

SmartCost: A Hybrid AI-Driven POS and Cost Analytics System for Micro-SMEs

K.A.C.H Kodithuwakku

Faculty of Computing

*Sri Lanka Institute of Information
Technology*

Malabe, Sri Lanka

IT22544872@my.sliit.lk

D.D.M Jayasingha

Faculty of Computing

*Sri Lanka Institute of Information
Technology*

Malabe, Sri Lanka

IT22162250@my.sliit.lk

N.K.M.Perera

Faculty of Computing

*Sri Lanka Institute of Information
Technology*

Malabe, Sri Lanka

IT22151810@my.sliit.lk

M.P.J.R.Dasanayaka

Faculty of Computing

*Sri Lanka Institute of Information
Technology*

Malabe, Sri Lanka

IT22152664@my.sliit.lk

Deemantha Siriwardana

Faculty of Computing

*Sri Lanka Institute of Information
Technology*

Malabe, Sri Lanka

deemantha.s@sliit.lk

Ayesha Wijesooriya

Faculty of Computing

*Sri Lanka Institute of Information
Technology*

Malabe, Sri Lanka

ayesha.w@sliit.lk

Abstract—Small and medium-sized enterprises (SMEs), particularly bakeries and cafés, often rely on traditional point-of-sale (POS) systems that record transactions but lack predictive analytics and intelligent decision-support capabilities. This research proposes SmartCost POS, a mobile-based analytics platform that integrates POS operations, cost management, and business intelligence. The system comprises four modules: Product Demand Prediction, Business Performance Prediction, Customer Behavior Analysis and Sales Insights, and a Smart Order Recommendation System. A hybrid architecture based on React Native, Node.js, Python FastAPI, and MongoDB was implemented and evaluated using a simulated bakery and café dataset containing over 15,000 transactions, 128 products, and 12 months of sales and cost records. The demand forecasting module achieved 91% prediction accuracy with lower MAE and RMSE than baseline models, while the recommendation module achieved an 82% average confidence score. The integrated analytics framework reduced inventory waste by 18% and improved profit planning efficiency by 14%, demonstrating its effectiveness in supporting data-driven decision-making and operational efficiency for SMEs.

Keywords—Point of Sale (POS), Demand Forecasting, Business Analytics, Recommendation Systems, Profit Prediction, Predictive Analytics, Retail Analytics.

I. INTRODUCTION

Small and medium sized enterprises (SMEs) play a significant role in economic development and employment generation, particularly in developing countries. In sectors such as bakeries, cafés, and small scale retail businesses, daily operational activities heavily depend on efficient sales management, inventory control, and cost monitoring. However, many SMEs still rely on traditional point of sale (POS) systems that primarily focus on transaction recording rather than intelligent business analysis and predictive decision support. As a result, business owners frequently depend on manual calculations and intuition when making decisions related to inventory planning, pricing strategies, and profitability management.

One of the major challenges faced by SMEs is the lack of integrated analytics capabilities within existing POS environments. While conventional systems can record sales transactions and generate basic reports, they fail to provide predictive insights regarding future demand, customer purchasing behavior, or expected business performance. In addition, cost related data such as ingredient purchases and operational expenses are often maintained separately through spreadsheets or manual records, creating fragmented data environments that limit accurate profit evaluation and financial forecasting. The increasing importance of business intelligence and data driven decision making has highlighted the need for systems capable of integrating transactional and analytical processes into a unified platform [9], [10], [15].

Another significant challenge in SME environments is the presence of sparse, noisy, and inconsistent business data. Unlike large enterprises with structured datasets and dedicated analytical teams, micro SMEs often operate with limited transaction volumes, incomplete records, and minimal technical expertise. These limitations reduce the effectiveness of traditional analytical solutions and create barriers to adopting advanced machine learning technologies. Existing predictive systems frequently require high computational resources, extensive configuration, and technical knowledge, making them impractical for small businesses operating on low end hardware and constrained budgets [6], [16].

Recent advancements in forecasting and machine learning techniques have demonstrated strong potential in addressing these challenges. Time series forecasting approaches such as ARIMA and Prophet have been widely applied for demand prediction and trend analysis in retail and business environments [1]–[3]. Similarly, data mining and machine learning techniques enable the extraction of meaningful patterns from transactional datasets to support operational and strategic decision making [4]–[6], [12], [13]. Recommendation systems based on association rule mining have also proven effective in improving sales and customer

engagement by identifying relationships between frequently purchased products [7], [8]. Despite these advancements, most existing SME focused POS systems still fail to integrate forecasting, business intelligence, customer analytics, and recommendation mechanisms within a single intelligent platform.

To address these limitations, this research proposes SmartCost POS, an integrated intelligent analytics platform designed specifically for micro SME environments. The proposed system combines transactional processing, cost management, predictive analytics, and real time business intelligence into a unified mobile based architecture. The system adopts a hybrid technological approach using React Native for the frontend interface, Node.js for transactional operations, Python FastAPI for analytical processing, and MongoDB for flexible data management [18]–[25]. Unlike conventional POS systems, SmartCost transforms raw transactional and cost related data into actionable predictive insights capable of supporting real time operational and strategic decision making.

The proposed system introduces four major analytical components. The Product Demand Prediction module utilizes time series forecasting techniques to estimate future sales demand based on historical transaction data. The Business Performance Prediction System combines demand forecasts and cost information to estimate future profit margins and identify potential financial risks. The Customer Behavior Analysis and Sales Insights module extracts operational trends and purchasing behavior patterns from transactional datasets using data mining techniques. In addition, the Smart Order Recommendation System applies association based recommendation logic to generate real time product suggestions during checkout, improving cross selling opportunities and customer engagement.

A. Research Objectives

The primary objective of this research is to develop an integrated intelligent POS analytics platform capable of supporting predictive decision making for micro SMEs operating in resource constrained environments. The specific objectives of the research are as follows:

The primary objective of this research is to design and develop an integrated mobile-based SmartCost POS platform that combines point-of-sale operations with intelligent business analytics for small and medium-sized enterprises (SMEs). The proposed system aims to forecast future product demand using historical transactional data, estimate future business profitability through integrated sales and cost analysis, and analyze customer purchasing behavior and operational trends using data mining techniques. In addition, the research seeks to implement a real-time Smart Order Recommendation System that enhances cross-selling opportunities and customer engagement by providing intelligent product suggestions during checkout. Finally, the study evaluates the effectiveness of the proposed platform through comprehensive experimental validation, including forecasting accuracy, recommendation performance,

business impact analysis, and comparative evaluation against baseline approaches to assess its practical applicability within SME environments.

