

Inventory Analyzer: A Comprehensive Solution for Inventory Management

Mrs. M. Santhi^{*1} | Ch. Sai Ashok² | A. Sukeerth Jonathan³ | D. Abhiram⁴ | K. Kalyan⁵ | E. Lohith Kumar⁶

^{*1,2,3,4,5&6}Department of Computer Science and Engineering, Narayana Engineering College, Nellore, Andhra Pradesh, India

Abstract: *The Inventory Analyzer represents an advanced Android app with the goal of transforming inventory management. It targets suppliers, dealers, and users to address the limitations of conventional systems including inaccuracies and outdated data availability. By utilizing Java and Kotlin in its development and integrating an SQL database for backend storage, the Inventory Analyzer guarantees robust performance and scalability. The app is divided into specific modules catering to individual user needs: suppliers can add and oversee dealers, products, and orders; dealers can adjust product prices, review user requests, and monitor their sales history; users can explore dealers by location, request products, view order past records, and offer feedback. A crucial aspect of the Inventory Analyzer is its real-time data processing mechanism ensuring that all users have access to the most up-to-date information available. During its development phase, various testing methods like unit testing, integration testing, black-box testing, and white-box testing were employed to verify the reliability, functionality, and overall quality of the application. The user interface has been designed for intuitive use by individuals at all expertise levels in order to simplify inventory management duties. Future enhancements planned for the Inventory Analyzer comprise advanced analytics for inventory projection, integration with other business systems, upgraded security elements, as well as support for multiple languages. In summary, the Inventory Analyzer presents a notable leap forward from traditional inventory systems by offering a modern, effective, and scalable solution which not only streamlines inventory operations but also enhances overall business efficiency leading to heightened user satisfaction.*

Keywords: *Mobile application; Android, Inventory; Product Assignment; Order History; Request Handling; Profile Management; Access Control*

I. INTRODUCTION

Effective inventory management is crucial for businesses to maintain optimal stock levels, minimize operational costs, and improve customer satisfaction. Traditional inventory systems often face issues with inaccuracies, delays, and lack of real-time data, resulting in overstocking, stockouts, and missed opportunities. In response to these challenges, we've created the Inventory, an Android-based app crafted to streamline and enhance inventory management processes for suppliers, dealers, and users. The Inventory Analyzer aims to tackle the limitations of conventional inventory systems by providing a sturdy yet user-friendly platform that supports comprehensive inventory operations. Suppliers can manage their dealer network, add products, assign products, and keep track of orders and requests using the application. Dealers can efficiently handle product pricing, view user purchase requests, and track their sales history. Users enjoy a simple interface to explore dealers, place orders, and share feedback for a smooth and engaging experience. Developed in Android Studio using Java and Kotlin while utilizing an SQL database for backend storage, the Inventory Analyzer ensures compatibility with various Android devices. The development process includes requirements gathering, system design, module development, integration, and thorough testing to guarantee reliability and performance. This document elaborates on the design and execution of the Inventory Analyzer app highlighting its essential features and benefits for inventory management. By

incorporating modern technologies and adhering to software development best practices, the Inventory Analyzer offers a scalable and effective solution for businesses seeking to enhance their inventory operations. The document also delves into potential future upgrades like advanced analytics and broader system integrations to expand the application's capabilities further.

1.1. PROBLEM STATEMENT

The issue stems from the absence of a central platform for supplier-user interactions which poses notable obstacles. communication channels being separate lead to inefficiencies, delays, and a lack of transparency in procurement procedures. Manual approaches for browsing products, submitting requests, and tracking orders worsen these problems. Thus, an urgent requirement exists for an Android app that simplifies communication, improves transparency, and offers a user-friendly interface for smooth transactions among suppliers, dealers, and users.

1.2. OBJECTIVE OF THE PAPER

The project's goal is to create an Android app that helps, dealers, and users communicate and do business more efficiently. It aims to simplify the buying process, increase transparency, and offer a friendly interface for easy interaction, improving the overall experience for everyone involved. With all product information, orders, and requests centralized in a single platform, the app intends to streamline procurement, promote transparency, and make it user-friendly. This will cut down on delays, build trust, and guarantee a positive buying and selling experience for all.

