

Garment360: Analysis and Design of an Integrated Computerized Management System for Garment Industry Operations

MRS. K. BAKYA M.E.¹, S. PACHAIYAMMAL², K. BHUVANA³, A. AGALYA⁴, C. ABITHA⁵

¹Assistant Professor, ^{2,3,4,5}Student

Department of Computer Science and Engineering
Tagore Institute of Engineering and Technology – Deviyakurichi

Abstract—The garment industry continues to rely on fragmented, manual processes for managing inventory, workforce attendance, billing, and financial transactions, leading to data inconsistencies, processing delays, and poor decision-making. This paper presents the analysis and design of Garment360, a computerized integrated management system developed for garment business operations. The system requirement study was carried out across garment business workflows encompassing billing, labour attendance, stock management, transaction processing, and reporting. System analysis is performed using Data Flow Diagrams (DFDs) identifying key processes across four functional sub-systems: Billing, Labour Attendance, Stock Management, and Transaction Management. Design of system entities, their relationships, and attributes is carried out using Entity Relationship (ER) modelling. Structural design is accomplished using Unified Modelling Language (UML) techniques. The complete database consists of five normalized tables designed up to Third Normal Form (3NF). Built using Python with Tkinter and SQLite, the platform supports multi-branch operations across five locations with three-tier role-based access control (Admin, Manager, Staff) secured by SHA-256 hashing. All eight system test cases passed, confirming the reliability of the platform for small to medium-scale garment enterprises.

Keywords—garment management system; DFD; ERD; UML; database normalization; Python Tkinter; SQLite; inventory control; labour attendance; multi-branch operations; role-based access control; SME digitalization

I. INTRODUCTION

Enterprise Resource Planning (ERP) systems are computer-based information systems that enable integration of application programs for various business functions such as sales, purchasing, financial accounting, manufacturing, and human resources. Different business applications share a common database that serves as the integrating mechanism. However, generalized ERP systems impose their own organizational structure on a company, requiring significant changes to existing processes. The imposition of a different knowledge structure influences an organization's ability to internalize new knowledge into its business routines [1].

Garment manufacturing firms do not normally possess the resources of large firms in terms of technical expertise, financial strength, and professional management. It is therefore logical that they face greater challenges in implementing generalized ERP systems [1, 2]. Most garment manufacturing firms also rely on idiosyncratic processes that enable flexible customer demand fulfilment — a structure that generalized ERP systems tend to threaten, leading to resistance and non-acceptance.

The garment industry in India faces intense competition from neighbouring countries such as Sri Lanka, China, and Bangladesh. In order to sustainably hold a firm position in the global Ready-Made Garment (RMG) market, it is essential to implement computerized management systems. Major garment centres in India include Mumbai, Coimbatore, Tirupur, Ahmedabad, Ludhiana, Bangalore, and Indore [2].

This paper presents **Garment360**, a centralized Python-Tkinter desktop platform that consolidates billing, labour attendance, stock management, transaction processing, and reporting into a single cohesive system. The key contributions of this work are: (i) thorough requirement study of garment business operations; (ii) system analysis using Data Flow Diagrams (DFDs) identifying processes across four sub-systems; (iii) system design using ERD and UML; (iv) a complete normalized database schema; and (v) a

QR-code-based warehouse path planner, camera-integrated attendance, SHA-256 RBAC, and three-tier analytics dashboards.

II. NEED FOR A COMPUTERIZED SYSTEM FOR GARMENT INDUSTRY

Computerization has become a prevalent corporate response to free-trade related import competition in the garment sector. The garment manufacturing industry always aims for new product development and efficiency improvement in production. In past years, advanced computer technologies have facilitated new manufacturing operations and built management tools. Today's manufacturers focus increasingly on different types of networking tools to improve coordination [3].

In the globalization era, rapid data communication and timely access to accurate information is critical for correct decision-making. At present, many organizations in India use paper-based methods, which are time-consuming and error-prone. The use of internet-based and computer-assisted technologies to communicate information is one of the best approaches to support the informational needs of various departments [3].

