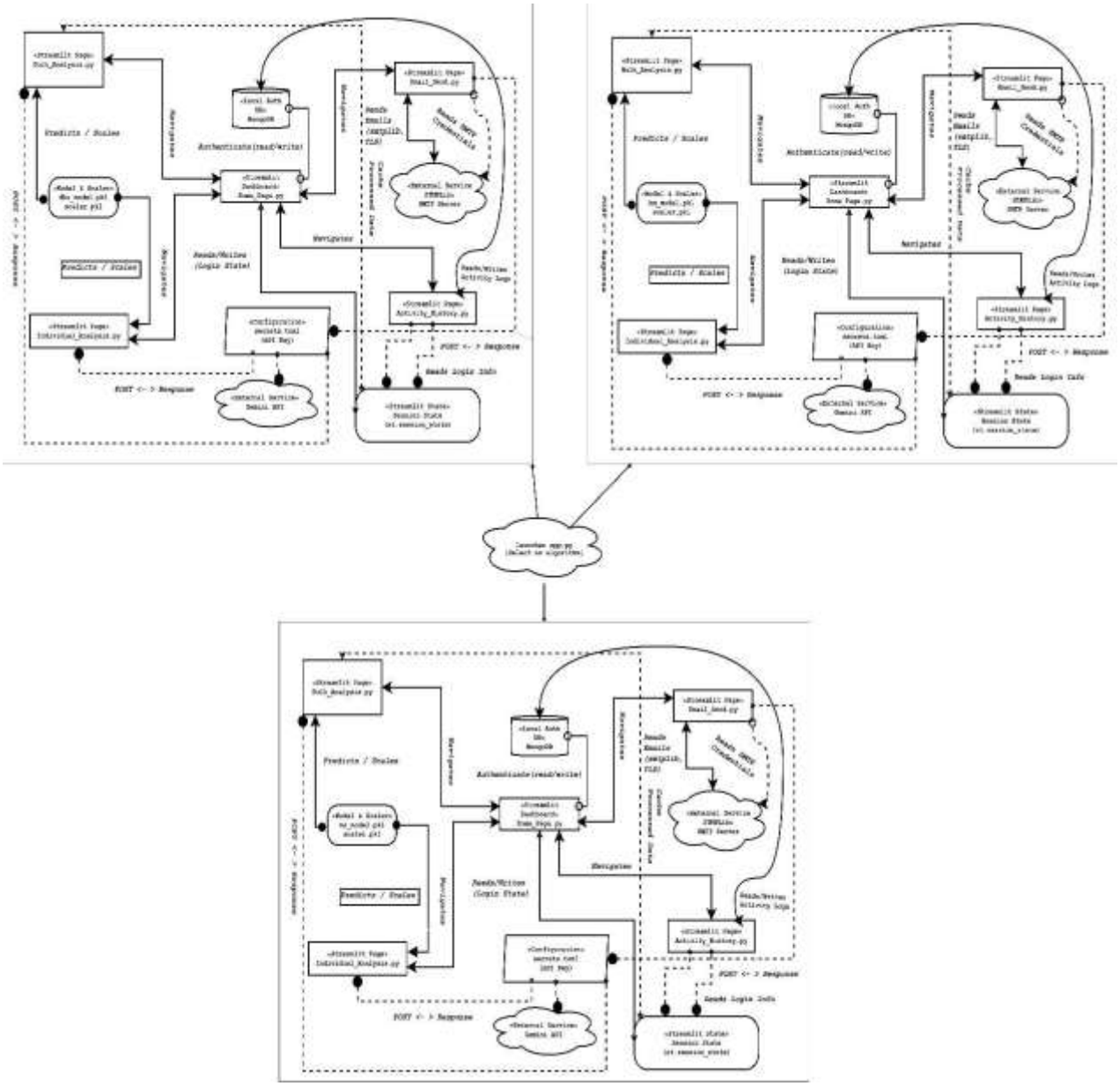

Fig. 1: System Architecture Diagram

4.2 UML Diagrams

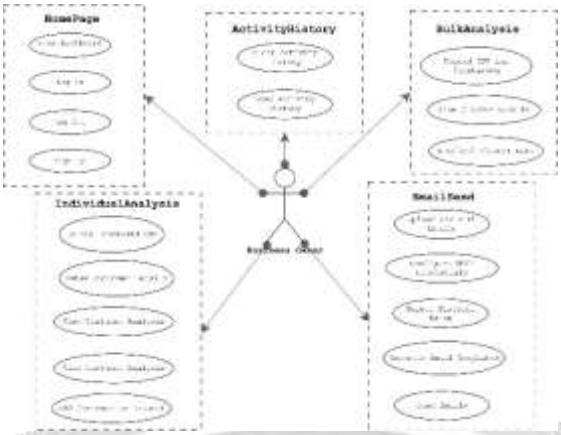

Fig. 2: Use Case Diagram

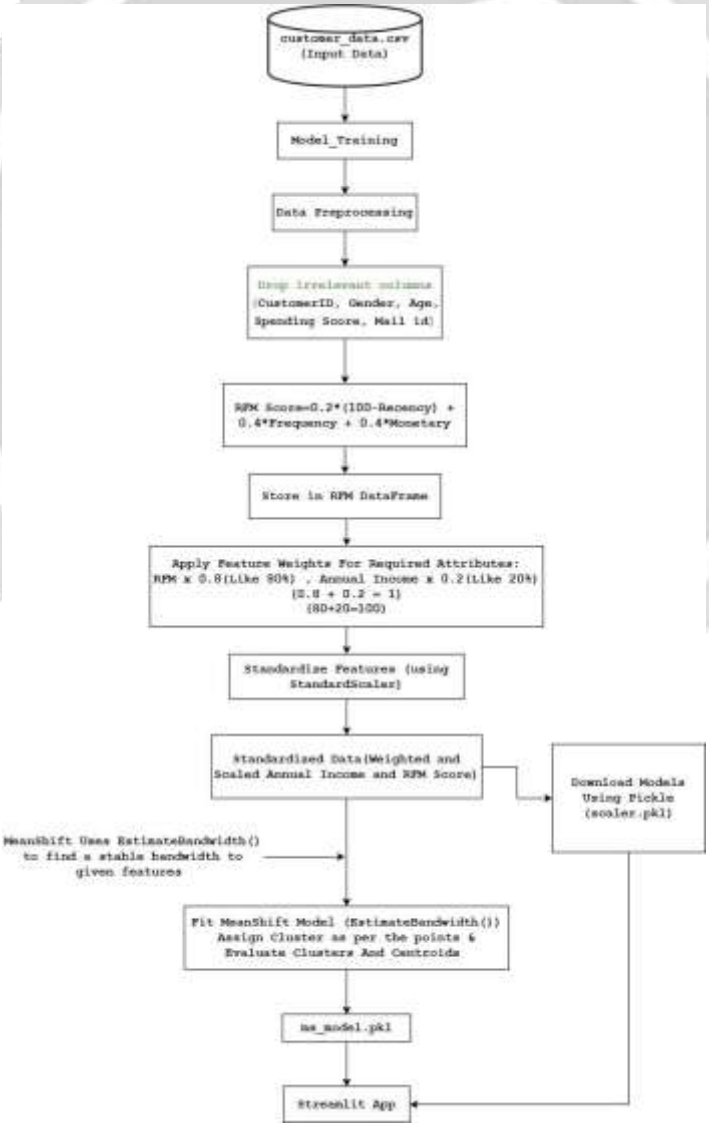

Fig. 3: Model Training Data Flow Diagram

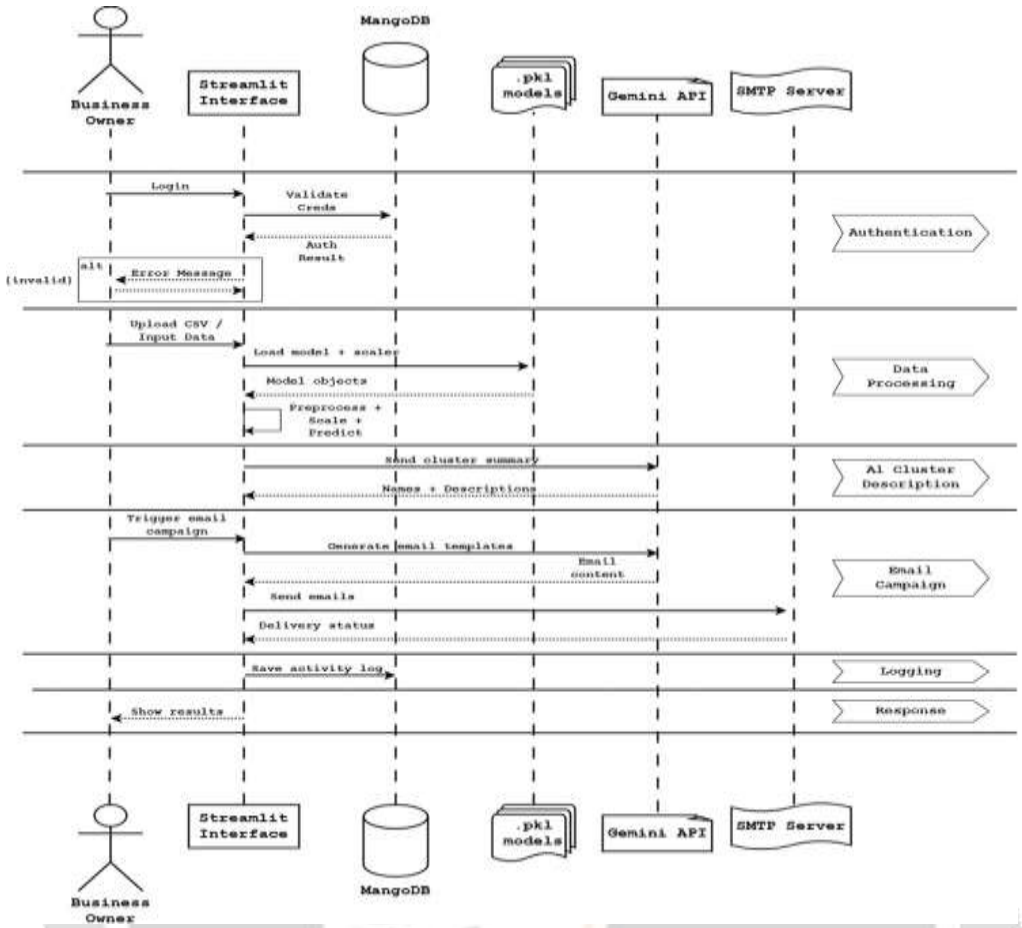

Fig. 4: Sequence Diagram

4.3 System Flow

The system flow is linear and intuitive: the user launches launcher_app.py and selects a clustering method. The corresponding Streamlit app opens in the browser. After MongoDB-based authentication, the user chooses between Bulk Analysis (CSV upload), Individual Analysis (single customer input), Email Send, or Activity History. All actions are automatically logged in MongoDB with timestamps.