II. EXISTING WORK

Inventory management systems play a vital role in maintaining optimal stock levels, cutting costs, and boosting customer satisfaction. A of systems and technologies have been to tackle the challenges of inventory management. realm of inventory management work spans from traditional manual techniques to sophisticated automated systems, each having its own merits and drawbacks.

2.1 TRADITIONAL INVENTORY MANAGEMENT SYSTEMS

In the realm Traditional Inventory Management Systems, stock levels are tracked manually using spreadsheets or paper records. These methods are prone to errors, time-consuming, and lack real-time visibility into data. While they are cost-effective and simple to implement, struggle to keep pace with growing inventory size and complexity.

2.2 BARCODE AND RFID SYSTEMS

Barcodes & RFID Systems have been embraced by many businesses to improve accuracy and efficiency. These technologies automate the inventory tracking process and provide precise data. Barcodes & RFID tags are scanned for updating records, reducing errors and speeding up operations. Nevertheless, these systems demand substantial upfront investments in hardware and might lag in real-time data processing.

2.3 ENTERPRISE RESOURCE PLANNING (ERP)

Enterprise Resource Planning (ERP) Systems amalgamate various business processes, including inventory management, onto a unified platform. These systems offer comprehensive functionality like real-time inventory tracking, automated reorder points, and integration with sales & procurement processes. Examples of popular ERP solutions comprise SAP, Oracle, & Microsoft Dynamics. Despite being robust & feature-rich, ERP systems can be pricey to implement & maintain, restricting their accessibility for small-medium enterprises (SMEs).

2.4 CLOUD BASED INVENTORY MANAGEMENT SYSTEMS

Cloud-Based Inventory Management Systems have emerged due to innovations in cloud computing. They provide real-time data access via the cloud with scalability & reduced IT overheads. Notable examples include TradeGecko, Zoho Inventory, & Unleashed. Cloud-based solutions offer flexibility and can be accessed from any location with internet connectivity but might raise concerns regarding data security.

2.5 MOBILE INVENTORY MANAGEMENT APPLICATIONS

Mobile Inventory Management Applications have gained traction with the widespread use of smartphones. These apps enable users to access inventory data on-the-go for managing stock levels & placing orders from mobile devices. Popular apps include Stockpile, Sortly, & Inventory Now. While convenient & accessible on mobile devices platforms lack the extensive features of full-fledged ERP systems.

2.6 LIMITATIONS OF EXISTING SYSTEMS

Despite technological advancements in inventory management technology existent systems still grapple with challenges like limited real-time data updates leading to discrepancies in stock levels; complex integration with other business processes; user-friendliness issues for non-technical users; high implementation & maintenance costs hindering SME accessibility.

2.7 THE INVENTORY ANALYZER APPROACH

The Inventory Analyzer application endeavors to tackle these limitations by offering a mobile-friendly inventory management app developing using Java & Kotlin technologies with an SQL database backend storage system. User-friendly features cater to suppliers, dealers & users facilitating real-time updates and seamless integration across various inventory tasks.

III. PROPOSED WORK

The Inventory Analyzer project aims to develop and launch a mobile-based inventory management system designed for suppliers, dealers, and users. The objective is to create a comprehensive and user-friendly application that tackles the drawbacks of traditional inventory systems by utilizing modern technology for real-time data, seamless integration, and improved user experience.

3.1 REQUIREMENTS ANALYSIS AND SYSTEM DESIGN

Goals: Clearly outline the requirements and scope of the project to ensure meeting all stakeholder.

Tasks:

- Conduct stakeholder interviews and surveys to gather detailed requirements.
- Create detailed use cases and user stories.
- Develop high-level system architecture, database schema, and user interface designs.

3.2. DEVELOPMENT OF CORE MODULES

Supplier Module:

- Features: Add and handle dealers, add products, assign products to dealers, view user details, and order history.
- Tasks: Implement dealer management, product management, and order management functionalities.

Dealer Module:

- Features: Secure login, profile management, view and set product prices, view user purchase requests, manage order history.
- Tasks: Develop dealer authentication, profile update, product pricing, and order handling features.