It is extremely difficult to organize and maintain current and historical records manually. Most garment organizations store data manually, placing precious records at risk without security. Searching for a specific record is cumbersome and unreliable, often resulting in failure. Many existing ERP systems are expensive and difficult to maintain, making them inaccessible to most apparel industries [2]. These difficulties create the opportunity to build an effective and efficient system that consolidates all information access and updates across individual departments.

A. Present Status of Garment Industry Systems

Over the last 15–20 years the apparel sector has been in continuous restructuring. Technological and socio-economic changes, production cost pressures, liberalization, and the emergence of international competitors from Asia are influencing apparel manufacturing in India [3]. Efficient information flows within and

between departments are an important key factor for garment industries to improve coordination.

Most garment enterprises currently maintain isolated records per department: inventory, payroll, billing, and procurement are managed independently without cross-referencing. This decentralized approach produces: (i) data fragmentation and inconsistencies; (ii) high probability of human calculation errors; (iii) absence of real-time data visibility; (iv) no access control over sensitive financial and personnel records; and (v) time-consuming manual report generation requiring 30–45 minutes of effort per branch per day. In order to sustain competitive standing it is essential to implement a computerized information system that addresses all of these deficiencies.

III. SCOPE OF WORK

Development of an integrated management system for a garment company involves extensive work. Complete software design and development is carried out in four stages: (i) System Requirement Study; (ii) System Analysis; (iii) System Design; and (iv) System Development and Implementation. In the present work, all four stages are completed for the Garment360 platform.

The scope of work is structured as follows:

- Carrying out a literature survey on ERP systems and garment manufacturing management challenges.
- Carrying out system requirement analysis by understanding garment business operations, data flows, and information requirements across billing, attendance, stock, and transactions.
- Analysing data and information flow requirements using Data Flow Diagrams (DFDs) for all four functional sub-systems.
- Carrying out system design using Entity Relationship (ER) modelling and Unified Modelling Language (UML) techniques, and designing the complete normalized database.
- Implementing the designed system as a functional Python-Tkinter desktop application with SQLite persistence, multi-branch support, and role-based access control.
- Conducting system testing across unit, integration, security, and acceptance levels.

The platform is evaluated across five representative branches: Deviyakurichi (HQ), Salem, Erode, Coimbatore, and Tirupur, representing garment manufacturing units, textile warehouses, retail stores, and multi-branch business contexts.

TABLE I. Feature Comparison of Management Systems

Feature	G360	ERP	Standalone	Cloud
Cost	Low	Very High	Low	Medium
Full Integration	✓	✓	✗	✓
Offline Use	✓	Limited	✓	✗
Multi-Branch	✓	✓	✗	✓
RBAC Security	✓	✓	✗	✓
Open Source	✓	✗	Partial	✗

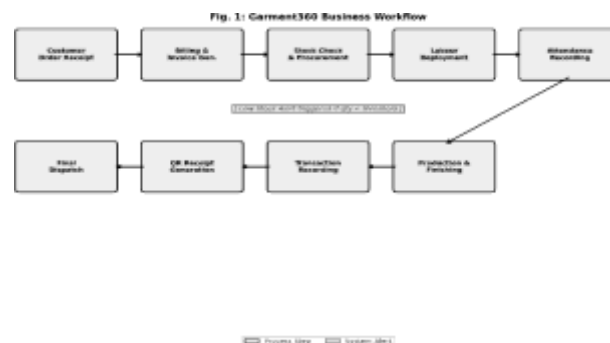
WH Path Planner	✓	✗	✗	✗
Payslip Generator	✓	✓	✗	✗

✓=Supported ✗=Not Supported WH=Warehouse G360=Garment360

IV. SYSTEM REQUIREMENT STUDY

A thorough requirement study of Garment360 was carried out by understanding the core business operations of garment enterprises. The method employed in garment business operations follows a sequential workflow: customer order receipt, billing and invoice generation, stock procurement and warehouse storage, labour deployment across shifts, production and finishing, transaction recording, and final dispatch.

The garment business process starts with a customer order. The billing module processes this into a GST-inclusive invoice. Stock levels are checked in real time, and if below threshold, a low-stock alert is triggered. Labour workers are assigned to departments (Cutting, Sewing, Finishing, Packing) across morning and evening shifts. Attendance is recorded at shift entry with camera-assisted identification. Each product movement — whether incoming from a supplier or outgoing to a customer — is recorded as a transaction. The complete flow from order receipt to dispatch is managed within Garment360.