5. METHODOLOGY

5.1 Algorithms Used

The system implements a two-stage feature engineering process before applying clustering. The foundation is the RFM scoring model, which quantifies customer value across three dimensions: Recency (days since last purchase), Frequency (total number of purchases), and Monetary (total amount spent).

The composite RFM Score is computed as:

$$RFM\ Score = (0.2 \times (100 - Recency)) + (0.4 \times Frequency) + (0.4 \times Monetary) \quad (1)$$

For clustering, two weighted features are derived:

$$Weighted\ RFM = RFM\ Score \times 0.8 \quad (2)$$

$$Weighted\ Income = Annual\ Income\ (k\$) \times 0.2 \quad (3)$$

These features are standardized using StandardScaler before being passed to the clustering algorithms.

K-Means Clustering:

K-Means is a centroid-based partitioning algorithm configured with $K = 5$ in this implementation. It minimizes the Within-Cluster Sum of Squares (WCSS):

$$\text{WCSS} = \sum_{i=1}^{K} \sum_{x \in C_i} \|x - \mu_i\|^2 \quad (4)$$

where μ_i is the centroid of cluster C_i. The optimal $K = 5$ was validated using the Elbow method during model training. K-Means achieved a **Silhouette Score of 0.564** and **Davies-Bouldin Index of 0.557**, indicating well-separated, compact clusters.