User Module:

- Features: User registration and login, browse dealers by location, request to purchase products, view order history, provide feedback.
- Tasks: Implement user registration, dealer browsing, purchase requests, order history viewing, and feedback submission functionalities.

3.3. INTEGRATION & DATA MANAGEMENT

Goals: Ensure smooth interaction & data flow between supplier-dealer-user modules.

Tasks:

- Develop & implement RESTful APIs for frontend-backend communication.
- Ensure data consistency & integrity through proper validation & error handling.
- Set up & configure the SQL database for efficient data storage & retrieval.

3.4. USER INTERFACE & EXPERIENCE ENHANCEMENT

Goals: Create an intuitive & responsive interface for all users.

Tasks:

- Design & implement responsive interfaces for suppliers, dealers & users.
- Ensure accessibility & easy navigation of the application.
- Include real-time updates for inventory levels, order status & user requests.

3.5. TESTING

Goals: Validate functionality performance security of the application.

Tasks:

- Conduct unit testing for components to ensure correct functioning.
- Perform integration testing for module interactions validation along with whitebox and blackbox testing.

IV. EXPERIMENTAL RESULTS

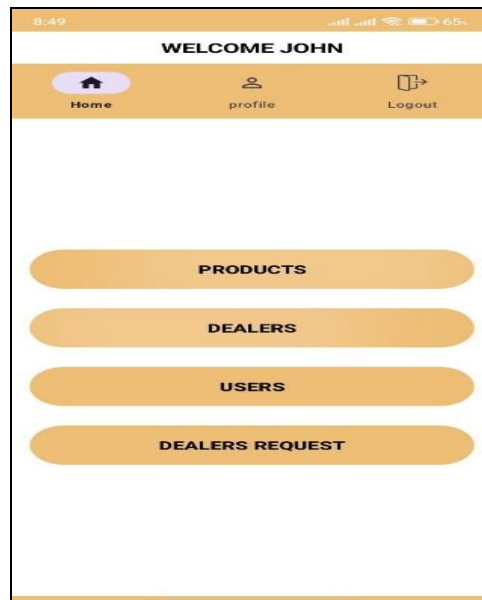


Fig: 4.1 Supplier Home

4.1 SUPPLIER HOME

The supplier homepage offers a snapshot of essential functions and details pertinent to the supplier. The screen grants swift entry to five primary sections via a grid of icons or buttons bearing labels.

Products: Permits the supplier to see, include, and modify product details.

Dealers: Empowers the supplier to handle dealers, encompassing adding, deleting, and checking dealer particulars.

Users: Furnishes the supplier with choices to view user information or filter by location (as indicated by the "%" symbol in the image).

Dealer Requests: Authorizes the supplier to scrutinize and oversee requests put forth by dealers (e.g., requests for new products or alterations in pricing).

Profile: Grants the supplier access to and ability to adjust their profile details.

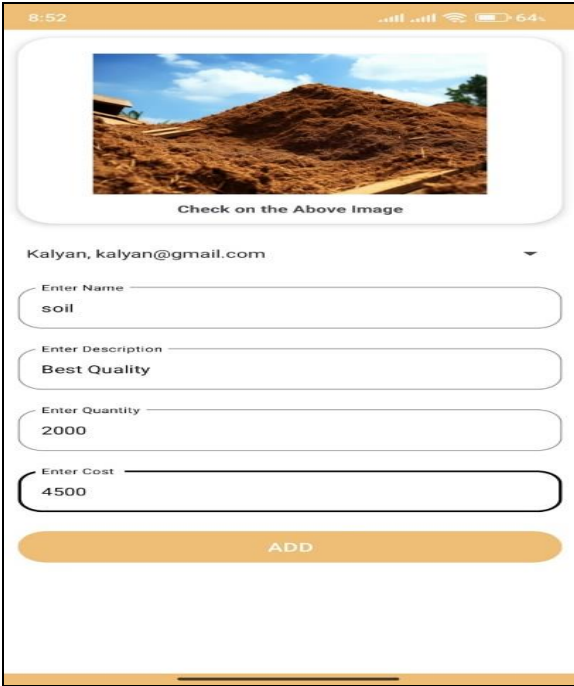


Fig: 4.2 Add Product with Dealer

4.2 ADD PRODUCT WITH DEALER

The screen allows a supplier to introduce a new product and link it to a particular dealer. At the top, you'll see the supplier's name for easy reference. The form consists of fields to input the product details, such as:

- **Name:** Type in the product name here.
- **Description:** Add a description of the product in this field.
- **Quantity:** Specify the initial quantity of the product using this number field.
- **Cost:** Enter the cost price of the product in this number field.