A. Billing Sub-System Requirements

The billing process requires: product ID and name, size, quantity, unit price, and customer information as inputs. The system must auto-compute GST amounts and grand totals, assign unique bill numbers, and persist to the database. Users must be able to navigate previous and next bills, edit line items, delete records, and print QR-encoded receipts. Multi-item bills must be supported across multiple branches.

B. Labour Attendance Sub-System Requirements

The attendance process requires: Labour ID, name, department, shift (Morning/Evening), in-time, out-time, and attendance status (Present/Absent/Half-Day/On Leave) per worker. A camera feed must enable real-time face-based identification at entry. Worker Daily Work Entry must capture units stitched and defective units per shift for productivity and quality tracking. Payslip computation must draw from attendance data automatically.

C. Stock Management Sub-System Requirements

The stock module requires: product ID, name, colour, size, current quantity, branch/warehouse location, and last-updated timestamp. The system must provide real-time inventory visibility across all branches and trigger configurable low-stock alerts when quantity

falls below threshold. A QR-code-based warehouse path planner must identify the shelf location and render a 2D navigation path on a visual warehouse grid.

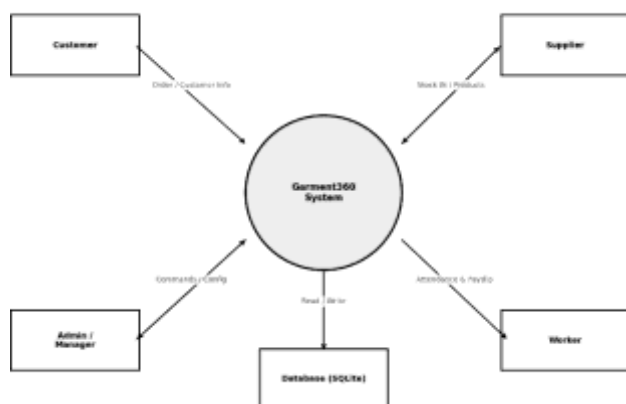
D. Transaction Management Sub-System Requirements

The transaction module requires: transaction ID, date-time, branch, warehouse, movement type (IN/OUT), product ID, quantity, unit price, total, party name, and party ID. Stock levels must be automatically recalculated after each transaction. The module must support branch-level filtering and maintain a complete immutable transaction ledger for audit purposes.

V. SYSTEM ANALYSIS

System analysis is performed using Data Flow Diagrams (DFDs) for proper understanding and ease of analysing the system. A DFD, also known as a “bubble chart,” clarifies system requirements and identifies major transformations that will become modules in the system design [4]. The Garment360 system comprises four functional sub-systems: (i) Billing Management, (ii) Labour Attendance Management, (iii) Stock Management, and (iv) Transaction Management. Each sub-system is described below.

Fig. 2: Level-0 Context DFD - Garment360



A. Billing Management Sub-System

There are six processes in this sub-system:

Customer Input: Customer and product information is received and stored in the customer and product data stores.

Product Validation: Product ID and stock availability are verified against the stock data store before billing proceeds.

GST Computation: Unit price and quantity inputs are processed to calculate GST-inclusive subtotals and grand totals automatically.

Invoice Generation: A unique bill number is assigned and the complete invoice record is created and stored in the transactions data store.

Receipt Printing: The invoice is rendered for print with an embedded QR code encoding the bill number and customer ID.

Bill Navigation & Edit: Previous/next bill navigation, line-item editing, and record deletion are supported from the billing history data store.

B. Labour Attendance Management Sub-System

There are seven processes in this sub-system:

Worker Identification: The camera feed panel captures a live feed at shift entry; worker presence is confirmed against the labour data store by Labour ID.

Attendance Marking: Shift type (Morning/Evening), in-time, out-time, and attendance status (Present/Absent/Half-Day/On Leave) are recorded and stored.

Daily Work Entry: Units stitched and defective units per shift are entered by the worker and stored in the work entry data store.