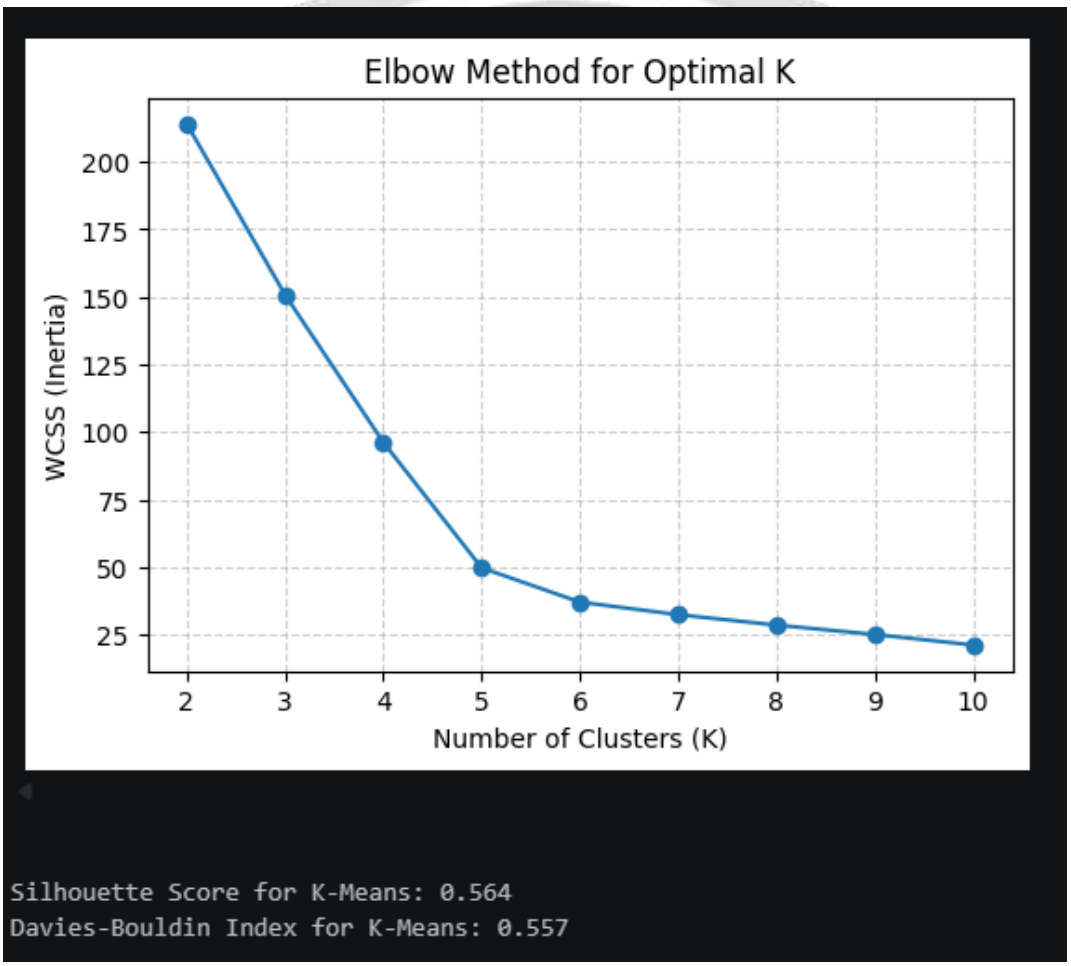

Fig. 5: K-Means Model Training and Evaluation

MeanShift Clustering:

MeanShift is a density-based, non-parametric algorithm that automatically discovers the optimal number of clusters. It estimates an optimal bandwidth parameter (0.877 in this project using estimate bandwidth with quantile=0.15) and iteratively shifts each data point toward the region of highest density within the bandwidth radius. Points converg- ing to the same mode are grouped into the same cluster. MeanShift requires no predefined K, making it highly flexible for real-world data. It achieved a **Silhouette Score of 0.534** and **Davies-Bouldin Index of 0.584**.

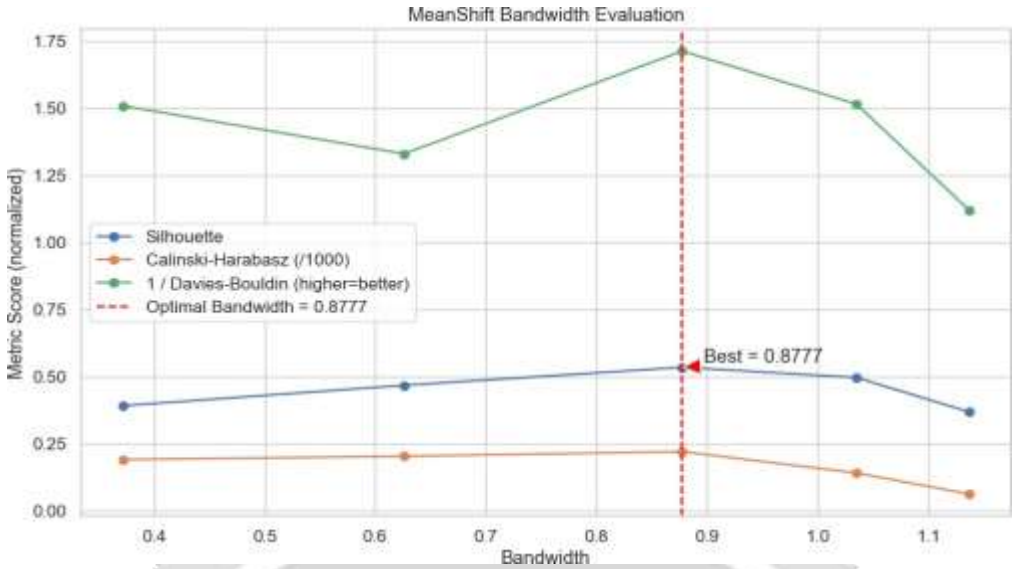

Fig. 6: MeanShift Clustering and Evaluation

DBSCAN (Density-Based Spatial Clustering of Applications with Noise):