There's a "Check on the Above Image" button available, which is used for adding an image. Towards the bottom, there's a large "ADD" button that kicks off the process of adding the new product to the app's database and linking it to the chosen dealer.



Fig: 4.3 Dealer List (Supplier View)

4.3 DEALER LIST (Supplier View)

The screen shows the supplier a rundown of all dealers they oversee in the Inventory app.

- Name: Shows dealer names.
- Email: Shows dealer email addresses.
- Number: Shows dealer phone numbers.
- City: Shows dealer cities.

Feedback: Check out user feedback for that dealer.

Delete: Hit this button (with a trash can) to kick the dealer off the list.

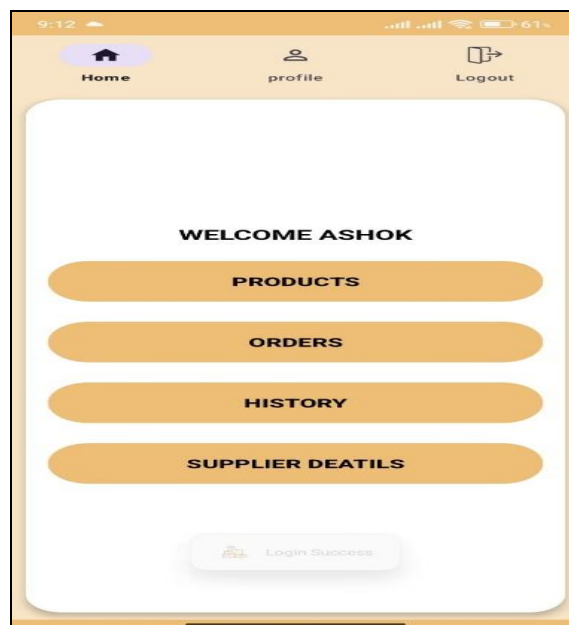


Fig: 4.4 Dealer Home

4.4 DEALER HOME

The home screen for dealers presents a snapshot of essential functions and information relevant to them.

Quick access to four main sections through a grid of icons or buttons labelled.

Products: Permits the dealer to see products assigned by the supplier & potentially set selling prices.

Orders: Enables viewing order history & managing sales tasks.

History: Provides options for reviewing past interactions or sales data.

Also included is "Supplier Details" for accessing contact information & relevant supplier details.

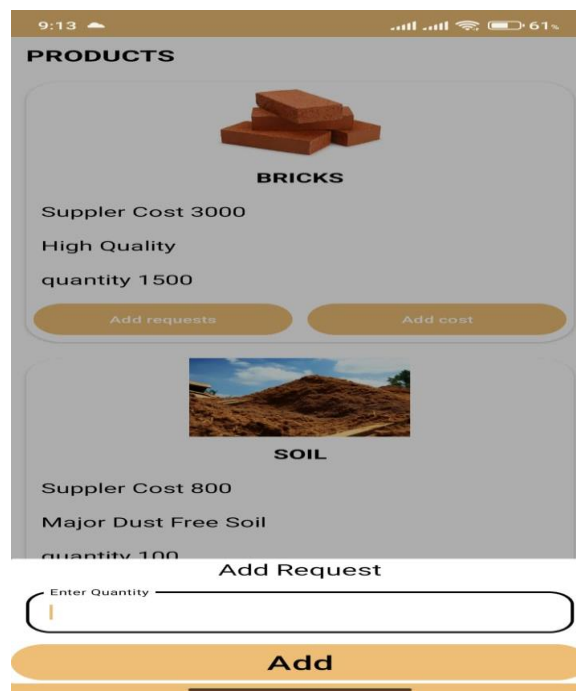


Fig: 4.5 Dealer Request for Product

4.5 DEALER REQUEST FOR PRODUCT

The dealer can easily ask the supplier for more products using this screen.