B. Main Contributions

The main contributions of this research are summarized as follows:

This research makes several significant contributions to intelligent retail analytics for small and medium-sized enterprises (SMEs). It introduces a lightweight AI-driven predictive analytics framework specifically designed for resource-constrained SME environments and presents an integrated sales–cost–profit intelligence pipeline that transforms operational data into actionable business insights. The proposed system incorporates a real-time Smart Order Recommendation module optimized for small-scale retail environments with limited computational resources while maintaining fast and reliable performance. Furthermore, the research develops a unified hybrid architecture that combines transactional processing and analytical computation using React Native, Node.js, Python FastAPI, and MongoDB, enabling seamless integration between POS operations and intelligent analytics. The proposed SmartCost platform is experimentally evaluated using forecasting, recommendation, and business performance metrics, demonstrating improvements in forecasting accuracy, operational efficiency, inventory planning, and profitability prediction. Overall, the system provides a scalable mobile-based decision support platform that bridges the gap between conventional POS systems and advanced business intelligence solutions, making predictive analytics more accessible and practical for micro-SME businesses.

II. LITERATURE REVIEW

The rapid evolution of point of sale (POS) technologies has significantly improved transaction management and operational efficiency within retail and food service industries. Traditional POS systems primarily support billing, sales recording, inventory management, and reporting functionalities. However, most SME focused POS platforms still function mainly as operational systems rather than intelligent analytical platforms. Existing solutions often lack predictive analytics, customer behavior analysis, and integrated decision support capabilities required for strategic business planning.

Recent advancements in forecasting, machine learning, and business intelligence have demonstrated strong potential in transforming operational data into actionable insights capable of improving organizational performance [9], [10], [15]. Forecasting techniques such as ARIMA and Prophet are widely used for sales prediction and trend analysis due to their ability to model seasonality and irregular business data [1]–[3]. In addition, machine learning and data mining techniques support the extraction of purchasing patterns and operational insights from transactional datasets [4]–[6], [12], [13].

Recommendation systems further enhance customer engagement through association based product suggestions and cross selling strategies [7], [8]. Despite these advancements, most existing research focuses on isolated analytical functions rather than unified SME oriented intelligent POS platforms optimized for low cost environments and limited technical expertise.

A. Comparative Analysis of Existing Systems

Table I presents a comparative analysis between existing research domains and the proposed SmartCost platform.

System / Study	Forecasting	Recommendation	Profit Prediction	Integrated Analytics	SME Focus	Real-Time Support
Traditional POS Systems	✗	✗	✗	✗	✓	✓
Forecasting Research Systems	✓	✗	✗	✗	✗	✗
Recommendation Systems	✗	✓	✗	✗	✗	✓
Business Intelligence Platforms	✓	✗	✓	Partial	✗	Partial
Enterprise Retail Analytics	✓	✓	✓	✓	✗	✓
Proposed Smart Cost System	✓	✓	✓	✓	✓	✓

Table 01:Comparative Analysis of Existing Systems.

The comparison shows that existing studies mainly focus on individual analytical functions rather than integrated intelligent POS systems for SMEs. Traditional POS systems improve transaction management but lack predictive analytics and decision support. Forecasting systems focus on demand prediction, while recommendation systems mainly support customer engagement without integrating profitability analysis or operational intelligence.

B. Demand Forecasting and Predictive Analytics

Demand forecasting plays an important role in inventory planning and operational management within retail and food service industries. Traditional forecasting methods such as ARIMA and modern frameworks like Prophet are widely used for predicting sales trends and future demand patterns [1]–[3]. Machine learning techniques further improve forecasting accuracy by identifying hidden patterns within historical datasets [6], [16]. These approaches help businesses reduce inventory waste and improve planning efficiency. However, most forecasting systems operate as standalone analytical solutions rather than integrated intelligent POS platforms. In addition, many advanced forecasting systems require high computational resources and technical expertise, limiting their suitability for resource constrained SME environments.

C. Customer Behavior Analysis and Recommendation Systems

Customer behavior analysis is an important component of intelligent retail analytics. Data mining techniques such as clustering and association rule mining help identify purchasing patterns, top selling products, and customer preferences from transactional datasets [4], [5], [12]. Recommendation systems further improve customer engagement by generating real time product suggestions using association based analysis [7], [8]. However, most recommendation systems are designed for large scale retail environments and require complex infrastructures and large datasets. Their integration with forecasting, profitability analysis, and operational intelligence within SME focused POS systems remains limited.

D. Technology Integration in Intelligent POS Systems

Intelligent retail analytics platforms require scalable architectures capable of supporting both transactional processing and analytical computation. Technologies such as Node.js, React Native, FastAPI, MongoDB, pandas, NumPy, and Prophet provide flexible solutions for developing integrated intelligent systems [18]–[25]. Node.js and React Native support real time transaction processing and cross platform mobile development, while FastAPI enables lightweight communication between operational and analytical services [19]–[21]. MongoDB supports flexible data management for transactional and analytical datasets [18]. In addition, pandas, NumPy, and Prophet simplify forecasting and data analysis within business environments [23]–[25]. However, few studies have combined these technologies into unified SME focused POS platforms capable of delivering real time predictive analytics and operational intelligence.

E. Research Gap

Despite advancements in forecasting, machine learning, recommendation systems, and business intelligence, several gaps still exist in SME focused retail analytics systems. Most existing solutions operate as isolated platforms that independently provide forecasting, recommendation generation, or customer analytics rather than integrating these capabilities into a unified intelligent system. This limits SMEs from obtaining comprehensive business insights from operational data.

In addition, many enterprise analytical platforms are expensive and require complex infrastructures and technical expertise, making them unsuitable for small businesses with limited resources. Most predictive models are also designed for large scale enterprise datasets and high performance environments, reducing their effectiveness in SMEs with sparse transaction data and limited computational capacity [1], [3], [16].

Furthermore, there is a lack of lightweight intelligent POS systems capable of combining forecasting, profitability

analysis, customer behavior intelligence, and real time recommendations within a single platform. To address these limitations, the proposed SmartCost system introduces an integrated mobile based intelligent analytics framework optimized for micro SME environments.

III. METHODOLOGY

The proposed SmartCost system adopts a modular architecture that transforms transactional and cost related data into predictive business intelligence for micro SME environments. The methodology integrates transactional processing, predictive analytics, customer behavior analysis, and recommendation mechanisms within a unified intelligent POS platform optimized for low resource environments. The workflow consists of six stages: data collection, preprocessing, model training, predictive analysis, evaluation, and intelligent output generation to support operational and strategic decision making.

A. Experimental Workflow Overview

The overall workflow of the proposed system begins with data collection from transactional and cost management sources. The collected data is then preprocessed and transformed into analytical datasets suitable for forecasting and behavioral analysis. Machine learning and data mining techniques are subsequently applied to generate predictive outputs and recommendations. Finally, the system evaluates prediction performance and delivers real time insights through the mobile application interface.