Performance Analysis: Productivity metrics are aggregated across the 14-day rolling window to produce the productivity chart in the Worker Portal.

Attendance Report Generation: Daily, weekly, and monthly attendance summaries are aggregated by branch and department and exported to Excel.

Payslip Computation: Base salary, attendance bonus, performance bonus, overtime pay, and PF deduction are computed automatically from attendance and work entry records.

Department Summary: Department-wise attendance breakdown (Cutting, Stitching, Finishing, Packing) is computed and displayed in the Manager Dashboard.

C. Stock Management Sub-System

There are five processes in this sub-system:

QR Code Scan: The product QR code is scanned within the warehouse Stock Point interface; the product ID is matched against the stock data store.

Warehouse Path Planning: The shelf location is identified and a highlighted 2D navigation path is rendered on the warehouse grid in under 2 seconds.

Stock Level Monitoring: Current quantity is checked against the configurable low-stock threshold (default: 10 units) after every transaction.

Alert Generation: A low-stock alert notification is triggered and displayed in the Manager Dashboard when quantity falls below threshold.

Stock Report Generation: Stock level audits showing current quantities per product per branch are aggregated and exported to Excel.

D. Transaction Management Sub-System

There are four processes in this sub-system:

Transaction Input: Movement type (IN/OUT), product ID, quantity, unit price, and party details are entered and validated.

Stock Update: The stock data store is updated immediately — OUT transactions decrement quantity; IN transactions increment quantity.

Transaction Ledger Recording: A unique transaction ID and complete movement record are stored in the transactions data store with date-time stamp.

Transaction Report Generation: Complete audit-trail transaction logs are exported to Excel with branch-level and date-range filtering.

TABLE II. DFD Process Summary

Sub-System	No. of Processes	Key Outputs
Billing Management	6	GST Invoice, Receipt, Bill Log
Labour Attendance	7	Attendance Record, Payslip, Perf. Report

Stock Management	5	Path Map, Low-Stock Alert, Stock Report
Transaction Mgmt	4	Transaction Ledger, Stock Update, Report
Authentication	3	Session Token, Role Dashboard, Audit Log

VI. METHODOLOGY

The analysis determines whether the problem warrants further development and refines the problem statement. System analysis is carried out using Data Flow Diagrams (DFDs) for proper understanding and ease of analysing the system [4]. The software application's success is one of the most enduring research topics in the field of information systems. The design of the database for the proposed management system is carried out with the help of ERD and UML diagrams, which clearly represent the data flow in the system [4, 8].

The use of ERD gives a structural idea about the system and the activities taking place within it. UML diagrams then provide a clearer, more detailed and well-organized design that generates the database tables. Our investigation is informed by prior research that provides a framework for analysis [5, 6].

The methodology followed for Garment360 is: (i) literature survey on ERP systems and garment management; (ii) system requirement study identifying data and information flows; (iii) DFD-based system analysis across four sub-systems; (iv) ERD design identifying entities and relationships; (v) UML-based structural design and database table generation; (vi) normalized database implementation in SQLite; and (vii) test case validation across unit, integration, security, and acceptance levels.

VII. DESIGN OF THE SYSTEM

The Garment360 database tracks entities across five core domains: users (authentication and access control), branches (multi-location management), labour (workforce and attendance), stock (inventory per branch), and transactions (financial and stock movement records). According to the system requirements, five entity types are identified, each corresponding to a distinct operational domain.

A. Entity Descriptions

User Entity: The User entity has a unique User_ID (primary key), username, SHA-256 hashed password, and role (Admin/Manager/Staff). It controls access to all other entities based on role permissions. Admin has full global access; Manager has branch-level access; Staff is restricted to data entry.

Branch Entity: The Branch entity has a unique Branch_ID (primary key), branch name, and location. It serves as the organizational unit linking all operational data. Five branches are registered: Deviyakurichi, Salem, Erode, Coimbatore, and Tirupur.

Labour Entity: The Labour entity has a unique Labour_ID (primary key), worker name, Branch_ID (foreign key), department, attendance status, and date. It tracks all workforce presence and productivity data. At evaluation, 158 workers are registered across all branches.