Product Information:

Product Image: Shows a picture of the product for the dealer to see.

Product Name: Clearly displays the name of the product.

Supplier Cost: Reveals how much the dealer paid for the product from the supplier before.

Request Quantity: Lets the dealer type in how many more of the product they want.

Submit Button: Click this button to send the request to the supplier.

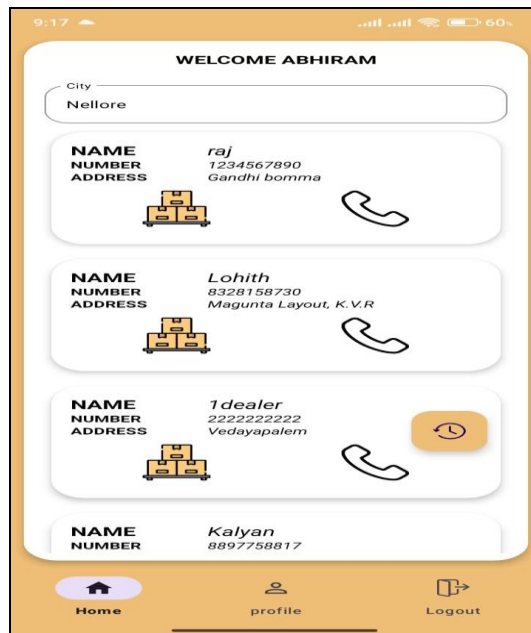


Fig: 4.6 User Home

4.6 USER HOME

The user home screen provides a comprehensive view of important functions and information essential for the user. It gives quick access to three main sections via a grid of icons or buttons named:

- Search Bar: This box is used for locating dealers based on location.
- Home: This button takes the user back to the current screen (landing page post-login).
- Profile: Enables the user to view and modify their profile details like name, address, and contact information.
- Logout: This button ends the user's session and logs them out of the application.

History: This button allows the user to check their previous orders in the app along with their status.

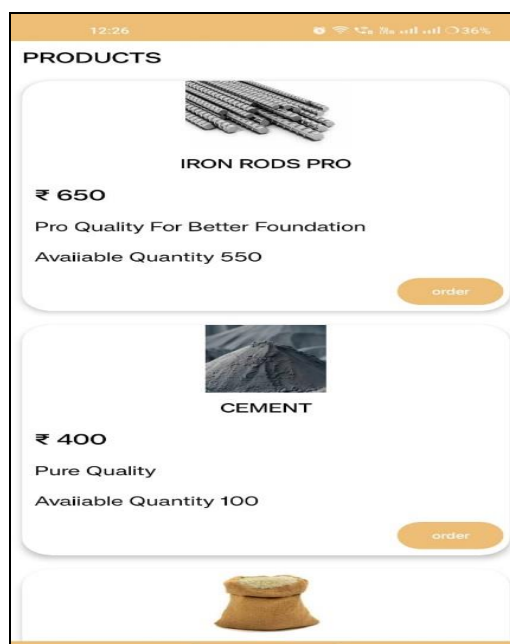


Fig: 4.7 Product List (User's View)

4.7 PRODUCTS LIST (USER'S VIEW)

The screen exhibits a roster of products ready for users to purchase. Here's a breakdown of the essential.

Items: The screen features an array of products with the following details:

Product Visual: Each product is highlighted with a picture, offering a visual guide for users.

Name of Product: The product's name is clearly presented below the image.

Cost: The price of each product be shown alongside the name.

Availability: Details on inventory will be provided, indicating the quantity available for each product.

Order Button: Adjacent to each product, there is a button enabling users to start the ordering process for that specific product.