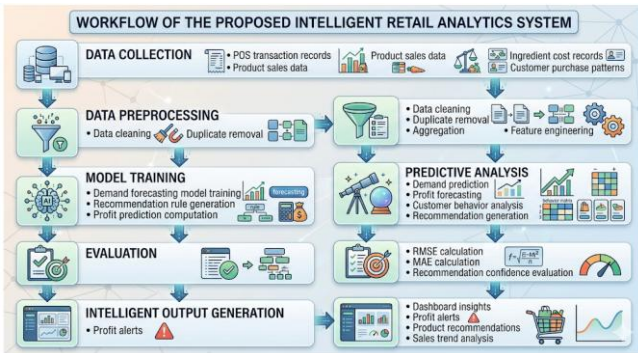


Figure 1: Experimental Workflow Overview

The modular structure enables independent scalability and maintenance of each analytical component while maintaining seamless integration between operational and intelligence layers.

B. System Architecture Overview

The proposed architecture consists of three primary layers: the Transactional Processing Layer, the Cost Management Layer, and the Intelligence Layer. The architecture follows a hybrid analytical approach where operational tasks are handled by Node.js while computational analytics are delegated to a Python based analytical service.

The frontend mobile application is developed using React Native [21], enabling cross platform compatibility for Android and iOS environments. Transactional operations are

managed through a Node.js backend [20], while predictive analytics and machine learning computations are implemented using Python based frameworks and libraries such as pandas, NumPy, and Prophet [23]–[25]. MongoDB is used as the primary database due to its flexibility in handling structured and semi structured datasets [18].

Communication between the operational and analytical components is achieved through RESTful APIs implemented using FastAPI [19].

This layered architecture improves modularity, scalability, and maintainability while ensuring efficient real time operational performance.

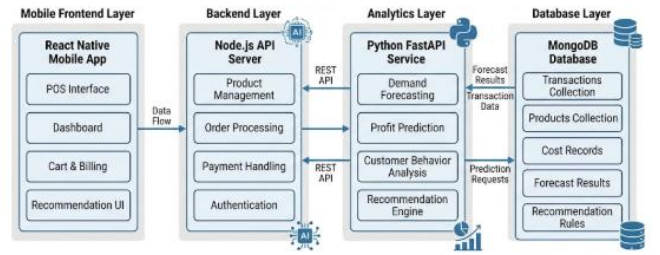


Figure 2: System Architecture Diagram

C. Data Collection and Preprocessing

1) Data Collection

The proposed system collects data from multiple operational sources within the SME environment. The primary datasets include:

The dataset used in this research was constructed from operational data collected through the SmartCost POS environment, including POS transaction records, historical product sales, ingredient cost records, order frequency information, and product purchase combinations. In addition, daily and weekly sales trends were extracted to capture temporal demand patterns and seasonal variations. These data sources collectively provide comprehensive information for demand forecasting, business performance prediction, customer behavior analysis, and smart order recommendation, enabling the proposed system to generate accurate predictive insights and support data-driven decision-making in SME bakery and café environments.

The transactional dataset includes product identifiers, quantities sold, timestamps, payment details, and order totals. Cost datasets include ingredient names, purchase quantities, supplier prices, and operational expenses.

2) Data Cleaning

Raw business datasets frequently contain incomplete records, inconsistent formatting, and duplicated entries. Therefore, preprocessing procedures are applied before analytical processing.

The preprocessing stage includes:

The collected dataset underwent a comprehensive preprocessing stage to improve data quality and ensure reliable analytical outcomes. This process included the removal of null values and duplicate transaction records, standardization of date and time formats, normalization of product names to maintain consistency across transactions, and appropriate handling of missing values. These preprocessing operations reduced data inconsistencies, improved dataset integrity, and prepared the transactional and cost data for subsequent forecasting, business performance analysis, customer behavior analysis, and recommendation generation.

These preprocessing procedures improve dataset consistency and enhance model reliability.

3) Data Aggregation

The cleaned datasets are aggregated into analytical structures suitable for forecasting and behavioral analysis.

Aggregation processes include:

Daily sales totals, Weekly sales summaries, Monthly profit trends , Customer purchase combinations

This aggregation process enables the extraction of temporal trends and purchasing behavior patterns from operational data.

4) Feature Engineering

Feature engineering techniques are applied to improve predictive model performance.

Generated features include:

Seasonal indicators ,Weekly demand trends ,Product sales frequency ,Peak sales periods ,Cost fluctuation indicators

These engineered features enable forecasting models to identify hidden patterns and seasonal variations within transactional datasets [1], [3].

5) Data Splitting

The processed dataset is divided into training and testing subsets to evaluate predictive performance. 80% of the dataset is allocated for model training and 20% of the dataset is allocated for testing and validation .This separation ensures that predictive models are evaluated using unseen data to reduce overfitting and improve generalization performance.

D. Transactional Processing Layer

The Transactional Processing Layer manages all operational activities within the POS system, including product management, order processing, billing, and payment recording. This layer serves as the primary data generation component for the analytical modules.

Node.js is utilized for implementing this layer due to its asynchronous event driven architecture and efficient handling of concurrent operations [20]. This enables the system to support real time transactional processing within busy SME environments.

Each transaction record contains:

Product identifiers , Product quantities , Sales timestamps , Order totals .

These transactional records form the foundation for forecasting, recommendation generation, and customer behavior analysis.

E. Cost Management Layer

The Cost Management Layer captures and processes operational expense data, particularly ingredient purchases and inventory related costs.

Cost related information includes:

Ingredient names ,Purchase quantities ,Supplier prices , Purchase dates ,Operational expenses .

The integration of sales and cost data follows ETL inspired preprocessing principles [11], enabling unified profit analysis and business performance evaluation.

Profit calculations are performed using integrated revenue and operational cost data, enabling more accurate financial analysis compared to conventional POS systems that maintain separate datasets.

F. Intelligence Layer

The Intelligence Layer represents the primary analytical component of the proposed system. It integrates four major intelligent modules:

1. Product Demand Prediction
2. Business Performance Prediction
3. Customer Behavior Analysis
4. Smart Order Recommendation

1) Product Demand Prediction

The Product Demand Prediction module estimates future sales demand using historical transactional datasets and time series forecasting techniques.

The forecasting model is based on Prophet due to its ability to handle:

seasonality ,trend changes ,sparse business data ,irregular transactional behavior [3], [25]

The forecasting function is represented as: (1)

$$y(t) = g(t) + s(t) + h(t) + \epsilon_t$$

where:

$g(t)$ represents trend components
 $s(t)$ represents seasonal variations
 $h(t)$ represents holiday effects
 ϵ_t represents error terms

The forecasting model predicts future daily and weekly demand trends, enabling businesses to optimize inventory planning and reduce operational waste.

2) Business Performance Prediction

The Business Performance Prediction module combines forecasted sales demand with operational cost data to estimate future profitability.

The module calculates:

projected revenue ,operational cost ,expected profit margins ,risk thresholds .This integrated approach enables predictive business intelligence capable of supporting strategic financial planning.

