International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Smart Inventory Monitoring System (SIMS)

Mrs. Vinothini T[1], JayaSri R[2], Muhseena B[3], Nividha V[4]

[1]Assistant Professor, B.E Computer Science and Engineering, Adhiyamaan College Of Engineering
[2,3,4]Student, B.E Computer Science and Engineering, Adhiyamaan College Of Engineering

Abstract
Small and medium-scale retail businesses often rely on manual record-keeping methods such as handwritten ledgers and basic billing software, leading to stock discrepancies, inaccurate profit calculation, poor credit tracking, and lack of financial transparency. The absence of real-time monitoring, centralized data storage, and secure multi-user access further limits operational efficiency and scalability.To address these challenges, the Smart Inventory Monitoring System (SIMS) is proposed as a cloud-based, multi-tenant Software-as-a-Service (SaaS) retail management platform. SIMS integrates inventory tracking, automated billing, customer credit (Udhaar) management, supplier payable monitoring, expense tracking, and business analytics into a unified digital ecosystem. The mobile application is developed using React Native, while the web-based admin panel is built with Next.js. The backend infrastructure is powered by Supabase and PostgreSQL, incorporating secure authentication, role-based access control, and Row Level Security (RLS) for tenant-level data isolation.The system enables real-time stock synchronization, accurate profit computation, subscription-based feature control, and scalable multi-company management. As a result, SIMS enhances operational accuracy, financial clarity, data security, and strategic decision-making for modern retail enterprises.

Keywords: Smart Inventory Monitoring System (SIMS), Software-as-a-Service (SaaS), Multi-Tenant Architecture, Retail Management System, Cloud-Based Inventory Control, Role-Based Access Control (RBAC).

1. Introduction

Small and medium-scale retail businesses traditionally rely on manual methods such as handwritten stock registers, physical ledgers, and standalone billing systems to manage inventory and financial records. These approaches often result in stock discrepancies, delayed updates, inaccurate profit estimation, and difficulty in tracking customer credit and supplier payments. The absence of integrated systems limits operational visibility and increases the risk of data errors, financial miscalculations, and inefficiencies in decision-making.

With increasing competition and growing transaction volumes, automation has become essential for modern retail operations. Automated inventory tracking, real-time billing integration, expense monitoring, and structured ledger management significantly reduce human error while improving accuracy and operational efficiency. Automation also enhances transparency, accountability, and business intelligence through analytical insights.

Furthermore, cloud-based solutions address limitations associated with local storage and standalone systems. Cloud infrastructure ensures real-time data synchronization, secure access control, centralized

management, and scalability without requiring heavy infrastructure investment. It also reduces risks related to data loss and enables multi-user access with proper authentication mechanisms.

To address these challenges, the Smart Inventory Monitoring System (SIMS) is proposed as a cloud-based, multi-tenant SaaS retail management platform. SIMS integrates inventory control, automated billing, financial tracking, subscription-based feature management, and real-time analytics into a unified mobile and web ecosystem, providing a secure, scalable, and data-driven solution for modern retail enterprises.

2. Problem Statement

Small and medium-scale retail businesses continue to rely on manual inventory management methods such as handwritten stock registers, separate billing books, and basic spreadsheet tools. These traditional systems lead to frequent stock mismatches due to delayed or inaccurate updates, resulting in overstocking, understocking, and loss of sales opportunities. Manual billing processes increase the risk of calculation errors and inconsistencies in transaction records. Furthermore, the absence of real-time visibility prevents business owners from accurately monitoring inventory levels, sales performance, and outstanding payments. Financial tracking is often incomplete, as operational expenses are not systematically integrated with revenue data, leading to inaccurate profit estimation and poor decision-making. These limitations highlight the need for an automated, integrated, and cloud-based retail management solution.