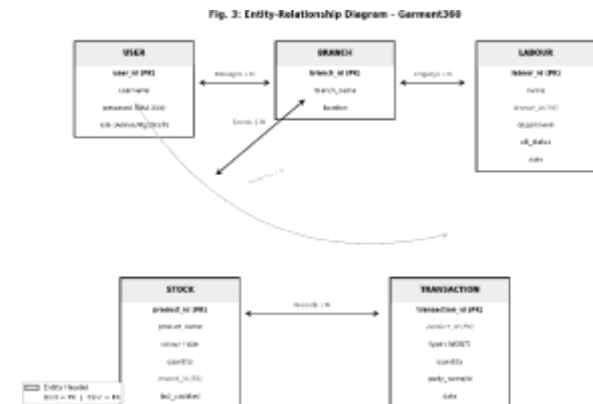
Stock Entity: The Stock entity has a unique Product_ID (primary key), product name, colour, size, quantity, Branch_ID (foreign key), and last-updated timestamp. It maintains real-time inventory across all branches. Total stock at evaluation: 18,430 units across all locations.

Transaction Entity: The Transaction entity has a unique Transaction_ID (primary key), Product_ID (foreign key referencing Stock), movement type (IN/OUT), quantity, party name, party ID, and date-time. It implements the complete auditable stock movement ledger.

B. ERD Relationships

The ERD for the Garment360 system identifies the following relationships between entities:

- (i) **User – Branch (Manages):** One Admin manages many branches (1:N). Each Manager is assigned to exactly one branch (1:1).
- (ii) **Branch – Labour (Employs):** One branch employs many labour workers (1:N). Each worker belongs to exactly one branch.
- (iii) **Branch – Stock (Stores):** One branch stores many products (1:N). Each product record belongs to one branch location.
- (iv) **Stock – Transaction (Records):** One product can appear in many transactions (1:N). Each transaction references exactly one product via foreign key.
- (v) **User – Transaction (Performs):** One staff member can perform many transactions (1:N). Each transaction is associated with one authorized user.



C. UML Class Structure

The UML class diagram defines four core classes: **User** (userID, username, password, role; login(), logout(), validateCredentials()) associates with all operational classes via role permissions. **Labour** (labourID, name, branch, department, shift; markAttendance(), getDailyReport(), computePayslip()) links to User via staffID. **Product** (productID, name, colour, size, quantity, branch; updateStock(), getLowStock(), getWarehousePath()) is referenced by Transaction. **Transaction** (transactionID, productID[FK], type, quantity, partyName, date; save(), getHistory(), generateReport()) holds a FK relationship to Product.

D. System Architecture

Garment360 employs a three-layer modular architecture. The **Presentation Layer** (Tkinter) renders all graphical interfaces. The **Application Layer** contains all business logic: input validation, CRUD operations, alert generation, invoice computation, payslip calculation, and report aggregation. The **Data Layer** uses SQLite for 3NF relational persistence and openpyxl for Excel report output. All layers support multi-branch routing through parameterized SQL queries preventing SQL injection.

Fig. 4: Garment360 Three-Layer Architecture

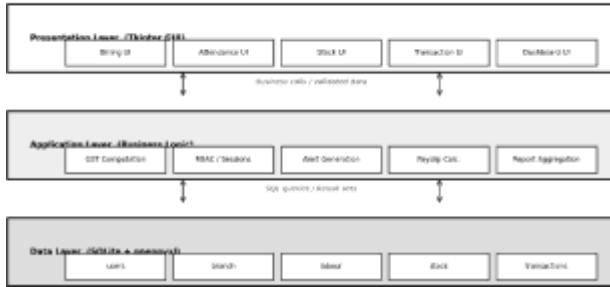


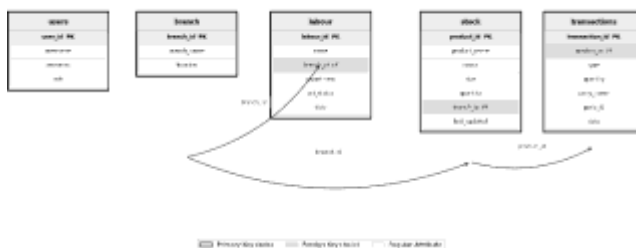
TABLE III. System Requirements

Component	Specification
Processor	Intel Core i3 or equivalent
RAM	4 GB minimum
Storage	500 GB HDD
OS	Windows 10/11, Linux, macOS
Language	Python 3.10+
GUI Framework	Tkinter (built-in)
Data Library	Pandas 2.x
Database	SQLite 3 (built-in)
Report Export	openpyxl (.xlsx)
Security	hashlib SHA-256