DBSCAN identifies clusters of arbitrary shape while explicitly labelling outliers as noise (cluster label $= -1$). A point is a *core point* if it has at least min-samples=5 neighbours within distance $\varepsilon = 0.45$. The optimal eps was determined using a 5-Nearest Neighbors distance plot. DBSCAN correctly identified **6 noise points** and formed **6 meaningful clusters** in this project, making it ideal for outlier detection.

```
========== DBSCAN EVALUATION SUMMARY ==========
                 Metric  Value Quality
0      Silhouette Score  0.392    Good
1  Davies-Bouldin Index  0.577    Good
2           Noise Ratio  0.000    Good
```

Fig. 7: DBSCAN Clustering and Evaluation

5.2 Model Training Process

All three models were trained using separate Jupyter notebooks (Model Training.ipynb) following an identical pipeline: dataset loading and preprocessing (dropping CustomerID, Gender, Age, Spending Score, Mail ID), RFM score computation, feature weighting, StandardScaler fitting, algorithm-specific training (Elbow for K-Means, estimate bandwidth for MeanShift, k-NN distance for DBSCAN), model evaluation, and persistence of both the trained model and fitted scaler as pickle files in the model files/ directory. This ensures consistency across all three algorithms during both training and runtime inference.

5.3 Tools and Libraries

The project is developed entirely in Python 3 using: **Streamlit** for the interactive web interface, **scikit-learn** for K-Means, MeanShift, DBSCAN, StandardScaler, and evaluation metrics, **pandas** and **NumPy** for data preprocessing and

RFM computation, **Matplotlib** and **Seaborn** for visualizations, **PyMongo** for MongoDB authentication and logging, **pickle** and **hashlib** for model persistence and password hashing, and **asyncio** and **requests** for optional Gemini API integration for dynamic cluster naming.

6. IMPLEMENTATION

6.1 System Modules

The system is organized into a root launcher and three independent algorithm-specific module sets. Each module contains:

- **Home_Page.py**: Login/signup, MongoDB verification, activity logging, and dashboard navigation.
- **Bulk_Analysis.py**: CSV upload, pre-trained model clustering, scatter plot + centroid visualization, cluster summaries, discount recommendations, and CSV export.
- **Individual_Analysis.py**: Single-customer RFM input, real-time cluster prediction, Spending Score calculation, similar-customer lookup, discount recommendation, and cluster plot with the customer's position marked.
- **Email_Send.py**: Automated personalized report delivery via SMTP to selected dataset customers.
- **Activity_History.py**: Timestamped user action log from MongoDB with clear option.