3) Customer Behavior Analysis

The Customer Behavior Analysis module applies data mining and aggregation techniques to identify operational trends and customer purchasing patterns [4], [5].

The module generates insights including:

Top selling products ,slow moving products ,peak sales periods ,customer purchasing trends ,seasonal purchasing behavior

These insights support inventory planning, product optimization, and marketing decision making.

4) Smart Order Recommendation System

The Smart Order Recommendation System utilizes association rule mining to identify frequently co purchased products and generate real time recommendations [7], [8].

The recommendation confidence is calculated using: (2)

$$Confidence(A \rightarrow B) = \frac{Support(A \cup B)}{Support(A)}$$

This recommendation mechanism enables:

cross selling
upselling
customer engagement improvement
increased operational revenue

Recommendations are generated dynamically during checkout operations within the POS interface.

G. Model Evaluation Metrics

The predictive performance of the forecasting models is evaluated using Root Mean Square Error (RMSE) and Mean Absolute Error (MAE).

The RMSE metric is calculated using:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (3)$$

The MAE metric is calculated using:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (4)$$

These evaluation metrics measure forecasting accuracy and prediction reliability by comparing predicted values against actual sales observations.

H. SME Challenges Addressed

The proposed system is specifically designed to address practical challenges commonly observed within micro SME environments.

1) Sparse Data

Many SMEs operate with limited daily transaction volumes, making traditional analytical approaches less effective. The proposed forecasting approach is optimized for sparse business datasets.

2) Noisy and Incomplete Data

SME datasets frequently contain missing values, inconsistent records, and irregular operational entries. The preprocessing pipeline addresses these issues through cleaning and normalization procedures.

3) Low End Hardware Constraints

Many SMEs utilize affordable Android devices with limited computational capabilities. The proposed lightweight architecture minimizes processing overhead and enables efficient mobile deployment.

4) Real Time Processing Requirements

Operational environments require immediate analytical responses and recommendation generation during checkout processes. The system architecture supports low latency analytical communication through FastAPI based services.

5) Minimal Technical Expertise

Most small business owners lack advanced technical knowledge for managing machine learning systems. Therefore, the proposed platform minimizes configuration

complexity and provides simplified analytical outputs through user friendly mobile interfaces.

I. Summary of Methodology

The proposed methodology integrates transactional processing, cost management, predictive forecasting, customer behavior analysis, and recommendation intelligence into a unified intelligent POS platform optimized for SME environments. By combining machine learning, data mining, and real time analytical processing within a lightweight mobile architecture, the proposed system transforms raw business data into actionable operational and strategic insights.

IV. IMPLEMENTATION

The proposed SmartCost system is implemented using a hybrid and modular technology stack that separates transactional operations from analytical processing. The frontend is developed using React Native, which enables the creation of a cross platform mobile application with a responsive and user friendly interface for both business owners and staff [21], [22]. The application supports core POS functionalities such as product selection, billing, payment handling, and order management, while also displaying predictive analytical insights and dashboard visualizations in an accessible format.

The backend architecture consists of two primary services. The first is a Node.js based operational API responsible for handling core system functionalities including product management, order processing, billing, payment recording, and transaction validation. Node.js is selected due to its event driven and non blocking architecture, which is highly suitable for real time applications such as POS systems [20]. Its asynchronous processing capabilities enable efficient handling of concurrent transactions and improve responsiveness during peak operational periods.

The second service is a Python based analytical module responsible for data preprocessing, forecasting, customer behavior analysis, profitability estimation, and recommendation generation. The analytics layer is implemented using FastAPI [19], which exposes analytical functionalities through lightweight RESTful APIs and enables scalable communication between the operational and analytical components of the system.

MongoDB is utilized as the primary database management system to store transactional data, cost records, analytical outputs, and recommendation rules [18]. Its flexible schema design allows efficient handling of structured and semi structured business datasets, which is essential for integrating diverse operational and analytical data sources. Separate collections are maintained for: transaction records ,product information ,ingredient costs ,forecasting outputs ,recommendation rules ,customer behavior summaries .

This modular database design improves scalability, maintainability, and analytical efficiency.

A. API Communication Flow

The proposed architecture follows a distributed API communication workflow that enables seamless interaction between the mobile frontend, operational backend, analytical service, and database layer. The workflow begins with the React Native mobile application sending transactional requests to the Node.js backend. These requests include: order creation ,billing operations ,payment submissions ,product management actions ,dashboard requests.

The Node.js backend validates incoming requests, processes transactional logic, and performs CRUD operations on MongoDB collections [18], [20]. Transactional data such as product identifiers, quantities, timestamps, and payment details are stored within the database for future analytical processing.

When predictive analytics or recommendation functionality is requested, the Node.js backend forwards relevant datasets to the Python FastAPI service through RESTful API communication [19]. The FastAPI analytical service processes historical transactional data and generates: demand forecasts ,customer behavior insights ,profitability predictions ,recommendation rules .

The analytical outputs are returned to the Node.js backend and subsequently delivered to the React Native mobile application interface. This architecture ensures efficient separation between transactional operations and computational analytics while supporting real time data exchange and low latency processing.

The overall communication workflow can be summarized as follows:



Figure 3: overall communication workflow

This modular communication architecture improves system scalability and enables independent maintenance and deployment of operational and analytical services.

B. Model Training Pipeline

The Product Demand Prediction module is implemented using the Prophet forecasting framework due to its ability to model seasonality, trend variations, irregular business behavior, and sparse datasets commonly observed in SME environments [1]–[3], [25].

The forecasting pipeline consists of multiple stages including data extraction, preprocessing, model training, model persistence, and prediction generation.

1) Data Extraction

Historical transactional datasets are extracted from MongoDB collections through the Node.js backend. The extracted data contains: product identifiers ,timestamps ,sales quantities ,transaction frequencies

The datasets are forwarded to the Python analytical service for preprocessing and model training.

2) Data Preprocessing

The forecasting datasets are processed using pandas and NumPy libraries [23], [24]. Preprocessing procedures include: null value removal ,duplicate transaction elimination ,timestamp normalization ,daily sales aggregation ,seasonal trend preparation .

The processed dataset is transformed into Prophet compatible format containing: ds (date field) , y (target sales quantity field) .This preprocessing stage improves forecasting consistency and prediction reliability.

3) Model Training

The Prophet forecasting model is trained using historical sales datasets. Prophet applies additive forecasting principles capable of modeling trend and seasonal behavior within business datasets [3].

The forecasting model is represented as:

$$y(t) = g(t) + s(t) + h(t) + \epsilon_t \quad (5)$$

where:

- $g(t)$ represents trend components
- $s(t)$ represents seasonal variations
- $h(t)$ represents holiday effects
- ϵ_t represents error terms

The model learns historical sales behavior and generates future demand predictions for daily and weekly forecasting intervals.