VIII. NORMALISATION OF DATABASE TABLES

Normalization is a technique for designing relational database tables to minimize duplication of information and to safeguard the database against logical or structural problems such as data anomalies [4]. The Garment360 database is normalized through the following normal forms:

Fig. 5: Normalized Database Schema (3NF/BCNF) - Garment360



a. First Normal Form (1NF)

Eliminate Repeating Groups — make a separate table for each set of related attributes and give each table a primary key. In Garment360: the labour table stores one attendance record per worker per date (atomic values, no repeating groups); the stock table stores one row per product per branch; the transactions table stores one row per movement event. All five tables carry explicit primary keys (user_id, branch_id, labour_id, product_id, transaction_id).

b. Second Normal Form (2NF)

Eliminate Redundant Data — if an attribute depends on only part of a multi-valued key, remove it to a separate table. In Garment360: branch information (name, location) is separated into its own branch table rather than duplicated in every labour and stock record. This ensures that updating a branch name requires changing exactly one row, not hundreds of labour and stock records.

c. Third Normal Form (3NF)

Eliminate Columns Not Dependent on Key — if attributes do not contribute to a description of the key, remove them to a separate table. In Garment360: party details (party_name, party_id) in the transactions table describe the transaction event (transaction_id), not the product. Role permissions are stored in the users table against user_id, not duplicated in labour or transaction records. No transitive dependencies exist in the final schema.

d. Boyce-Codd Normal Form (BCNF)

The remaining anomalies from functional dependencies are removed. In Garment360: each table has exactly one candidate key (the primary key). No non-trivial functional dependency $X \rightarrow Y$ exists where X is not a superkey. The foreign key constraints (branch_id in labour and stock, product_id in transactions) enforce referential integrity without introducing BCNF violations.

```
PRAGMA foreign_keys = ON;
```

```
-- 1NF: atomic values, primary keys on all tables
-- 2NF: no partial dependencies on composite keys
-- 3NF: no transitive dependencies
```

```
CREATE TABLE IF NOT EXISTS users (
    user_id TEXT PRIMARY KEY,
    username TEXT NOT NULL UNIQUE,
    password TEXT NOT NULL, -- SHA-256 hash
    role TEXT NOT NULL -- Admin | Manager | Staff
);
```

```
CREATE TABLE IF NOT EXISTS branch (
    branch_id TEXT PRIMARY KEY,
    branch_name TEXT NOT NULL,
    location TEXT NOT NULL
);
```

```
CREATE TABLE IF NOT EXISTS labour (
    labour_id TEXT PRIMARY KEY,
    name TEXT NOT NULL,
    branch_id TEXT NOT NULL,
    department TEXT NOT NULL,
    attendance_status TEXT NOT NULL,
    date DATE NOT NULL,
    FOREIGN KEY (branch_id) REFERENCES branch(branch_id)
);
```

```
CREATE TABLE IF NOT EXISTS stock (
    product_id TEXT PRIMARY KEY,
    product_name TEXT NOT NULL,
    colour TEXT,
    size TEXT,
    quantity INTEGER NOT NULL DEFAULT 0,
    branch_id TEXT NOT NULL,
    last_updated DATETIME NOT NULL,
    FOREIGN KEY (branch_id) REFERENCES branch(branch_id)
);
```

```
CREATE TABLE IF NOT EXISTS transactions (
    transaction_id TEXT PRIMARY KEY,
    product_id TEXT NOT NULL,
    type TEXT NOT NULL, -- IN | OUT
    quantity INTEGER NOT NULL,
    party_name TEXT,
    party_id TEXT,
    date DATETIME NOT NULL,
    FOREIGN KEY (product_id) REFERENCES stock(product_id)
);
```

IX. MODULE DESCRIPTION

A. Authentication & RBAC

All access is gated through a secure login module. Passwords are stored as SHA-256 hashes via Python's hashlib library. Three roles are enforced: **Admin** (full global access), **Manager** (branch-level operations and analytics), **Staff** (data entry only). Session tokens are issued on successful login and invalidated at logout. Invalid login attempts display a generic error without field-level disclosure.