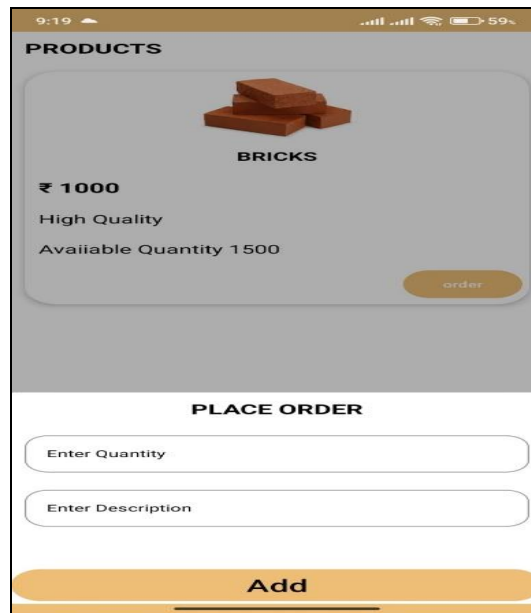


Fig: 4.8. User Order Product

4.8 USER ORDER PRODUCT

The screen you're looking at helps you start ordering a product from a chosen dealer. At the top, you'll see all the important info about the product.

Product Image: Check out the image of what you're ordering.

Product Name: The name of the product is right there.

Dealer Name: You'll see the dealer's name displayed.

Dealer Location: Where the dealer is located (city) will be shown.

In the middle section, you'll likely find details about your order:

Quantity: Use this field to enter how many of the product you want.

Request: If you have any specific instructions or requests, type them in here.

V. CONCLUSION

In conclusion, our Android application represents a significant step forward in revolutionizing the procurement process and enhancing the mobile commerce experience for suppliers, dealers, and users alike. By providing a centralized platform with tailored interfaces for each user type, we have streamlined communication, improved transparency, and fostered trust among stakeholders. Throughout the development and implementation of this application, our focus has been on enhancing efficiency, usability, and accessibility. The integration of features such as registration, login, product management, order tracking, and transaction history has empowered users to navigate the complexities of procurement with ease.

VI. FUTURE ENHANCEMENTS

Looking ahead, we recognize the potential for further enhancements to enrich the user experience and stay ahead of evolving market trends. Future iterations may include personalized recommendations, augmented reality integration, and machine learning algorithms to optimize inventory management and improve product recommendations. Ultimately, our goal is to continue pushing the boundaries of mobile commerce, leveraging technology to create seamless and intuitive experiences that exceed user expectations. By fostering transparent communication, enhancing efficiency, and prioritizing user satisfaction, we believe our application will play a pivotal role in shaping the future of e-commerce and transforming the way stakeholders interact in the digital marketplace.

References

- [1] Tan, S., Liu, Y., & Zhou, J. (2021). The development of an E-Commerce Android App utilizing MVC Architecture was discussed at the International Conference on Computer Science, Information Technology, and Applications (pp. 1-6).
- [2] Zhang, Y., & Chen, X. (2020). A study on an E-commerce Application System on the Android Platform was presented at the International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (pp. 226-231).
- [3] Zheng, Y., & Xu, H. (2019). The creation of an Android-based E-commerce Application catering to Small Businesses was explored at the International Conference on Management Science and Industrial Engineering (pp. 84-88).
- [4] Guo, Y., Zhang, W., & Wang, J. (2018). A detailed examination of an Android-based Mobile E-commerce Platform was conducted at the International Conference on Computer Science and Electronics Engineering (pp. 117-122).
- [5] Liu, S., Wang, Y., & Li, X. (2017). An in-depth look into an Android-based Mobile E-commerce Application was shared at the International Conference on Information Science and Control Engineering (pp. 100-105).
- [6] Hu, Q., Yu, J., & Xiong, J. (2016). The process of developing an Android-based E-commerce Application with User Behaviour Analysis was outlined at the International Conference on Advanced Cloud and Big Data (pp. 122-127).
- [7] Jiang, L., Zhang, H., & Wu, S. (2015). The design and implementation of an Android-based Mobile E-commerce System were reviewed at the International Conference on Electronics, Communications and Control (pp. 1221-1225).
- [8] Chen, T., Li, J., & Zhang, Q. (2014). The development of an Android-based E-commerce Application incorporating Location-based Services was highlighted at the International Conference on Wireless Communications & Signal Processing (pp. 10-15).