3. Objectives Of Study

The primary objectives of the Smart Inventory Monitoring System (SIMS) are:

- To automate inventory management and eliminate manual stock recording errors.
- To enable real-time stock tracking with automated updates after every sales and purchase transaction.
- To implement low-stock and expiry alert mechanisms for proactive inventory control.
- To automate billing processes with integrated profit calculation based on cost and selling price.
- To improve financial accuracy by integrating expense tracking with revenue and profit analysis.
- To digitize customer credit (Udhaar) and supplier payable management through structured ledger systems.
- To develop a cloud-based, secure, and scalable multi-tenant SaaS platform for retail businesses.
- To implement role-based access control and tenant-level data isolation for secure multi-user operations.

4. Literature Review

Several research works have explored inventory management systems, SaaS-based enterprise platforms, and secure multi-tenant architectures relevant to retail automation.

Sharma et al. proposed a cloud-based inventory management system for small and medium enterprises emphasizing real-time updates and centralized data storage. Their study highlights the importance of remote accessibility and reduced manual errors, aligning with the cloud infrastructure adopted in SIMS. However, their work primarily focuses on inventory control without integrating financial analytics and subscription-based feature management.

Gupta and Rao emphasized profit analytics in retail ERP systems, demonstrating the need for expense-adjusted net profit computation rather than revenue-only reporting. While their work focuses on financial analytics, SIMS extends this concept by integrating profit calculation directly with inventory, billing, and expense modules in a unified system.

Verma and Kulkarni discussed design principles of multi-tenant SaaS ERP systems, including tenant isolation and subscription lifecycle management. SIMS adopts a similar shared database multi-tenant model but enhances security through Row Level Security (RLS) policies and role-based access control.
Lee et al. examined role-based access control mechanisms for cloud applications, emphasizing database-level security enforcement. SIMS implements these concepts using Supabase authentication and PostgreSQL RLS to ensure strict tenant-level data isolation.
Compared to existing systems, SIMS integrates inventory management, financial tracking, subscription control, and real-time synchronization into a single scalable SaaS ecosystem, offering a more comprehensive and secure retail management solution.

5. Proposed System Architecture

The Smart Inventory Monitoring System (SIMS) follows a cloud-based, multi-tenant SaaS architecture designed to integrate retail operations, administrative control, and secure data management within a unified ecosystem. The system adopts a layered structure consisting of client applications, backend services, and a centralized database with tenant-level isolation.

5.1. Overall System Design

SIMS operates on a shared database, shared schema multi-tenant model. Multiple businesses use the same cloud infrastructure while maintaining strict logical data isolation through role-based access control and Row Level Security (RLS). The architecture ensures scalability, real-time synchronization, and centralized management.

5.2. Mobile Application (Retail Operations Layer)

The mobile application, developed using React Native, serves store owners and staff. It manages inventory, sales billing, purchase entries, customer credit (Udhaar), supplier records, expense tracking, and analytics. The app communicates securely with the backend via HTTPS APIs and supports real-time data updates.

5.3. Admin Panel (System Administration Layer)

The web-based admin panel, built using Next.js, enables super administrators to manage companies, subscription plans, system modules, and user roles. It controls feature-level access based on subscription mapping and ensures centralized SaaS governance.

5.4. Backend Services (Application Layer)

Supabase acts as the Backend-as-a-Service (BaaS) provider. It handles authentication (OTP-based login), API processing, real-time synchronization, business logic execution, and role validation.

5.5. Database Layer

A PostgreSQL database stores operational and system-level data. Each record includes a company identifier to maintain tenant isolation. Row Level Security (RLS) policies enforce strict cross-company data protection.
This architecture ensures security, scalability, automation, and efficient retail management within a unified cloud environment.