B. Billing System

Generates GST-inclusive invoices from product ID, name, size, quantity, unit price, and customer data. Auto-computes tax and grand totals, assigns a unique bill number, and persists to SQLite. Users navigate bills, edit line items, delete records, and print QR-encoded receipts. Evaluation: Salem Branch Store 3 — Grand Total ₹7,261.31 across five product lines.

C. Labour Attendance System

Records shift data per worker: Labour ID, name, department, shift (Morning/Evening), in-time, out-time, and status. Camera feed panel enables real-time identification. Worker Daily Work Entry captures units stitched and defective units. Payslip for G360-0013 in April 2026: Base ₹12,000 + Bonuses ₹2,000 + Overtime ₹450 – PF ₹1,440 = Net ₹13,010.

D. Stock Management & Warehouse Path Planner

Real-time inventory visibility with QR-based Warehouse Path Planner. Products located in under 2 seconds on a 2D grid. Low-stock evaluation alerts: Denim Jeans 32" — 7 units (Critical), Women's Kurti L — 42 units, Cotton Saree Red — 31 units, Kids Pant 26" — 12 units, Men's Vest M — 28 units.

E. Transaction Management

Records all IN/OUT stock movements. Each transaction captures unique Transaction ID, date-time, branch, product, quantity, unit price, total, and party details. Stock auto-recalculates per transaction. Evaluation: Denim Jacket OUT 20 units at ₹200 = ₹4,000 total; Salem Warehouse updated 157→137 units instantly.

F. Reporting System

Aggregates all module data into Excel (.xlsx) reports via openpyxl: attendance summaries, stock audits, billing histories, transaction logs. Filterable by date range, branch, and department. Auto-generated reports replace 30–45 minutes of daily manual effort with under 2 minutes.

G. Three-Tier Analytics Dashboard

Admin: Enterprise KPIs — Revenue ₹24.6L (+12.4% MoM), Stock 18,430 units, Active Workers 142/158, Pending Transactions 37. **Manager** (Deviyakurichi): Bills 84, Stock OUT 1,320, Attendance 91%, Pending Approvals 6. **Worker Portal** (G360-0013): 22 days present, 148 units stitched, 96% attendance, Net Pay ₹13,010.

X. SYSTEM TESTING

Testing was conducted across four levels: Unit Testing (each module function in isolation), Integration Testing (data flow between billing–stock, transaction–stock, attendance–payslip), Security Testing (SHA-256 hashing, role boundary enforcement, session invalidation), and Acceptance Testing (domain users in real-world scenarios).

TABLE IV. System Test Cases

TC ID	Module	Expected Output	Result
-------	--------	-----------------	--------

TC01	Authentication	Dashboard with KPIs loaded	Pass
TC02	Authentication	Error shown, access denied	Pass
TC03	Billing	GST invoice generated	Pass
TC04	Attendance	Record saved in DB	Pass
TC05	Stock Mgmt	Stock level updated instantly	Pass
TC06	Stock Alert	Alert notification triggered	Pass
TC07	Reporting	Excel (.xlsx) report exported	Pass
TC08	Transactions	Transaction recorded, stock updated	Pass

All eight test cases passed with zero defects. The system correctly rejected invalid credentials, enforced role-based access at every module boundary, auto-computed GST totals, persisted records at sub-second latency, triggered configurable alerts, and exported well-formed Excel workbooks maintaining a complete immutable transaction ledger.

XI. RESULTS AND DISCUSSION

Garment360 was evaluated across five branches. The Admin Dashboard reported Total Revenue ₹24.6L (+12.4% MoM), Total Stock 18,430 units, Active Workers 142/158, and 37 Pending Transactions (8 overdue >3 days). Monthly revenue trend analysis (May 2025–Apr 2026) showed consistent growth against target.

