

Sales and Customer Management System

(A Web Based Project Management Application)

BY

Pranish Sris

TU Roll Number: 14196/21

TU Registration Number: 7-2-559-45-2021

Bishal Somera

TU Roll Number: 14208/21

TU Registration Number: 7-2-559-33-2021

Oxford College, Butwal

A Project Report Submitted to

Faculty of Management, Tribhuvan University

in partial fulfillment the requirement for the degree of

Bachelor of Information Management (BIM)

Butwal

May,2024

STUDENT DECLARATION

This is to certify that we have completed the Project entitled "**GanaSalesLite: A Sales and Customer Management System for Ganapati Enterprise**" under the guidance of **Mr. Suraj Khatri** in partial fulfillment of the requirements for the degree of Bachelor of Information Management at Faculty of Management, Tribhuvan University. This is our original work and we have not submitted it earlier elsewhere.

CERTIFICATE FROM THE SUPERVISOR

This is to certify that the project entitled "**GanaSalesLite: A Sales and Customer Management System for Ganapati Enterprise**" is an academic work done by **Bishal Somare** and **Pranish Sris** submitted in the partial fulfillment of the requirements for the degree of Bachelor of Information Management at Faculty of Management, Tribhuvan University under my guidance and supervision. To the best of my knowledge, the information presented by them in the project report has not been submitted earlier.

Signature of the Supervisor

Name: Suraj Khatri

Designation: Lecturer

Date:

APPROVAL SHEET

This is to certify that the project titled "**GanaSalesLite: A Sales and Customer Management System for Ganapati Enterprise**" submitted by **Bishal Somare** and **Pranish Sris** has been examined and approved. In our opinion, it meets the required scope and quality standards for a project submitted in partial fulfillment of the requirements for the degree of Bachelor of Information Management (BIM).

Approval Panel:

S.N	Name	Designation	Signature
1.	Suraj Khatri	Project Supervisor	
2.	Ashok Gurung	Program Coordinator	
3.	Om Khatri	Internal Examiner	
4.	External	External Examiner	

Date of Defense:

Department:

Faculty: Management

ACKNOWLEDGEMENTS

The successful completion of this project, “**GanaSalesLite**,” is the culmination of dedicated effort, guidance, and support from various individuals, to whom we extend our deepest gratitude.

First and foremost, we are immensely grateful to our project supervisor, **Mr. Suraj Khatri**, for his invaluable guidance, constructive feedback, and unwavering support throughout the project lifecycle. His insights and expertise were instrumental in shaping this project and navigating challenges.

We would like to express our sincere thanks to **Mr. Ashok Gurung**, Program Coordinator (BIM), and the **Faculty of Management, Tribhuvan University**, for providing us with the opportunity to undertake this project as part of our curriculum and for their continuous encouragement.

Our heartfelt appreciation goes to **Oxford College, Butwal**, for providing the necessary resources and a conducive learning environment.

We are also thankful to **Ganapati Enterprise** for allowing us to understand their operational context, which formed the basis of this project.

Finally, we wish to thank our families and friends for their constant encouragement, patience, and support during this endeavor.

Name: Pranish Sris

TU Roll Number: 14208/21 TU

Registration Number: 7-2-559-45-2021

Name: Bishal Somera

TU Roll Number: 14196/21 TU

Registration Number: 7-2-559-33-2021

ABSTRACT

Small and medium-sized enterprises (SMEs) like Ganapati Enterprise often face challenges in efficiently managing their sales and customer data due to reliance on manual processes or overly complex software. This project, “GanaSalesLite” presents a streamlined Sales and Customer Management System designed specifically for the single-admin/owner of Ganapati Enterprise. The system aims to automate and simplify core business operations, including product management, customer record keeping, sales transaction entry, billing, and basic inventory overview and sales reporting.

Developed using the RAD Model, GanaSalesLite employs a technology stack comprising Python with the Django framework for the backend, postgresql for database management, and HTML, CSS, and JavaScript (with Bootstrap) for a responsive and user-friendly frontend. The system focuses on providing essential functionalities without the overhead of features typically found in larger ERP systems, making it suitable for a small-scale enterprise with a single primary user.

Key features include intuitive interfaces for managing products and customer details, an efficient sales entry process with automated bill generation, and a dashboard providing an overview of sales and inventory status. The project is expected to enhance operational efficiency, improve data accuracy, provide better insights through simple reports, and ultimately support Ganapati Enterprise in its day-to-day business activities.