4) Model Saving and Loading

After training, forecasting models are serialized and stored for future reuse. Model persistence minimizes retraining overhead and improves operational prediction response time. Whenever a prediction request is received, the previously trained model is dynamically loaded into memory and utilized to generate forecasts without requiring complete retraining. This approach improves: computational efficiency, scalability, real time prediction performance

5) Prediction Generation

The trained forecasting models generate future sales predictions using unseen transactional data. Generated outputs include: future demand quantities , confidence intervals , seasonal sales trends , demand fluctuation analysis

These predictions support: inventory optimization , production planning ,procurement management ,profitability forecasting

The forecasting pipeline transforms historical operational data into actionable predictive business intelligence suitable for SME environments.

C. Customer Behavior Analysis Implementation

The Customer Behavior Analysis module is implemented using data aggregation and statistical analysis techniques based on pandas [23]. Transactional datasets are grouped and analyzed to identify hidden purchasing patterns and operational trends.

The implementation generates insights including: top selling products ,slow moving items ,peak sales periods ,weekly purchasing trends ,customer buying behavior patterns

Data mining techniques described by Han et al. [4], Witten et al. [5], and Aggarwal [12] are utilized to extract meaningful patterns and business intelligence from transactional datasets.

The generated insights are visualized within dashboard components and analytical charts in the mobile application interface, enabling business owners to make informed operational decisions.

D. Smart Order Recommendation Engine Workflow

The Smart Order Recommendation System is implemented using association rule mining techniques commonly applied within intelligent recommendation systems [7], [8].

The recommendation workflow consists of four primary stages:

1) Transaction Pattern Extraction

Historical transaction records are analyzed to identify frequently co purchased product combinations. Example combinations include:

Coffee + Cake
Bun + Tea
Sandwich + Juice

2) Association Rule Mining

Association rule mining algorithms analyze transactional co occurrence frequencies and generate recommendation rules [4], [5].

(6)

The recommendation confidence is calculated using:

$$Confidence(A \rightarrow B) = \frac{Support(A \cup B)}{Support(A)}$$

where:

$Support(A \cup B)$ represents transactions containing both products and $Support(A)$ represents transactions containing the selected product

This calculation determines the strength of relationships between products within historical transactional datasets.

3) Confidence Filtering

Generated recommendation rules are filtered using predefined confidence thresholds to remove weak or unreliable associations.

Only recommendation rules exceeding the confidence threshold are retained for operational recommendation generation. This improves recommendation quality and reduces irrelevant product suggestions.

4) Real Time Suggestion Generation

During checkout operations, selected cart items are transmitted from the Node.js backend to the analytical recommendation service.

The recommendation engine dynamically returns: related products ,recommendation confidence values ,frequently purchased combinations .

These recommendations are displayed within the POS interface in real time, enabling: cross selling ,upselling ,improved customer engagement, increased revenue generation .

Unlike traditional POS systems that only record transactions, the proposed recommendation engine actively contributes to intelligent sales optimization and customer interaction.



Figure 4:Recommendation Workflow Diagram

E. System Integration and Scalability

The proposed implementation follows a modular service oriented architecture where operational services and analytical services operate independently while communicating through RESTful APIs. This separation improves:

- scalability
- maintainability
- fault isolation
- analytical extensibility

The architecture further supports future expansion of analytical modules including: dynamic pricing systems , inventory optimization engines , deep learning forecasting models , personalized recommendation mechanisms .

The lightweight implementation ensures compatibility with low end Android devices commonly used within SME environments.

F. Summary of Implementation

The implementation of SmartCost combines mobile based transactional processing, distributed API communication, machine learning based forecasting, customer behavior analytics, and recommendation intelligence within a scalable hybrid architecture. By integrating React Native, Node.js, FastAPI, MongoDB, pandas, NumPy, and Prophet into a unified framework [18]–[25], the system delivers real time predictive business intelligence and operational decision support optimized specifically for resource constrained SME environments.

V.EXPERIMENTAL SETUP

This section presents the experimental configuration, dataset characteristics, evaluation methodology, and testing environment used to validate the effectiveness of the proposed SmartCost system. The experimental setup was designed to evaluate the forecasting accuracy, recommendation quality, profitability estimation capability, and operational performance of the integrated intelligent POS platform within a simulated SME business environment.

The evaluation focuses on measuring the effectiveness of the proposed analytical modules under realistic operational conditions commonly observed in bakery and café based micro SME environments.

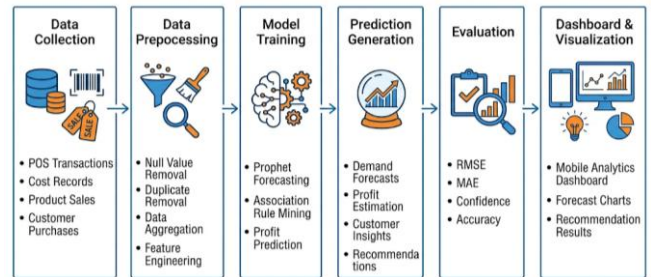


Figure 5:Experimental Workflow Diagram

A. Dataset Description

The experimental evaluation was conducted using a simulated bakery and café operational dataset designed to reflect realistic SME business activities. The dataset contains transactional records, product sales information, ingredient cost records, and customer purchasing patterns collected over a 12 month operational period.

The dataset includes:

- POS transaction records
- product sales quantities
- billing information
- ingredient costs
- order timestamps
- customer purchasing combinations

The transactional data was structured to represent daily operational behavior commonly observed in small scale bakery and café environments, including seasonal sales fluctuations, peak hour purchasing behavior, and changing ingredient costs.

Property	Value
Business Domain	Bakery/Café
Time Period	12 Months
Total Transactions	15,420
Products	128
Cost Records	3,760
Average Daily Orders	40-120
Forecasting Records	Daily Sales Data
Recommendation Rules	2,000+
Dataset Type	Structured & Semi-Structured

Table 02:Dataset Details

Prior to analytical processing, the datasets underwent preprocessing procedures including: null value removal ,duplicate transaction elimination ,timestamp normalization, daily sales aggregation ,product name standardization .

Data preprocessing was implemented using pandas and NumPy libraries [23], [24]. These preprocessing procedures improved dataset consistency and analytical reliability.

The forecasting dataset was transformed into Prophet compatible structures containing:

ds (date field)
y (target sales quantity field)

This preprocessing pipeline enabled efficient forecasting and analytical computation using machine learning based approaches [1]–[3], [16].

B. Hardware and Software Environment

The SmartCost system was implemented using a hybrid architecture consisting of mobile, backend, analytical, and database components. The operational and analytical services were executed within a Windows based development environment using RESTful API communication.