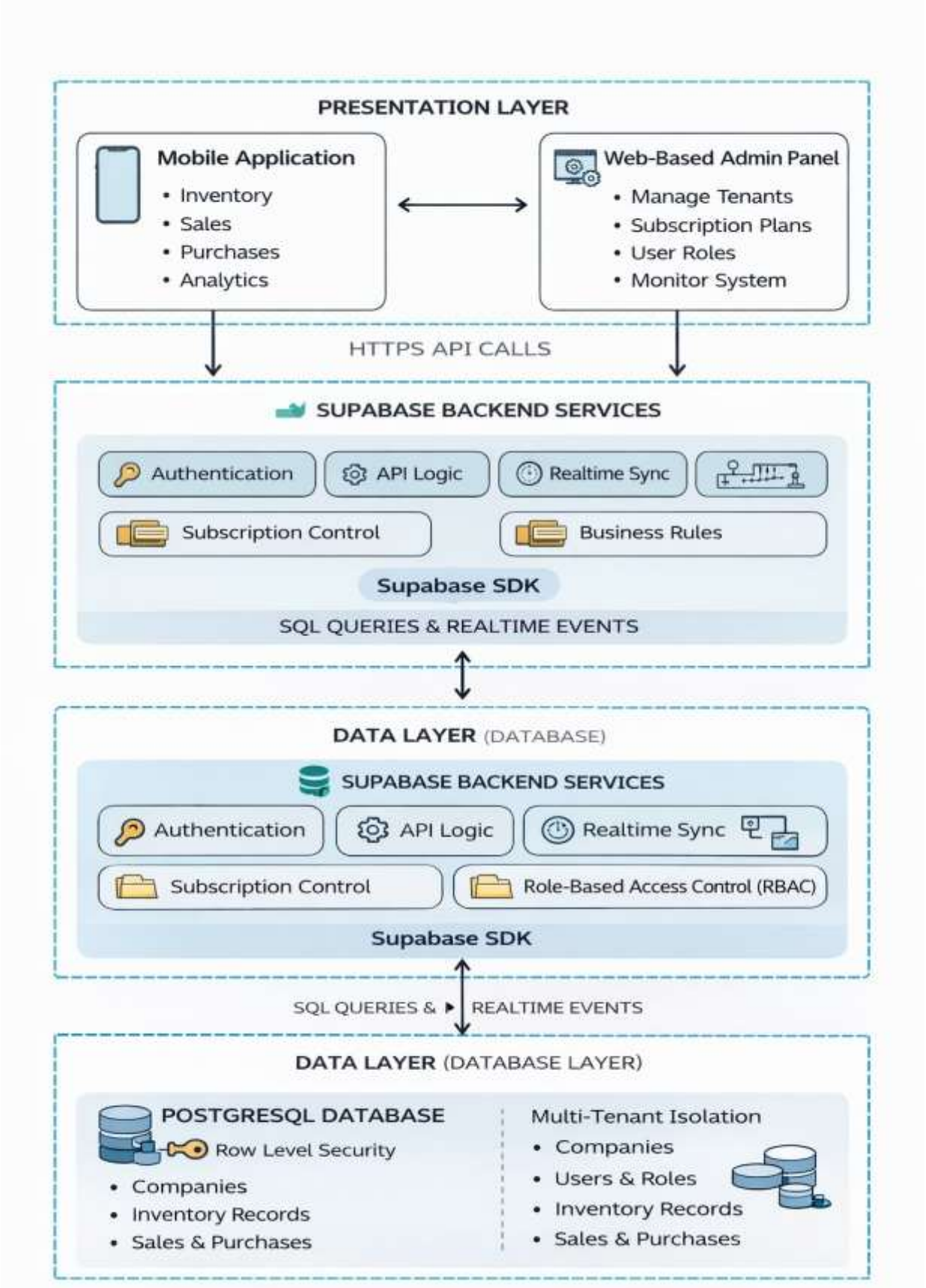

Figure 1: Overall System Architecture

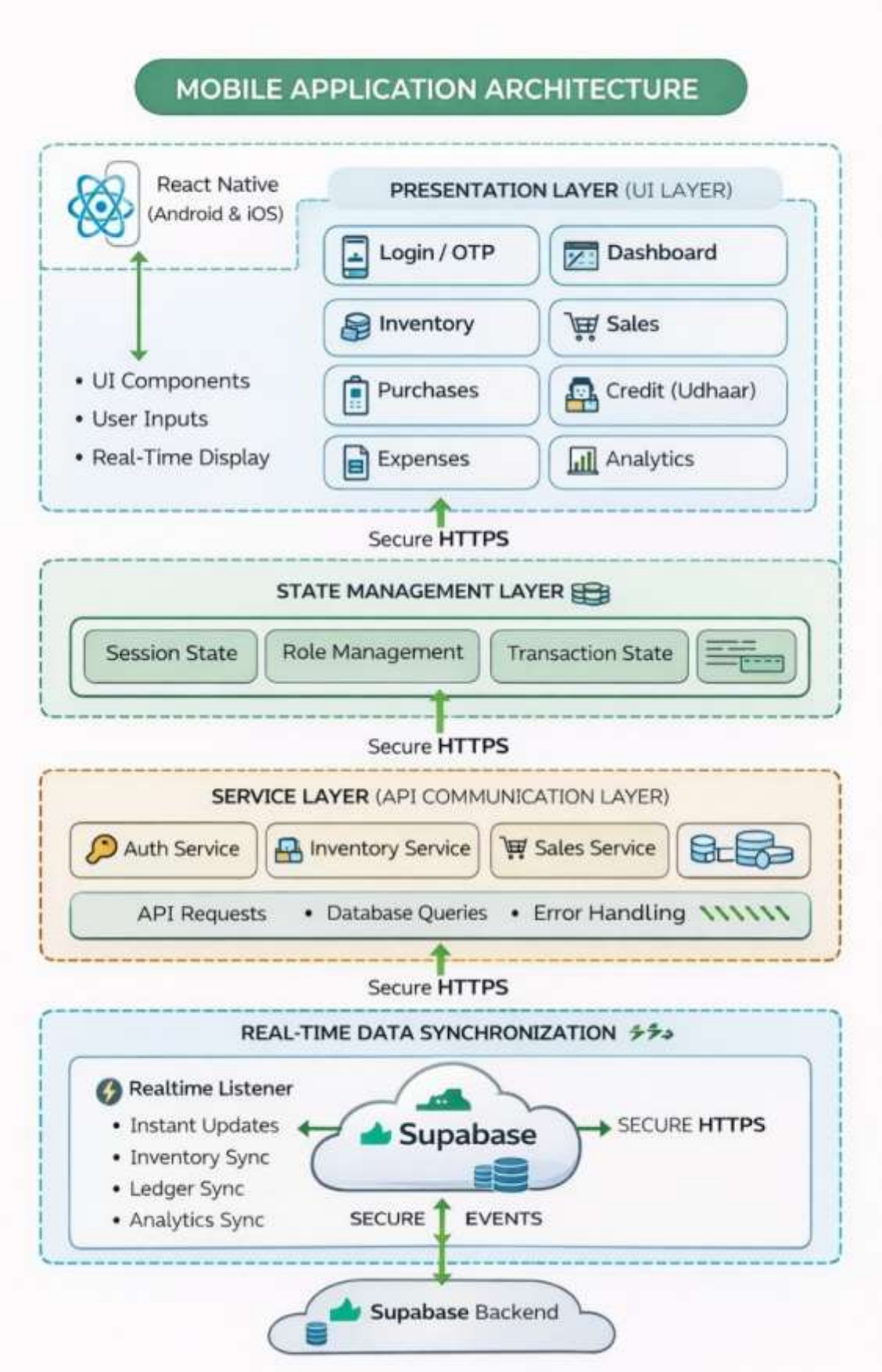

Figure 2: Mobile System Architecture

6. Systems Modules Description

6.1. Inventory Management Module

The Inventory Management Module handles product creation, category organization, stock tracking, and low-stock monitoring. Stock levels are automatically updated after every sales and purchase transaction, ensuring real-time accuracy. The system prevents negative stock entries and provides threshold-based alerts to avoid stockouts. All inventory data is secured using tenant-level isolation mechanisms.

6.2. Sales & Purchase Processing Module

This module manages billing operations and procurement activities. During sales, the system validates stock availability, calculates total amount and profit, deducts inventory automatically, and records the transaction atomically. In purchase processing, stock quantities are increased, and supplier records are updated. Real-time synchronization ensures immediate reflection of transaction updates across dashboards.