Table of Contents

STUDENT DECLARATION	ii
CERTIFICATE FROM THE SUPERVISOR	iii
APPROVAL SHEET.....	iv
ACKNOWLEDGEMENTS.....	v
ABSTRACT.....	vi
CHAPTER ONE: INTRODUCTION.....	1
1.1 PROJECT BACKGROUND	1
1.2 PROBLEM STATEMENT	1
1.3 OBJECTIVES	2
1.4 Development Methodology.....	3
1.5 Scope and Limitations.....	5
1.5.1 Scope.....	5
1.5.2 Limitations	6
1.6 Report Organization.....	7
CHAPTER TWO: SYSTEM DEVELOPMENT PROCESS	9
2.1 Analysis.....	9
2.1.2 Feasibility Study	10
2.1.3 Modelling.....	10
2.2 Design	13
2.2.1 User Interface (UI) Design	13
2.2.2 Database Design	14
2.3 Implementation	14
2.3.1 Tools and Technologies Used	15
2.3.2 Module Description	15
2.3.3 Testing.....	17
CHAPTER THREE: CONCLUSION AND RECOMMENDATION	20

3.1 Summary	20
3.2 Conclusion	21
3.3 Recommendation	22
REFERENCES	24

List of Figures

Figure 1: Rapid Application Development	5
Figure 2:Use Case Diagram	11
Figure 3:Sales Processing	12
Figure 4:Activity for Product Management.....	13
Figure 5:ER Diagram	14

List of Tables

Table 1: Tools and Technologies Used in GanaSalesLite Development.....	15
Table 2: Sample Test Case – Login Module	18
Table 3: Sample Test Case – Product Management.....	18
Table 4: Sample Test Case – Sales and Billing Module	19
Table 5: Sample Test Case – Reporting Module.....	19

CHAPTER ONE: INTRODUCTION

1.1 PROJECT BACKGROUND

In today's competitive business environment, efficient management of sales, customer relationships, and inventory is crucial for the success and sustainability of any enterprise, regardless of its size. Small and medium-sized enterprises (SMEs) often struggle with these aspects due to limited resources, reliance on manual record-keeping, or the use of disparate, non-integrated tools. Ganapati Enterprise, a Distributor located in Devdaha, Ghodaha, currently manages its operations using traditional methods, which can be time-consuming, prone to errors, and may not provide timely insights for decision-making.

The “GanaSalesLite” project is conceived to address these challenges by developing a dedicated Sales and Customer Management System tailored to the specific needs of Ganapati Enterprise. The system is designed for a single user – the owner/admin – to provide a simple, intuitive, and effective tool for managing daily business operations. By automating key processes such as product tracking, customer data management, sales recording, and basic reporting, GanaSalesLite aims to empower Ganapati Enterprise with better control and visibility over its business.

1.2 PROBLEM STATEMENT

Ganapati Enterprise currently faces several operational inefficiencies due to its manual or semi-automated system for managing sales and customer interactions. The primary problems identified are:

- **Inefficient Data Management:** Tracking product details, customer information, and sales transactions manually (e.g., in ledgers or basic spreadsheets) is laborious and susceptible to human error. This can lead to inaccurate records and difficulty in retrieving information quickly.
- **Time-Consuming Sales Process:** Generating bills, updating inventory after each sale, and tracking customer purchase history by hand consumes significant time that could be utilized for other core business activities.
- **Limited Customer Relationship Management:** Without a structured system, it's challenging to maintain a comprehensive customer database, track purchase patterns, or offer personalized service.
- **Lack of Real-time Inventory Overview:** Manual tracking makes it difficult to ascertain current stock levels accurately, potentially leading to stock-outs or overstocking, both of which impact profitability.

- **Difficulty in Generating Reports:** Extracting meaningful insights from manual records to understand sales trends, popular products, or overall business performance is a complex and often neglected task.
- **No Centralized System: Information** might be scattered across different physical books or files, making it hard to get a holistic view of the business operations.

The proposed “GanaSalesLite” system aims to provide a centralized, user-friendly solution to these problems, specifically for the single owner/admin of Ganapati Enterprise.

1.3 OBJECTIVES

The main goal of our project, “GanaSalesLite,” is to build a simple and effective computer system to help the owner of Ganapati Enterprise manage their daily sales, customer information, and products more easily.

What We Aim to Achieve (General Objectives):

- Make everyday tasks like recording sales, keeping track of customers, and managing product stock faster and less manual.
- Help Ganapati Enterprise run more smoothly by cutting down on paperwork and mistakes.
- Create one central place where the owner can find all important business information about products, customers, and sales.
- Give the owner a better overview of their business so they can make informed decisions.

What the System Will Specifically Do (Specific Objectives):

1. **Secure Login:** Only the owner (admin) will be able to log in to the system securely.
2. **Manage Products:** The owner will be able to:
 - Add new products with details like name, price, and category.
 - See a list of all products.
 - Change product details.
 - Organize products into different categories.
3. **Manage Customers:** The owner will be able to:
 - Add new customer information (like name and contact details).

- See a list of all customers.
- Update customer details.
- See what a customer has bought in the past.