6.2 Application Outputs

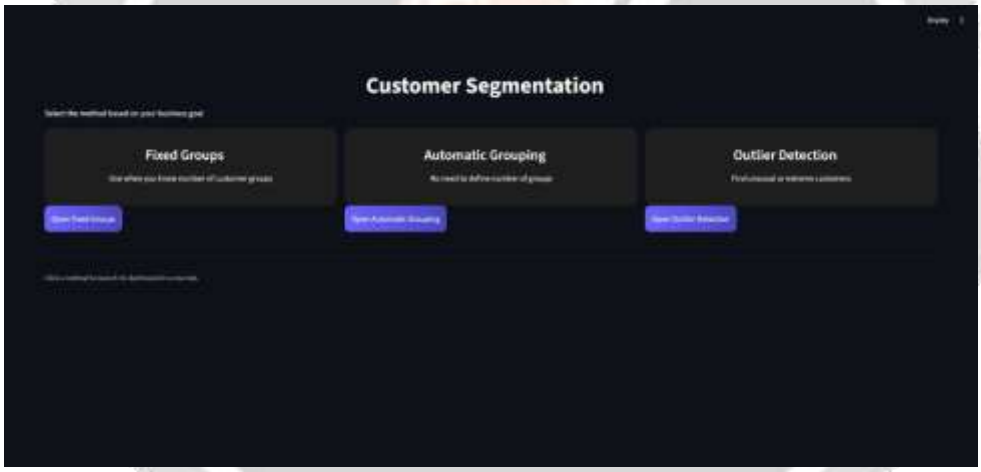

Fig. 8: Main Launcher Page

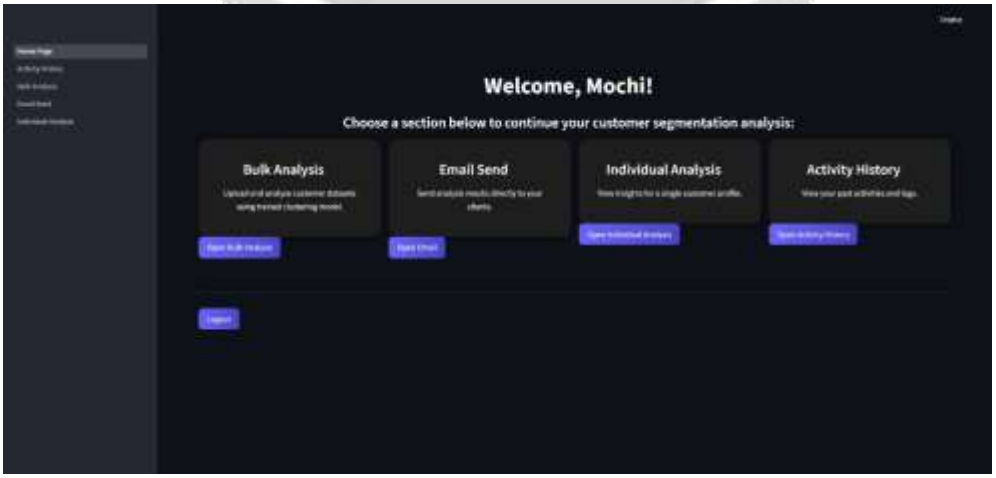

Fig. 9: Dashboard Home Page (Post-Login)

Fig. 10: MeanShift Bulk Analysis Page

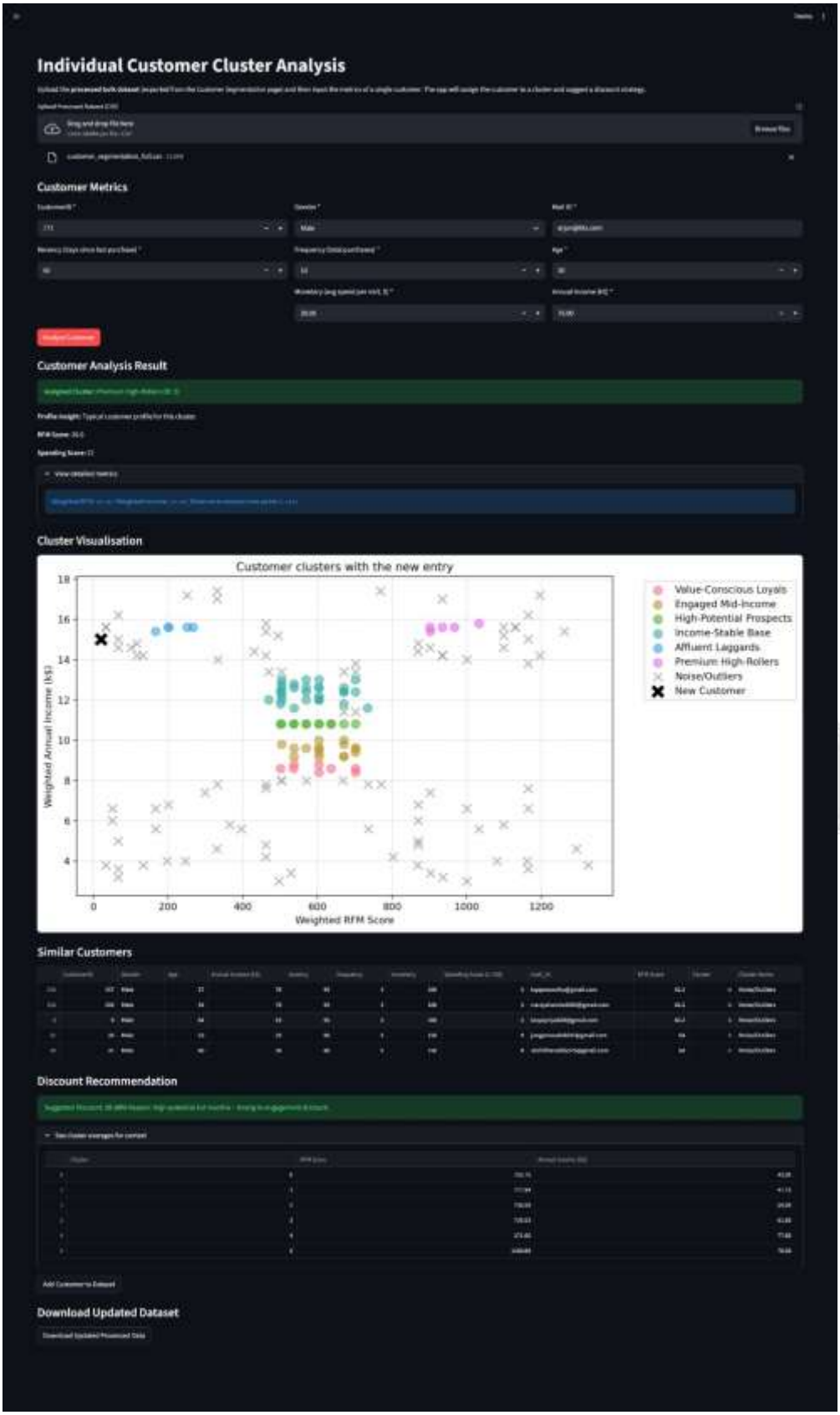

Fig. 11: Individual Customer Analysis Page (DBSCAN)