6.3. Customer & Supplier Ledger Module

The ledger module digitizes customer credit (Udhaar) and supplier payable tracking. It maintains structured records of outstanding balances, payment history, and transaction references. This ensures financial transparency, reduces disputes, and improves accountability.

6.4. Admin Panel Module

The Admin Panel provides centralized system control. It allows super administrators to manage companies, configure subscription plans, map modules, and monitor system-level operations. Role-based access control ensures secure administrative governance.

6.5. Subscription Management Module

This module enables SaaS-based feature control. Businesses are assigned subscription plans that determine accessible modules. Plan-to-module mapping dynamically activates or restricts features, ensuring scalability, monetization, and flexible system expansion.

7. Technologies Used

- **Frontend (Mobile Application):** React Native (Expo Framework) – Cross-platform mobile development for Android and iOS.
- **Frontend (Admin Panel):** Next.js (App Router with TypeScript) – Web-based administrative interface with server-side rendering.
- **Backend Services:** Supabase (Backend-as-a-Service) – Authentication, API management, and real-time data synchronization.
- **Database:** PostgreSQL – Relational database with multi-tenant architecture and Row Level Security (RLS).
- **Deployment & Hosting:** Cloud Platform (Supabase Cloud and Vercel) – Scalable cloud hosting and serverless deployment.

8. Implementation Methodology

The implementation of the Smart Inventory Monitoring System (SIMS) was carried out using a structured and modular development approach to ensure scalability, security, and maintainability.

8.1. Application Setup

The mobile application was developed using React Native with the Expo framework to support both Android and iOS platforms through a single codebase. The admin panel was built using Next.js with server-

side rendering capabilities for secure and efficient administrative operations. Project directories were organized into modular components such as screens, services, navigation, and state management to maintain clean architecture. Environment variables were configured to securely connect the applications to backend services without exposing sensitive credentials.

8.2. Database Configuration

The backend database was implemented using PostgreSQL provided by Supabase. Core system-level and tenant-level tables were created to manage companies, users, products, transactions, subscription plans, and ledgers. Proper relational modeling, primary and foreign key constraints, indexing, and validation rules were applied to ensure data integrity and performance optimization.

8.3. Multi-Tenant Implementation

SIMS follows a shared database, shared schema multi-tenant architecture. Each operational table includes a company_id field to logically separate business data. All queries filter records based on the authenticated user's company. This approach allows multiple businesses to operate on the same infrastructure while maintaining strict data isolation.

8.4. Security Measures (RLS & RBAC)

Security is enforced through Supabase Authentication with OTP-based login and JWT session tokens. Role-Based Access Control (RBAC) defines permissions for super_admin, company_admin, and staff roles. Row Level Security (RLS) policies are implemented at the database level to restrict data access to authorized tenants only. This layered security model ensures protection against unauthorized access and cross-company data exposure.

9. Results and Discussion

The Smart Inventory Monitoring System (SIMS) was successfully implemented and tested across mobile and web platforms. The system demonstrated stable performance in managing inventory operations, billing processes, ledger tracking, and subscription control within a multi-tenant SaaS environment.

System Performance Highlights

- Real-time stock updates occur immediately after each sales or purchase transaction.
- Atomic transaction handling prevents partial updates and maintains data consistency.
- Multi-tenant isolation ensures secure data access without cross-company exposure.
- Optimized database queries and indexing reduce response latency.
- Cloud-based infrastructure ensures high availability and scalability.

Benefits Observed

- Significant reduction in manual errors related to stock and billing calculations.
- Improved financial accuracy through automated profit and expense tracking.
- Faster decision-making due to real-time analytics and dashboard insights.
- Simplified customer credit (Udhaar) and supplier payable management.
- Centralized control through the admin panel for subscription and company management.

Real-Time Synchronization Results

Supabase real-time services ensured immediate reflection of changes across devices. When transactions were performed on the mobile application, corresponding updates were instantly visible in dashboards and reports. This confirmed effective synchronization, data consistency, and reliable cloud-based operation. Overall, the system achieved its objective of providing a secure, scalable, and automated retail inventory management solution.