4. Handle Sales and Billing: The owner will be able to:

- Easily enter new sales.
- Add several products to one sale.
- Have the system automatically calculate the total cost.
- Print or save a bill/receipt for the customer.

5. Basic Stock Tracking:

- The system will automatically reduce the stock count of a product when it's sold.
- The owner can see how much of each product is currently in stock.
- The system will show an alert when a product's stock is low and Expired Product.

6. Simple Reports:

- The main screen (dashboard) will show quick summaries, like total sales for the day or week.
- The owner can look at basic details, like a list of all sales or which products are selling most.

7. **Manage Vendors:** The owner will be able to keep basic information about the suppliers they buy products from.
8. **Easy to Use:** The system will have a clear and simple design so that it's easy for the owner to use, even if they aren't very familiar with computers.
9. **Reliable Information:** The system will help ensure that all the business information is accurate and dependable for daily use.

1.4 Development Methodology

For the development of the “GanaSalesLite” system, the **Rapid Application Development (RAD)** model has been selected. RAD is an adaptive software development approach that prioritizes rapid prototyping and iterative development cycles, with significant user involvement throughout the process. This methodology is particularly effective for projects where requirements can evolve, and getting a working

version of the system (or key parts of it) into the hands of the user quickly is beneficial for feedback and refinement.

The core principles of RAD that guided this project include:

1. Iterative Prototyping and User Feedback: Rather than spending extensive time on upfront detailed design, RAD emphasizes quickly building prototypes of key system components and user interfaces. These prototypes are then presented to the end-user (the owner/admin of Ganapati Enterprise in this case) for hands-on interaction and feedback. This feedback is then immediately used to refine the prototype in subsequent iterations, ensuring the system evolves to closely match the user's actual needs and workflow.

2. Active User Involvement: The success of RAD relies heavily on continuous communication and collaboration with the user. For "GanaSalesLite", this means regular interaction with the intended admin to review prototypes, discuss functionalities, and make adjustments. This direct involvement helps ensure the final system is practical, usable, and directly addresses the identified business problems.

3. Focus on Essential Functionality (Time-boxing): RAD often employs time-boxing for development cycles, focusing on delivering core, high-priority features within short, defined periods. This helps maintain momentum and ensures that a usable version of the system is produced relatively quickly. While we may not strictly time-box each micro-feature, the spirit is to prioritize essential modules first.

4. Use of Tools for Speed: Where appropriate, tools that accelerate development (like Django's ORM, admin interface, and templating engine) are leveraged to speed up the construction phase. The emphasis is on building functional software rather than getting bogged down in overly complex, low-level details initially.

The RAD lifecycle for “GanaSalesLite” can be broadly viewed through these stages:

- **Requirements Planning:** A condensed initial phase to gather high-level requirements and understand the core business needs of Ganapati Enterprise for sales and customer management.
- **User Design (Iterative Prototyping):** Developing and refining interactive prototypes of GanaSalesLite's modules (e.g., product entry, sales screen, customer view). Each iteration incorporates user feedback.
- **Construction:** Building the functional components of GanaSalesLite based on the finalized prototypes, with ongoing testing of individual parts and their integrations.
- **Cutover (Deployment):** Finalizing the system, conducting overall system testing, preparing user guidance, and delivering the working application to Ganapati Enterprise.

RAD Model Diagram

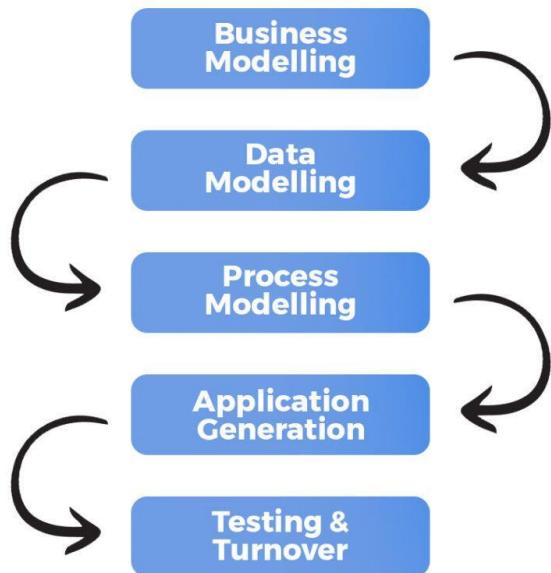


Figure 1: Rapid Application Development

The decision to use RAD for “GanaSalesLite” was driven by the desire to create a system that is highly tailored to the specific, single-user needs of Ganapati Enterprise. The iterative feedback loop allows for flexibility and ensures that the end product is not just technically sound but also genuinely practical and valuable for the owner/admin in their daily operations. This approach helps in quickly identifying and addressing any usability issues or overlooked requirements, leading to a more effective and user-accepted system.