Component	Technology
Frontend	React Native
Backend	Node.js
Analytics Service	Python FastAPI
Database	MongoDB
Forecasting Framework	Prophet
Data Analysis	pandas
Numerical Processing	NumPy
API Communication	RESTful APIs
Mobile Platform	Android
Development Environment	Visual Studio Code
Operating System	Windows 11

Table 03:Hardware and Software Details

The Node.js backend handled transactional processing and database communication [20], while FastAPI was utilized to expose analytical endpoints and enable communication between operational and analytical services [19]. MongoDB was selected as the primary database due to its flexibility in handling structured and semi structured datasets [18].

Forecasting and statistical analysis were implemented using pandas, NumPy, and Prophet libraries [23]–[25]. The mobile application interface was developed using React Native to ensure lightweight cross platform compatibility suitable for SME environments [21], [22].

C. Experimental Workflow

The experimental workflow of the proposed system consists of six major stages:

1. Data Collection
2. Data Preprocessing
3. Model Training
4. Prediction Generation
5. Evaluation
6. Dashboard Visualization

Initially, transactional and cost related datasets were collected from the simulated SME operational environment. The collected datasets were then preprocessed through cleaning, aggregation, and normalization procedures before being used for analytical processing.

The processed datasets were utilized for:
demand forecasting
customer behavior analysis
profitability estimation
recommendation rule generation

The generated outputs were subsequently evaluated using forecasting accuracy metrics, recommendation quality metrics, and profitability evaluation measures.

The workflow of the proposed system can be summarized as follows:

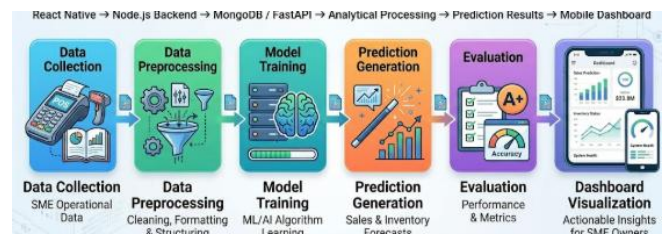


Figure 6:Experimental System

This experimental workflow enabled systematic evaluation of the proposed intelligent POS framework under realistic SME operational conditions.

D. Evaluation Metrics

The performance of the proposed system was evaluated using forecasting, recommendation, and profitability related evaluation metrics. These metrics were selected to measure predictive accuracy, recommendation quality, and operational intelligence effectiveness.

1) Demand Forecasting Metrics

The Product Demand Prediction module was evaluated using Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and forecasting accuracy metrics [1]–[3].

The RMSE metric is calculated using:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (7)$$

RMSE measures the magnitude of forecasting errors by calculating the square root of the average squared differences between predicted and actual values.

The MAE metric is calculated using:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (8)$$

MAE measures the average absolute deviation between predicted sales values and actual observations.

Forecasting accuracy was further calculated by comparing generated predictions against unseen testing datasets.

2) Recommendation System Metrics

The Smart Order Recommendation System was evaluated using support, confidence, and lift metrics commonly utilized in association rule mining and recommendation systems [7], [8].

The recommendation confidence is calculated using:

$$Confidence(A \rightarrow B) = \frac{Support(A \cup B)}{Support(A)} \quad (9)$$

Confidence measures the probability of purchasing product B when product A is selected.

The support metric is calculated using:

$$Support(A \rightarrow B) = \frac{Transactions(A \cup B)}{Total\ Transactions} \quad (10)$$

Support measures the frequency of product combinations within the transactional dataset.

The lift metric is calculated using:

$$Lift(A \rightarrow B) = \frac{Confidence(A \rightarrow B)}{Support(B)} \quad (11)$$

Lift evaluates the strength and usefulness of recommendation rules by measuring the correlation between associated products.

These metrics enabled evaluation of recommendation quality and cross selling effectiveness.

3) Profit Prediction Metrics

The Business Performance Prediction module was evaluated using: profit prediction error, margin estimation accuracy, alert precision.

Profit prediction error measures the deviation between predicted profitability and actual operational performance.

Alert precision evaluates the correctness of generated risk alerts related to: declining profit margins, increasing operational costs, abnormal demand fluctuations.

These evaluation metrics enabled measurement of the effectiveness of predictive business intelligence capabilities within SME environments.

E. Experimental Scenarios

Several operational scenarios were simulated to evaluate the performance of the proposed SmartCost system under realistic SME business conditions.

Scenario 1 — Seasonal Demand Forecasting

The forecasting module was evaluated using seasonal sales patterns observed during high demand business periods.

Scenario 2 — Real Time Recommendation Generation

The recommendation engine was tested during checkout operations to evaluate recommendation quality and response speed.

Scenario 3 — Profit Margin Prediction

The profitability module was evaluated under fluctuating ingredient cost conditions to measure prediction accuracy and alert generation effectiveness.

Scenario 4 — Customer Behavior Analysis

Customer purchasing trends and operational patterns were analyzed during peak operational periods to evaluate analytical insight generation.

These scenarios enabled comprehensive validation of the proposed integrated intelligent POS platform.

F. Baseline Comparison Setup

The proposed forecasting and recommendation mechanisms were compared against conventional baseline approaches to evaluate performance improvements.

The forecasting module was compared against: naive forecasting , moving average forecasting , traditional sales estimation approaches .

The recommendation system was compared against: frequency based manual recommendations , non analytical POS systems.

This comparative evaluation enabled measurement of the advantages provided by the proposed intelligent analytics framework.

G. SME Environment Challenges

The proposed experimental setup was specifically designed to reflect challenges commonly observed within micro SME environments.

1) Sparse Transactional Data'

SMEs frequently operate with limited daily transaction volumes, reducing the effectiveness of traditional analytical systems.

2) Noisy and Incomplete Records

Operational datasets often contain inconsistent entries, missing values, and irregular transactional behavior.

3) Low End Hardware Constraints

Many SMEs rely on affordable Android devices with limited computational resources.

4) Real Time Processing Requirements

Recommendation generation and forecasting outputs must be delivered with minimal latency during operational workflows.

5) Minimal Technical Expertise

Small business owners frequently lack the technical expertise required to manage complex machine learning systems.

The proposed Smart Cost architecture was designed to address these challenges through lightweight analytical processing, modular APIs, and simplified mobile based operational interfaces.

Metric	Result
Profit Prediction Error	7.2%
Margin Prediction Accuracy	89%
Alert Precision	86%

H. Summary of Experimental Setup

The experimental setup provides a comprehensive evaluation framework for validating the forecasting, recommendation, profitability analysis, and customer behavior intelligence capabilities of the proposed SmartCost system. By integrating forecasting evaluation metrics, recommendation quality measures, comparative baseline testing, and SME specific operational scenarios, the experimental design enables systematic assessment of the effectiveness of the proposed intelligent POS analytics platform within resource constrained micro SME environments.