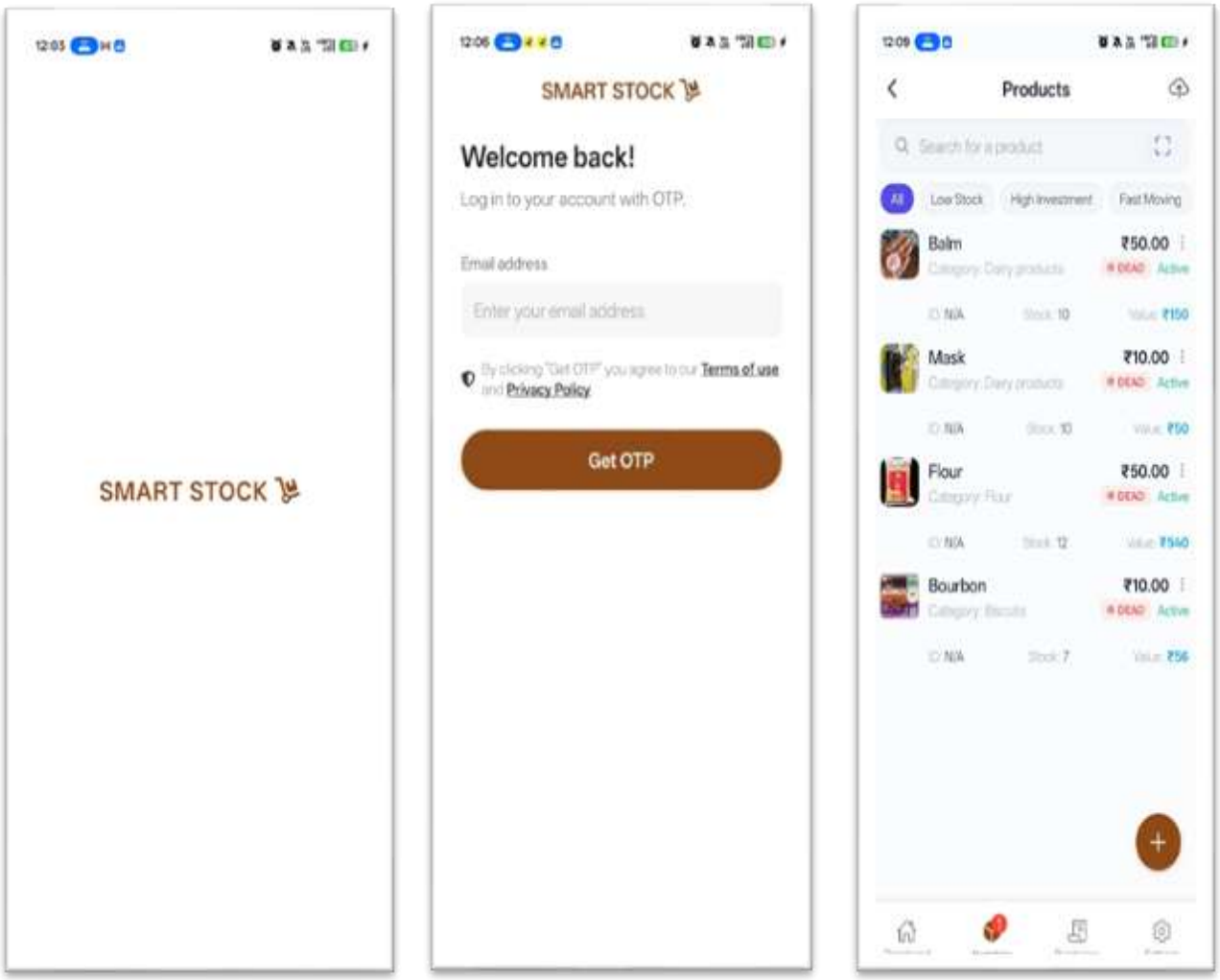

Figure 3: Result Screenshots

10. Advantages Of Proposed System

- **Real-Time Stock Updates:** Inventory levels are automatically updated after every sales and purchase transaction, ensuring accurate and up-to-date stock information.
- **Reduced Manual Errors:** Automated billing, stock deduction, and profit calculation minimize human errors associated with manual record-keeping.
- **Secure Multi-Tenant Data Isolation:** Implementation of Row Level Security (RLS) and Role-Based Access Control (RBAC) ensures strict separation of company data within a shared database.
- **Scalable SaaS Architecture:** The cloud-based shared infrastructure supports multiple businesses simultaneously while maintaining performance and security.
- **Improved Financial Accuracy:** Integrated expense tracking and profit computation provide precise financial insights.
- **Centralized Administrative Control:** The admin panel enables efficient management of companies, subscription plans, and feature access.

11. Limitations

- **Internet Dependency:** Since SIMS is a cloud-based system, stable internet connectivity is required

for real-time synchronization and data access.

- **Initial Learning Curve:** Retail users who are accustomed to manual record-keeping may require basic training to adapt to the digital interface and system workflow.
- **Limited Offline Support:** The current implementation primarily operates online, with minimal offline functionality, which may affect operations during network interruptions.

12. Future Enhancements

- **AI-Based Demand Forecasting:** Integration of machine learning models to analyze historical sales data and predict future demand trends, enabling smarter inventory planning and reduced stockouts.
- **Barcode / QR Code Scanning:** Implementation of barcode and QR scanning functionality for faster product identification, billing efficiency, and improved inventory accuracy.
- **GST Integration:** Incorporation of automated GST calculation, tax report generation, and compliance-ready invoice formatting to support regulatory requirements.
- **Advanced Analytics Dashboard:** Development of enhanced analytics with visual reports, sales trends, category-wise performance, profit margins, and predictive insights for better decision-making.