1.5 Scope and Limitations

This part of the report talks about what our “GanaSalesLite” system is designed to do – that’s the **scope**. It also explains what the system is not designed to do, or any restrictions it has – those are the **limitations**.

1.5.1 Scope

“GanaSalesLite” is essentially a digital helper built specifically for the owner of **Ganapati Enterprise** to make managing their daily sales and customer information much simpler and more organized. The owner will be able to use the system for a variety of important tasks.

For starters, only the **owner** will have secure access to the system using a **login** and **password**. Once logged in, they can manage all their products. This includes adding

new items with details such as product name, price, category, and available stock. They can view a full list of products, update details if necessary, and remove items no longer in use.

Customer management will also be simplified. The owner can store essential customer details like names and contact information. A full list of customers will be maintained, and past purchase history can be viewed and updated when needed.

For **sales operations**, the system makes billing faster and error-free. The owner can record new sales, add multiple items to a single bill, and let the system automatically calculate the total. It can generate a receipt or bill that can be printed or saved.

In terms of **stock management**, GanaSalesLite reduces stock count automatically when a sale is made. The system shows real-time stock levels and can alert the owner when any product falls below a predefined reorder level.

A **dashboard** will provide a quick overview with key sales summaries like daily and weekly sales. Basic reports can also be viewed, such as a list of all transactions or a summary of best-selling products.

The system also allows saving **supplier information** for reference. Since it's a **web-based application**, it can be accessed from any computer with a web browser.

1.5.2 Limitations

While "GanaSalesLite" is designed to be a very useful tool for Ganapati Enterprise, there are certain things it does not cover.

- It is a **single-user system**, meant for use by the owner only. It does not support multiple user logins or roles with different permissions.
- Being a **web-based application**, it requires an internet connection if hosted online. If run on a local computer, it won't require the internet, but there is no offline mode if the system or network fails. There is also no mobile app available.
- The **inventory management** is basic. It reduces stock after sales but does not support complex features like supplier ordering, invoice tracking, or stock auditing.
- The **reporting** feature provides only simple summaries. There is no functionality for generating highly customized or analytical business reports with advanced charts or forecasts.
- "GanaSalesLite" **does not handle payment processing**. It cannot connect with credit card machines or online payment gateways. It only records transactions.

- The system is **suitable for small businesses**. If Ganapati Enterprise expands into a large enterprise with complex requirements and a high volume of transactions, major updates or a more advanced system would be needed.
- While it supports storing customer details and viewing their purchase history, it does not offer advanced CRM (Customer Relationship Management) features such as marketing automation or detailed customer analytics.

1.6 Report Organization

This project report for “GanaSalesLite: A Sales and Customer Management System for Ganapati Enterprise” is structured into several key parts to provide a clear and comprehensive overview of the project from its conception to its conclusion.

The report begins with **preliminary sections** such as the Title Page, Student Declaration, Certificate from the Supervisor, Approval Sheet, Acknowledgements, Abstract, Table of Contents, List of Figures, List of Tables, and a List of Abbreviations used throughout the document.

Following these, the main body of the report is divided into **three primary chapters**:

- **Chapter I: Introduction**

This chapter sets the stage for the project. It starts with the **Background of the Project**, explaining the context and need for a system like “GanaSalesLite” for Ganapati Enterprise. It then clearly defines the **Problem Statement** that the project aims to address, followed by the main, general, and **Specific Objectives** of the system. **A Review of Related Work and Literature** is presented, discussing existing similar systems and relevant theoretical concepts. The **Development Methodology** chosen for the project (Rapid Application Development - RAD) is explained along with its justification. Finally, this chapter outlines the **Scope and Limitations** of “GanaSalesLite,” detailing what the system will and will not cover, and concludes with this section on **Report Organization**.

- **Chapter II: System Development Process**

This chapter delves into the practical journey of creating “GanaSalesLite.” It is structured according to the phases of the chosen development methodology.

The **Analysis** subsection details the process of requirement gathering (both functional and non-functional requirements) and the feasibility study (technical, operational, and economic). It also includes system modeling artifacts like Use Case Diagrams and Activity Diagrams to illustrate system behavior.

The **Design** subsection describes how the system was planned, covering User Interface (UI) design considerations, Database Design (including Entity-

Relationship Diagrams and schema/data dictionary), and Object-Oriented Design models like Class Diagrams relevant to the Django framework.

The **Implementation** subsection covers the actual building of the system, detailing the tools and technologies used, providing a description of the core modules developed, and outlining the testing strategies employed (including unit, integration, system, and user acceptance testing), along with sample test cases.