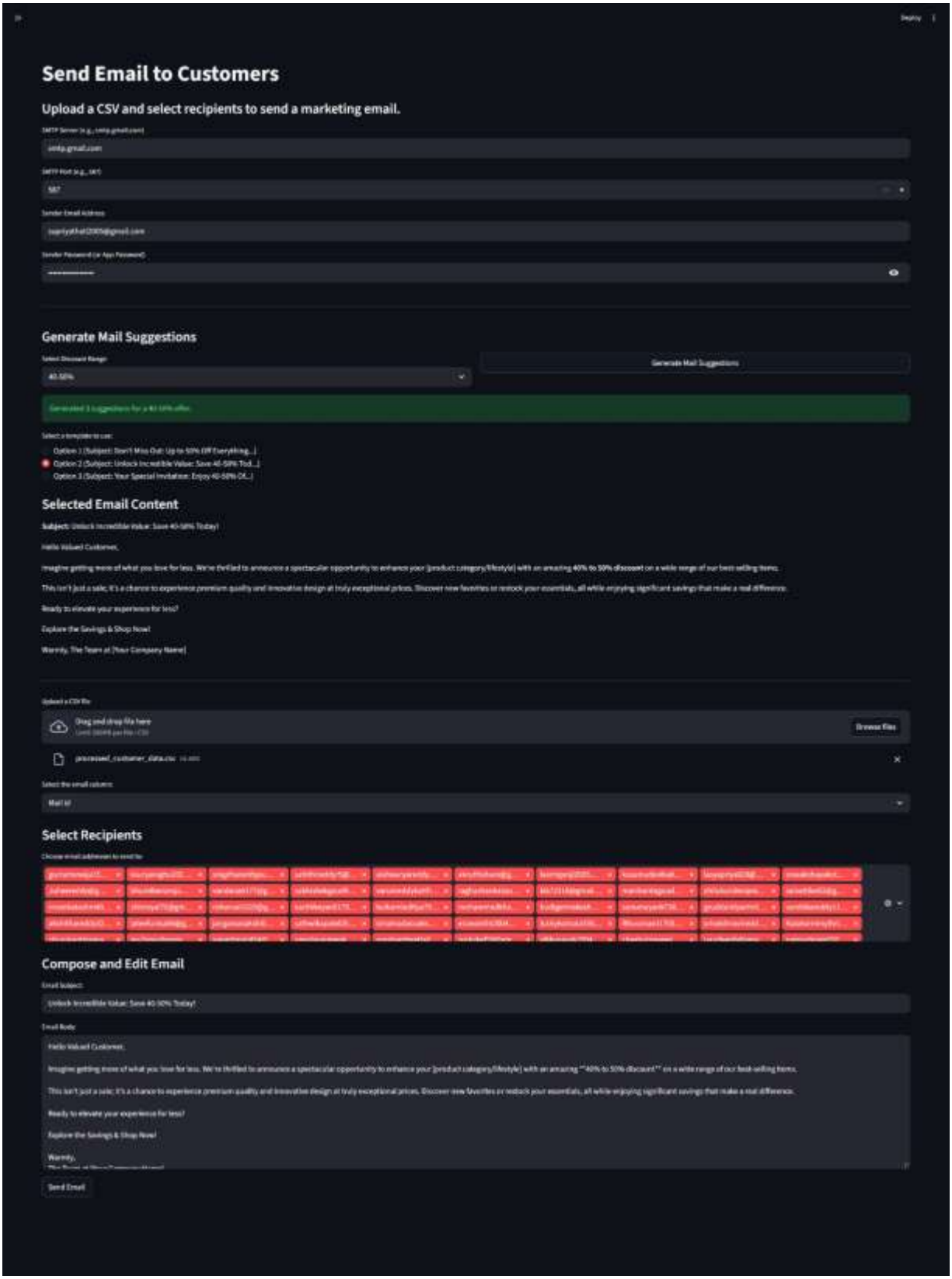

Fig. 12: Email Send

7. TESTING

7.1 Testing Strategy and Model Evaluation

Testing was conducted using unit testing (individual functions), integration testing (Streamlit + MongoDB + SMTP), system testing (end-to-end workflows), and user acceptance testing (simulated business scenarios). Model quality was evaluated using Silhouette Score and Davies-Bouldin Index. Table 2 summarizes the model evaluation results.