- **Offline Mode with Sync Mechanism:** Addition of local data caching with automatic synchronization once internet connectivity is restored.

13. Conclusion

The Smart Inventory Monitoring System (SIMS) addresses critical challenges faced by small and medium-scale retail businesses, including manual stock handling, inaccurate billing, lack of real-time visibility, and inefficient financial tracking. Traditional record-keeping methods often result in stock mismatches, delayed updates, and poor decision-making due to fragmented data management.

The proposed cloud-based, multi-tenant SaaS architecture successfully integrates inventory control, billing automation, ledger management, subscription handling, and analytics into a unified platform. Through real-time synchronization, secure tenant-level data isolation, and role-based access control, the system ensures operational accuracy, scalability, and data security. Automated profit calculation and structured financial tracking significantly enhance transparency and reliability.

Overall, SIMS contributes a scalable and secure digital transformation solution tailored for retail enterprises. By combining automation, cloud infrastructure, and multi-tenant design principles, the system demonstrates an effective approach to modernizing retail inventory management while supporting sustainable business growth.

14. References

[1] S. Sharma and R. Patel, "Cloud-Based Inventory Management System for Small and Medium Enterprises," *International Journal of Computer Applications*, vol. 178, no. 12, pp. 25–30, 2019.

[2] A. Gupta and P. Rao, "Financial Analytics Integration in Retail ERP Systems," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 4, pp. 210–217, 2020.

[3] R. Verma and S. Kulkarni, "Design and Implementation of Multi-Tenant SaaS Architectures," *IEEE Access*, vol. 8, pp. 145623–145635, 2020.

[4] J. Lee, H. Kim, and M. Park, "Role-Based Access Control Mechanisms in Cloud Computing Environments," *Future Generation Computer Systems*, vol. 87, pp. 786–798, 2018.
[5] K. Zhang and Y. Li, "Secure Data Isolation Techniques for Multi-Tenant Databases in Cloud Applications," *Journal of Cloud Computing*, vol. 9, no. 1, pp. 1–15, 2020.
[6] P. Singh and D. Kumar, "Real-Time Data Synchronization in Cloud-Based Retail Systems," *International Journal of Information Management*, vol. 54, pp. 102–110, 2020.