- **Chapter III: Conclusion and Recommendation**

This final chapter wraps up the project report. It includes a **Summary** of the entire project and its key activities. The **Conclusion** discusses whether the project objectives were met, the overall success of the system, and any significant findings or lessons learned during the development process. Finally, the **Recommendation** section suggests potential future enhancements and directions for “GanaSalesLite” that could further improve its functionality and benefit Ganapati Enterprise.

The report concludes with a **References** section, listing all the sources cited, and **Appendices**, which contain supplementary materials such as system screenshots, the project Gantt chart, and optional sample code snippets.

CHAPTER TWO: SYSTEM DEVELOPMENT PROCESS

2.1 Analysis

The analysis phase focused on thoroughly understanding the requirements for "GanaSalesLite," assessing its viability, and creating initial models to represent the system's intended functionality and scope.

2.1.1 Requirement Analysis

This sub-section details the process of identifying and documenting the needs that "GanaSalesLite" must fulfill for Ganapati Enterprise.

2.1.1.1 Functional Requirements:

Below is a list of functional requirements that the system must satisfy:

- Admin must be able to log in securely.
- Admin must be able to add, update, delete, and view products.
- Admin must be able to manage product categories.
- Admin must be able to add and manage customer information.
- Admin must be able to record new sales transactions.
- Admin must be able to add multiple items in a single sale.
- The system must automatically calculate the total bill.
- The system must update inventory levels after each sale.
- Admin must be able to generate and print bills.
- Admin must be able to view a list of sales.
- System must provide a dashboard with sales highlights.
- System must ensure secure logout functionality.
- System must restrict access to unauthorized users.

2.1.1.2 Non-Functional Requirements:

- **Performance:** The system should respond to user actions within 2–3 seconds.
- **Security:** User data and login credentials must be securely stored and protected.
- **Usability:** The interface must be user-friendly, suitable for a non-technical user.
- **Reliability:** The system should handle multiple operations without crashing.
- **Maintainability:** The codebase should follow best practices to allow future enhancements.

2.1.2 Feasibility Study

2.1.2.1 Technical Feasibility:

The system will be built using Python (Django Framework) for backend logic, PostgreSQL for database, and HTML/CSS/JS with Bootstrap for the frontend. Tools such as VS Code, Git, GitHub and pgAdmin are freely available and suitable for academic development. Technical knowledge is sufficient for implementation.

2.1.2.2 Operational Feasibility:

The system is easy to operate for the owner of Ganapati Enterprise. It reduces manual work, increases accuracy, and requires minimal training. The system enhances existing operations without disrupting them.

2.1.2.3 Economic Feasibility:

Costs are minimal as it uses open-source tools. Development is a part of the academic curriculum, so there's no labor cost. Long-term benefits include reduced human error and faster transactions.

2.1.3 Modelling

2.1.3.1 Use Case Diagram:

A Use Case Diagram illustrates the main interactions the Admin user has with the system.