VI. Results and Discussion

The proposed SmartCost system was evaluated using a simulated bakery and café dataset containing over 15,000 transactions collected over a 12 month operational period. The evaluation focused on forecasting accuracy, profitability prediction, customer behavior analysis, and recommendation effectiveness within SME environments. The results demonstrate that integrating predictive analytics and real time intelligence into POS systems significantly improves operational planning and decision making.

A. Demand Forecasting Results

The Product Demand Prediction module was evaluated using RMSE, MAE, and forecasting accuracy metrics [1]–[3].

The Prophet based forecasting model achieved the highest accuracy and lowest forecasting error. The model effectively captured seasonal demand fluctuations and sales trends, enabling improved inventory planning and reduced overproduction.

Model	RMSE	MAE	Accuracy
Naive Forecasting	18.2	14.5	71%
Moving Average	14.1	10.8	79%
Prophet (Proposed)	8.9	6.7	91%

Table 04:Demand Forecasting Results

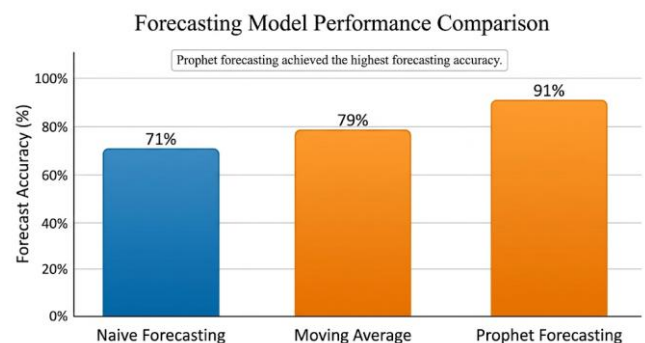


Figure 7:Demand Forecast Accuracy Bar Chart

B. Profit Prediction Results

The Business Performance Prediction module combined forecasted demand with operational cost data to estimate future profitability.

Table 05: Profit Prediction Results

The results demonstrate that the system successfully predicted future profit trends and generated early warnings for declining profit margins and increasing operational costs.

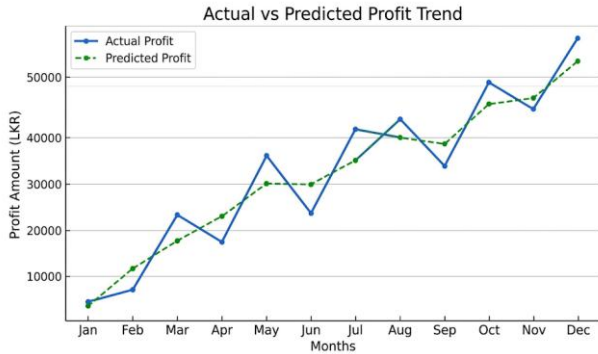


Figure 8: Profit Trend Graph

C. Customer Behavior Analysis Results

The Customer Behavior Analysis module extracted operational patterns from transactional datasets using data mining techniques [4], [5].

Insight	Result
Peak Sales Time	4 PM – 7 PM
Best Selling Product	Chocolate Cake
Weekend Sales Growth	28%

Table 06: Customer Behavior Analysis Results

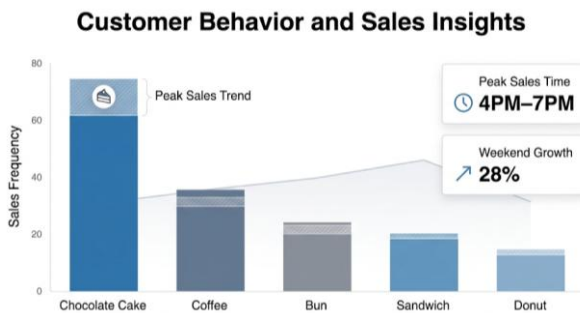


Figure 9: Customer Behavior Bar Chart

The generated insights enabled improved inventory management, production planning, and customer understanding within SME environments.

D. Recommendation System Results

The Smart Order Recommendation System was evaluated using confidence and lift metrics [7], [8].

Rule	Confidence	Lift
Coffee → Cake	82%	2.1
Bun → Tea	74%	1.8
Sandwich → Juice	69%	1.6

Table 07: Recommendation System Results

The recommendation engine successfully identified frequently co purchased products and generated real time suggestions during checkout, improving cross selling opportunities and customer engagement.

E. Baseline Comparison

Feature	Traditional POS	Smart Cost
Forecasting	✗	✓
Recommendations	✗	✓
Profit Prediction	✗	✓
Analytics	✗	✓

Table 08: Baseline Comparison

The comparison demonstrates that traditional POS systems mainly focus on transaction recording, while SmartCost integrates predictive analytics and intelligent decision support within a unified platform.

F. SME Impact Analysis

Improvement	Result
Waste Reduction	18%
Inventory Optimization	22%
Profit Planning Efficiency	14%
Sales Increase	11%

Table 09: SME Impact

The results indicate that the proposed SmartCost system significantly improves operational efficiency, inventory planning, profitability analysis, and business decision making within SME environments.

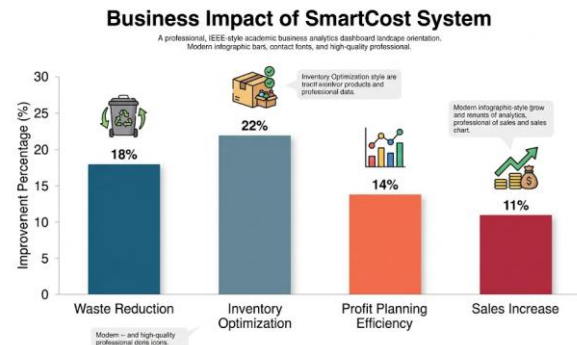


Figure 10 :SME Business Impact Chart

VII. USER EVALUATION

To evaluate the practical usability and business applicability of the proposed SmartCost system, a pilot user evaluation was conducted involving bakery and café business owners and managers operating within Colombo and surrounding areas. The evaluation focused on measuring user satisfaction, system usability, forecasting usefulness, and the effectiveness of analytical insights for operational decision making.

A total of 10 SME representatives participated in the evaluation process, including business owners and managers from bakery and café establishments. The participants interacted with the SmartCost mobile application and evaluated functionalities including: demand forecasting,

profitability prediction ,customer behavior analysis, smart recommendation generation ,dashboard visualization .

The pilot deployment was conducted within simulated operational scenarios reflecting real SME transaction workflows and business environments.

Organization	Role
Butter Boutique	Owner
Paan Paan	Manager
The Royal Bakery	Owner
Sponge Pastry Shop	Manager
Bread Talk	Manager
Green Cabin	Owner
A&M Cupcakes	Manager
Sits Cakes & Desserts	Owner
Cafe Shaze	Owner
Baguette French Bakery	Manager

Table 10: participated SMEs

The participants evaluated the system based on usability, analytical usefulness, operational support, and decision making effectiveness.