The Manager Dashboard (Deviyakurichi) captured: Bills Generated 84, Stock OUT 1,320 (high demand), Stock IN 540, Worker Attendance 91% (32/35), and 6 Pending Approvals. Department-wise: Cutting 8/8, Stitching 12/14, Finishing 7/8, Packing 5/5. Five active low-stock alerts enabled proactive procurement.

Key operational improvements validated: (i) billing errors reduced to zero through automated GST computation; (ii) attendance recording reduced from ~20 minutes to under 2 minutes per shift; (iii) stock reconciliation made instantaneous via automated transaction-linked updates; (iv) daily reporting reduced from 30–45 minutes to under 2 minutes per branch; (v) payslip generation reduced from manual calculation to automated sub-second computation.

XII. CONCLUSION & FUTURE ENHANCEMENT

Our findings confirm that lack of adequate resources constrains garment enterprises from effectively implementing generalized ERP systems, and that these constraints have different impacts at different operational stages.

(i) Thorough requirement study of Garment360 resulted in identifying important activities in garment business management and related data and information flows across four sub-systems.

(ii) After system analysis, the Garment360 platform comprises four functional sub-systems: Billing Management, Labour Attendance Management, Stock Management, and Transaction Management.

(iii) Analysis of data flow, information flow, and processing activities using DFDs identified six processes in Billing, seven in

Labour Attendance, five in Stock Management, and four in Transaction Management.

(iv) Preliminary system design using ERD identified five entity types (User, Branch, Labour, Stock, Transaction) with five relationships between them.

(v) Structural design using UML produced four core classes and a complete database schema normalized to 3NF/BCNF.

(vi) The complete database consists of five tables with attributes, data types, and relationship keys fully specified and implemented. All eight test cases passed, confirming platform reliability.

Future enhancements: (i) web-based architecture using Django/Flask; (ii) mobile companion application for Android/iOS; (iii) cloud storage integration (AWS S3/Firebase); (iv) ML-based demand forecasting and inventory optimization; (v) barcode scanning hardware, automated procurement notifications, and advanced drill-down analytics.

REFERENCES

- [1] T. H. Davenport, "The future of enterprise system-enabled organizations," *Information Systems Frontiers*, vol. 2, no. 2, pp. 163–180, 1998.
- [2] Z. Lee and J. Lee, "An ERP implementation case study from a knowledge transfer perspective," *J. Information Technology*, vol. 15, pp. 281–288, 2000.
- [3] M. Summer, "Risk factors in enterprise-wide/ERP projects," *J. Information Technology*, vol. 15, no. 4, pp. 317–327.
- [4] R. Elmasri and S. Navathe, *Fundamentals of Database Systems*, 3rd ed. Addison-Wesley, 2000.
- [5] E. Handoyo, "SRS Document Proposal Analysis on the Design of MIS According to IEEE STD 830-1998," *Procedia Social and Behavioural Sciences*, vol. 67, pp. 123–134, 2012.
- [6] R. K. Chakraborty and S. K. Paul, "Study and Implementation of Lean Manufacturing in a Garment Manufacturing Company," *J. Optimization in Industrial Engineering*, vol. 7, pp. 11–22, 2011.
- [7] I. Sommerville, *Software Engineering*, 10th ed. Pearson, 2016.
- [8] J. Rumbaugh, I. Jacobson, and G. Booch, *UML Reference Manual*, 2nd ed. Addison-Wesley, 2004.
- [9] W. McKinney, *Python for Data Analysis*, 3rd ed. O'Reilly, 2022.
- [10] M. T. Tsai and C. H. Lin, "Integrated Inventory Management for Apparel Industries," *J. Textile Engineering*, vol. 8, no. 2, 2022.
- [11] A. Tiwari and K. Sharma, "Digital Transformation in Small Garment Industries Using Open Source Tools," *Proc. IEEE ICET*, 2023.
- [12] S. Bhattacharya, "Role-Based Access Control in Lightweight Desktop ERP Applications," *Int. J. Inf. Syst. Manag.*, vol. 5, no. 1, 2021.

978-1-XXXX-XXXX-X/26/\$31.00 © 2026 IEEE