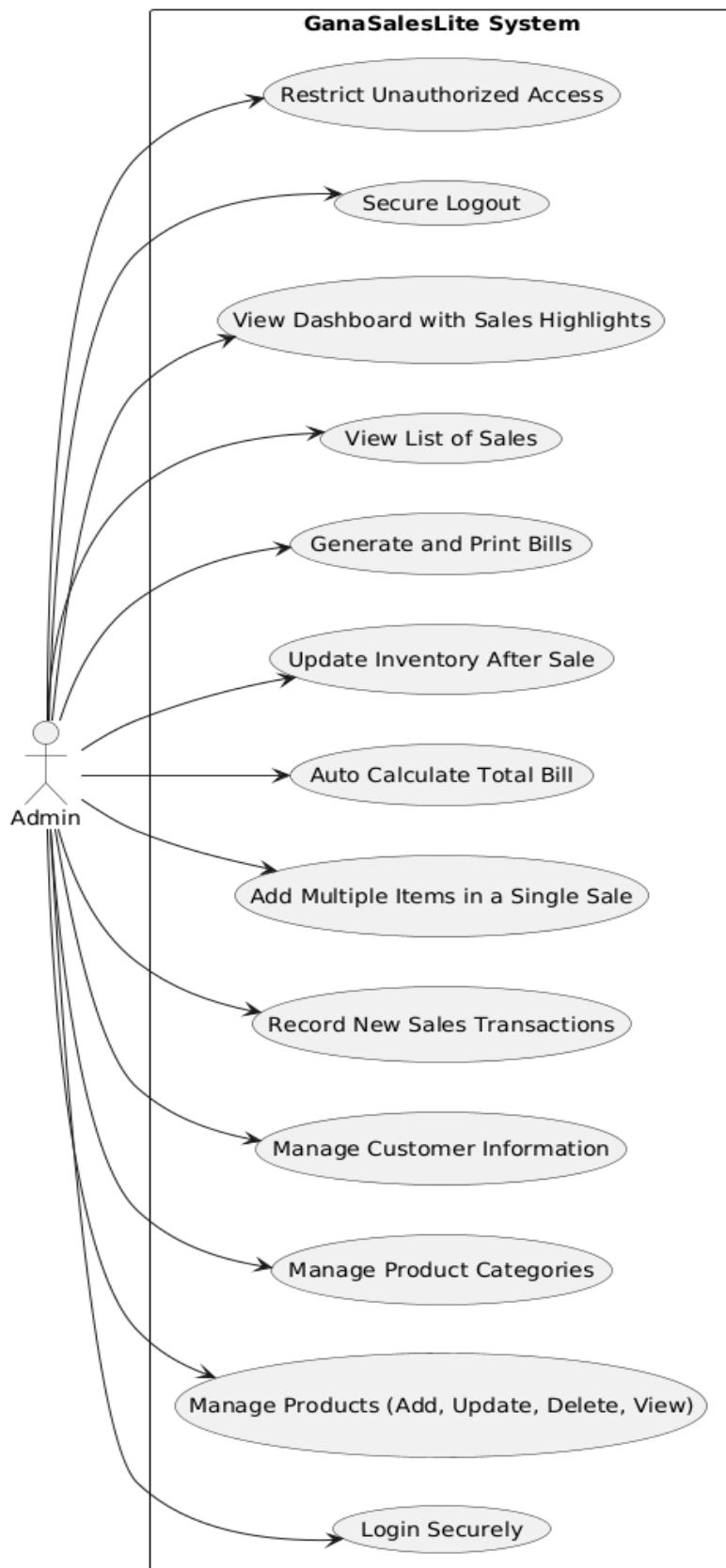


Figure 2: Use Case Diagram

2.1.3.2 Activity Diagram:

Activity diagrams were created to visualize important workflows:

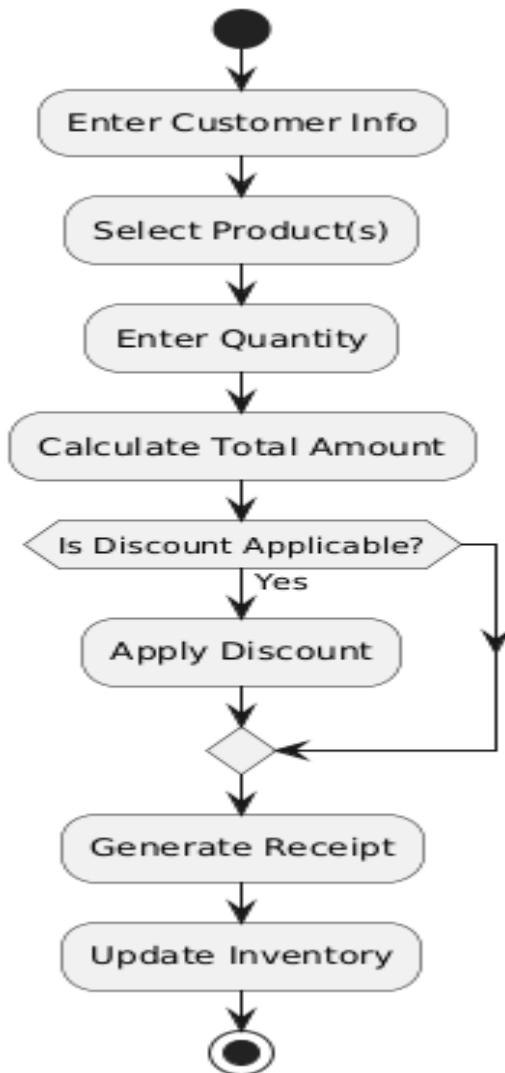


Figure 3: Sales Processing

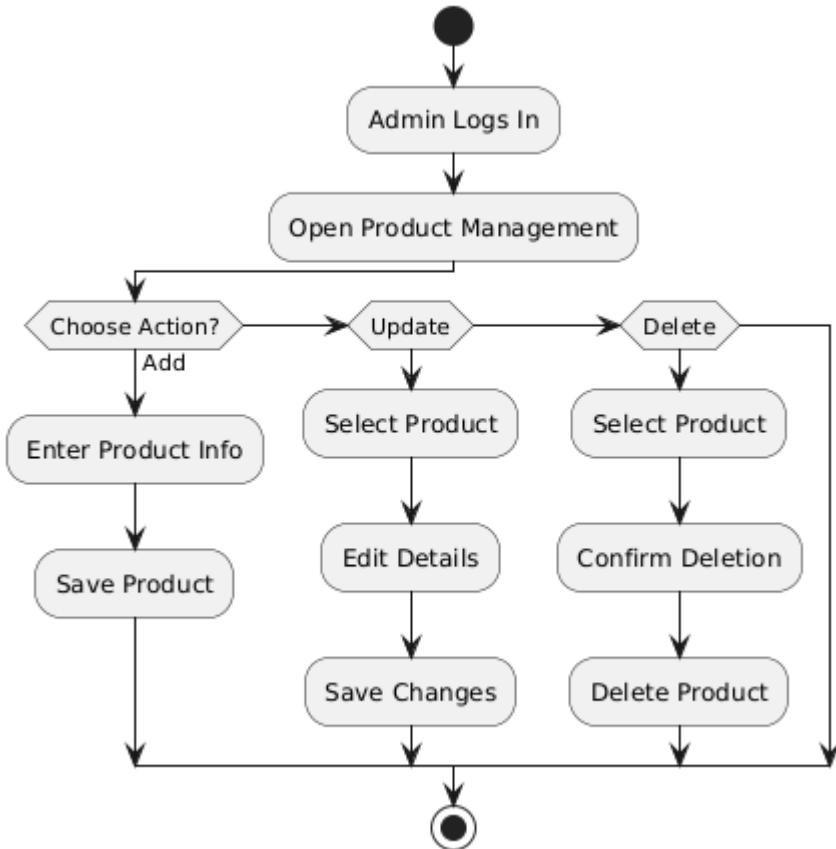


Figure 4: Activity for Product Management

2.2 Design

The design phase translates the analyzed requirements into a detailed blueprint for constructing "GanaSalesLite", including UI layout, database design, and class relationships.

2.2.1 User Interface (UI) Design

The UI design emphasizes simplicity, consistency, and usability for a single admin user. It uses **Bootstrap** for responsive design and includes:

- Clear navigation menus
- Easy-to-use forms for data entry
- Clean layout with color-coded sections