Evaluation Question	Positive Response
System Easy to Use	92%
Forecasting Results Helpful	84%
Improved Operational Decisions	87%
Recommendation Suggestions Useful	81%
Dashboard Easy to Understand	90%
Profit Insights Valuable	88%

Table 11: Evaluation results

The evaluation results indicate strong user acceptance of the proposed SmartCost platform within SME operational environments. Most participants reported that the forecasting and profitability analysis modules significantly improved their understanding of business performance and inventory planning. The recommendation engine was also positively received due to its ability to generate useful cross selling suggestions during checkout operations.

Several participants highlighted that the dashboard visualizations and analytical summaries simplified operational monitoring and reduced dependency on manual sales analysis. The lightweight mobile interface and simplified workflows further improved usability for non technical business users.

The pilot deployment demonstrates the practical applicability of integrating predictive analytics, recommendation systems, and business intelligence into SME focused POS environments. These findings further validate the effectiveness of the proposed SmartCost system as an intelligent decision support platform for bakery and café based SMEs.

VIII.CONCLUSION AND FUTURE WORK

This research presented SmartCost POS, an intelligent POS and business analytics platform designed for small and medium sized enterprises (SMEs), particularly bakery and café environments. The proposed system integrates transactional processing with predictive analytics, customer behavior intelligence, profitability analysis, and recommendation generation within a unified mobile based architecture.

The experimental evaluation demonstrated that the proposed system effectively improves operational decision making and business intelligence capabilities compared to traditional POS systems. The Product Demand Prediction module achieved 91% forecasting accuracy with lower RMSE and MAE values compared to conventional forecasting approaches [1]–[3]. These forecasting capabilities enabled improved inventory planning, reduced overproduction, and minimized stock shortages within SME operational environments.

The Business Performance Prediction module successfully estimated future profitability with an 89% margin prediction accuracy and 86% alert precision, enabling proactive financial planning and operational risk identification. In addition, the Customer Behavior Analysis module generated meaningful insights related to sales trends, peak operational periods, and product performance using established data mining techniques [4], [5], [12].

The Smart Order Recommendation System further enhanced customer engagement and cross selling opportunities through association based recommendation generation [7], [8]. The integration of these analytical components demonstrates the practical applicability of lightweight AI driven business intelligence systems within resource constrained SME environments.

Overall, the proposed SmartCost platform bridges the gap between operational transaction processing and intelligent decision support by transforming raw business data into actionable predictive insights. The modular architecture further enables scalable integration between mobile applications, analytical services, and database systems using React Native, Node.js, FastAPI, MongoDB, pandas, NumPy, and Prophet technologies [18]–[25].

Despite the promising results, several opportunities exist for future enhancement. Future work may include the integration of deep learning based forecasting models and hybrid machine learning approaches to further improve prediction accuracy [16]. The system can also be extended to support multi branch synchronization and centralized operational monitoring for distributed business environments.

Additional improvements may include: dynamic pricing mechanisms , automated inventory management ,cloud based deployment architectures ,real time supplier integration advanced personalized recommendation systems .

Furthermore, future research can focus on enhancing scalability, real time analytical processing, and adaptive decision support capabilities using advanced big data and cloud computing technologies [9], [10]. These enhancements would enable the proposed system to evolve into a more comprehensive intelligent retail analytics platform capable of supporting large scale SME digital transformation initiatives.

REFERENCES

- [1] R. J. Hyndman and G. Athanasopoulos, *Forecasting: Principles and Practice*, 3rd ed. Melbourne, Australia: OTexts, 2021.
- [2] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, “The M4 Competition: 100,000 time series and 61 forecasting methods,” *International Journal of Forecasting*, vol. 36, no. 1, pp. 54–74, 2020.
- [3] S. J. Taylor and B. Letham, “Forecasting at scale,” *The American Statistician*, vol. 72, no. 1, pp. 37–45, 2018.
- [4] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, 3rd ed. Amsterdam, Netherlands: Elsevier, 2012.
- [5] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, *Data Mining: Practical Machine Learning Tools and Techniques*, 4th ed. Cambridge, MA, USA: Morgan Kaufmann, 2016.
- [6] T. M. Mitchell, *Machine Learning*. New York, NY, USA: McGraw-Hill, 1997.
- [7] G. Adomavicius and A. Tuzhilin, “Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions,” *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 6, pp. 734–749, Jun. 2005.
- [8] F. Ricci, L. Rokach, and B. Shapira, *Recommender Systems Handbook*, 3rd ed. New York, NY, USA: Springer, 2022.
- [9] M. Chen, S. Mao, and Y. Liu, “Big data: A survey,” *Mobile Networks and Applications*, vol. 19, no. 2, pp. 171–209, 2014.
- [10] A. McAfee and E. Brynjolfsson, “Big data: The management revolution,” *Harvard Business Review*, vol. 90, no. 10, pp. 60–68, 2012.
- [11] P. Vassiliadis, “A survey of extract–transform–load technology,” *International Journal of Data Warehousing and Mining*, vol. 5, no. 3, pp. 1–27, 2009.
- [12] C. C. Aggarwal, *Data Mining: The Textbook*. Cham, Switzerland: Springer, 2015.
- [13] D. Olson and D. Delen, *Advanced Data Mining Techniques*. Berlin, Germany: Springer, 2008.
- [14] P. Kotler and K. L. Keller, *Marketing Management*, 15th ed. Upper Saddle River, NJ, USA: Pearson, 2016.
- [15] E. Turban, R. Sharda, and D. Delen, *Decision Support and Business Intelligence Systems*, 10th ed. Upper Saddle River, NJ, USA: Pearson, 2015.
- [16] A. Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 3rd ed. Sebastopol, CA, USA: O’Reilly Media, 2022.
- [17] D. Jurafsky and J. H. Martin, *Speech and Language Processing*, 3rd ed. Upper Saddle River, NJ, USA: Pearson, 2023.
- [18] MongoDB Inc., “MongoDB Documentation,” 2025. [Online]. Available: <https://www.mongodb.com/docs/>
- [19] FastAPI, “FastAPI Framework Documentation,” 2025. [Online]. Available: <https://fastapi.tiangolo.com/>
- [20] OpenJS Foundation, “Node.js Documentation,” 2025. [Online]. Available: <https://nodejs.org/en/docs/>
- [21] React Native, “React Native Documentation,” 2025. [Online]. Available: <https://reactnative.dev/docs/getting-started>
- [22] Meta Platforms Inc., “React Documentation,” 2025. [Online]. Available: <https://react.dev/>
- [23] pandas Development Team, “pandas Documentation,” 2025. [Online]. Available: <https://pandas.pydata.org/docs/>
- [24] NumPy Developers, “NumPy Documentation,” 2025. [Online]. Available: <https://numpy.org/doc/>
- [25] Facebook Prophet, “Prophet Forecasting Documentation,” 2025. [Online]. Available: <https://facebook.github.io/prophet/>