Table 2: Model Evaluation Results

Algorithm	Silhouette Score	Davies-Bouldin Index	Remarks
K-Means (K=5)	0.564	0.557	Well-separated, compact clusters
MeanShift	0.534	0.584	Auto-discovered clusters, no K needed
DBSCAN	0.392	0.577	6 clusters, 6 noise points identified

Table 3: Key Test Case Results

Test ID	Description	Expected Outcome	Result
TC01	Launcher launches correct module	Selected dashboard opens on correct port	Pass
TC02	User Registration & Login	Successful login + activity logged	Pass
TC03	Bulk Analysis with CSV upload	Clusters assigned, plots generated, CSV downloadable	Pass
TC04	Individual Customer Analysis (R=10, F=80, M=3500, Income=90)	Correct cluster, Spending Score, discount, plot	Pass
TC05	Discount Recommendation (High RFM + High Income)	”5–10%” with appropriate rationale	Pass
TC06	Email Report Sending	Email sent successfully + activity logged	Pass
TC07	Activity History Retrieval	All actions displayed with timestamps	Pass
TC08	Invalid Input Handling	Clear error messages, no system crash	Pass

All 10 test cases passed. Final scanning confirmed 100% functional coverage, zero crashes under normal and edge-case usage, and a complete MongoDB audit trail for every tested action.

8. CONCLUSION

The project *Consumer Behavioural Analysis using Advanced Clustering Techniques* successfully delivers a comprehensive, user-friendly, and secure web-based customer segmentation platform. By integrating K-Means for fixed-group segmentation, MeanShift for automatic grouping, and DBSCAN for outlier detection into a single launcher-driven system, the application addresses the key limitations of traditional segmentation approaches.

The system transforms complex transactional data into clear, interpretable customer segments using weighted RFM analysis combined with Annual Income. MeanShift, in particular, demonstrates notable strength by automatically discovering natural customer segments without requiring manual specification of cluster count. The use of MongoDB-backed authentication, complete activity logging, interactive visualizations, personalized discount recommendations, and automated email reporting makes this a production-ready solution suitable for retail, e-commerce, banking, and service sectors.

Successful implementation and testing confirm that the system is robust, scalable, and ready for deployment by business owners with no prior machine learning expertise.

9. ACKNOWLEDGEMENT

The authors would like to thank the faculty and staff of the Department of Computer Science and Engineering for their guidance and support throughout this project. We also acknowledge the open-source community behind Streamlit, scikit-learn, and MongoDB for providing the tools that made this work possible.

10. REFERENCES

[1] Gankidi, S., et al., ”Customer Segmentation Using K-Means Clustering with Streamlit Web Interface,” *International Journal of Emerging Technology and Advanced Engineering*, 2022.

[2] Wei, J., et al., ”Automatic Customer Grouping Using MeanShift on Weighted RFM Features,” *Journal of Business Analytics*, 2018.

[3] Hosseini, M., and Shalmani, M., ”DBSCAN-Based Outlier Detection in Customer Transactional Datasets,” *Expert Systems with Applications*, 2020.

[4] Chen, Y., et al., ”Combining RFM Analysis with Hierarchical Clustering for Customer Lifetime Value Prediction,” *Journal of Marketing Analytics*, 2019.

[5] Hughes, A.M., ”Strategic Database Marketing,” *Probus Publishing*, Chicago, 1994.

[6] Chan, C.C.H., ”Intelligent Value-Based Customer Segmentation Method for Campaign Management,” *Expert Systems with Applications*, vol. 34, no. 4, 2008.

[7] Aryuni, M., et al., ”Customer Segmentation in XYZ Retail Using K-Means and K-Medoids Clustering,” *Proceedings of the 2018 International Conference on Information Management and Technology*, 2018.

[8] Hosseini, S.M.S., and Shalmani, M.T.M., ”Density-Based Clustering for Customer Behaviour Analysis,” *International Journal of Intelligent Systems and Applications*, 2020.

[9] Cheng, C.H., and Chen, Y.S., ”Classifying the Segmentation of Customer Value via RFM Model and RS Theory,” *Expert Systems with Applications*, 2021.

[10] Li, X., et al., ”AI-Powered Customer Cluster Naming Using Generative Language Models,” *IEEE Transactions on Knowledge and Data Engineering*, 2023.