2.2.2 Database Design

2.2.2.1 Entity-Relationship (ER) Diagram:

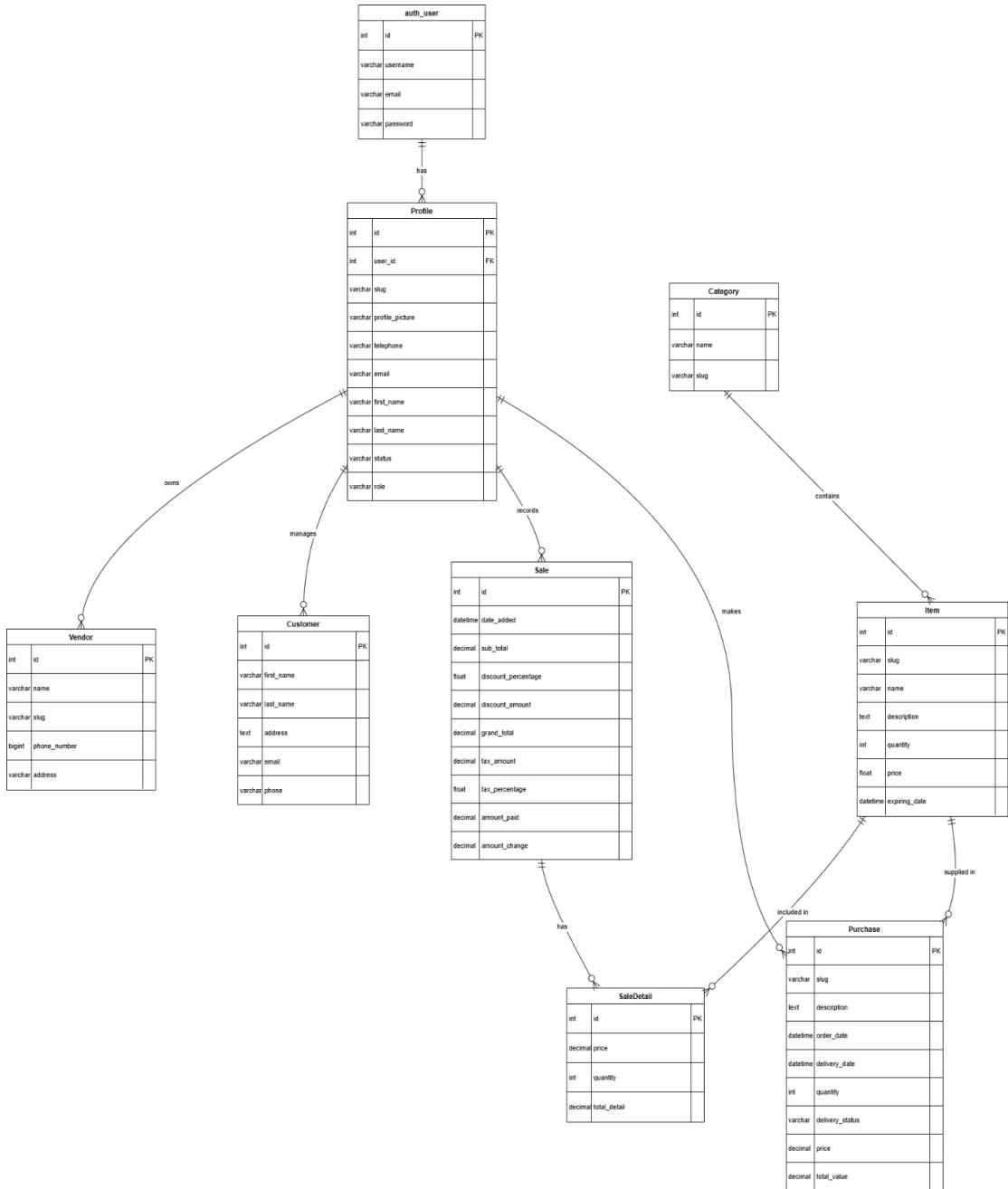


Figure 5:ER Diagram

2.3 Implementation

This section describes the actual construction phase of the "GanaSalesLite" system. It outlines the software tools and technologies utilized, details the functionalities and structure of the core modules developed, and discusses the testing strategies employed to ensure the system's quality and correctness. This phase translates the design specifications from section 2.2 into a working software application.

2.3.1 Tools and Technologies Used

The development of "GanaSalesLite" leveraged a combination of modern, open-source tools and technologies chosen for their suitability for web application development, robustness, and the development team's familiarity.

Table 1: Tools and Technologies Used in GanaSalesLite Development

Category	Tool/Technology	Purpose in Project
Backend Language	Python (Version 3.8+ recommended)	Core programming language for server-side logic and functionality.
Backend Framework	Django (Version 5.x recommended)	High-level Python web framework for rapid, secure development.
Database Management	PostgreSQL (e.g., Version 12+)	Relational database for persistent storage of all application data.
Frontend Technologies	HTML5, CSS3, JavaScript (ES6+)	Standard web technologies for content structure, styling, and interactivity.
CSS Framework	Bootstrap (Version 4.x or 5.x recommended)	For responsive, mobile-first UI with reusable components.
Asynchronous Operations	AJAX (via JavaScript or jQuery)	For dynamic content updates without full page reload.
Version Control System	Git	To manage code changes and collaboration.
Code Repository	GitHub	For hosting source code and collaboration.
Integrated Dev. Env.	Visual Studio Code (VS Code)	Main development editor with relevant extensions.
Web Server (Development)	Django Development Server	For testing and running the application locally.
Diagramming Tools	draw.io	Used to draw UML, ERD, and process diagrams.

2.3.2 Module Description

"GanaSalesLite" was developed using a modular approach, breaking down the system into distinct functional units. This enhances organization, simplifies development, and facilitates easier maintenance. The core modules of the system are described below:

2.3.2.1 Admin/User Authentication Module

- **Purpose:** Secure access for the system admin.
- **Key Functionalities:**
 - Login interface for username/password.
 - Credential validation.
 - Session management.
 - Logout functionality.
- **Implementation Notes:** Utilizes Django's built-in auth system.

2.3.2.2 Product Management Module

- **Purpose:** Manage product catalog.
- **Key Functionalities:**
 - Add/edit/delete products.
 - Manage product categories and vendors.
 - View/search products.
- **Implementation Notes:** Models: Product, Category, Vendor.

2.3.2.3 Customer Management Module

- **Purpose:** Maintain customer records.
- **Key Functionalities:**
 - Add/edit customer info.
 - View customer list.
 - Track purchase history.
- **Implementation Notes:** Model: Customer.

2.3.2.4 Sales and Billing Module

- **Purpose:** Record transactions and generate receipts.
- **Key Functionalities:**
 - Add products to sale.
 - Real-time total calculations.
 - Receipt generation.

- **Implementation Notes:** Models: Sale, SaleItem, JavaScript/AJAX for interactivity.

2.3.2.5 Inventory Overview Module

- **Purpose:** Monitor stock levels.
- **Key Functionalities:**
 - Auto-decrement stock on sale.
 - Stock level dashboard.
 - Reorder level alerts.
- **Implementation Notes:** Logic handled in Product model and views.

2.3.2.6 Reporting Module

- **Purpose:** Sales insights and reports.
- **Key Functionalities:**
 - Dashboard summaries.
 - Date-filtered sales reports.
 - Top-selling product stats.
- **Implementation Notes:** Uses Django ORM for data aggregation.

2.3.3 Testing

A thorough testing process was integral to the development of "GanaSalesLite" to identify defects, ensure functionalities align with requirements, and verify the overall quality of the application.

2.3.3.1 Unit Testing

- Objective: Test smallest code units (functions/models) for correctness.
- Tools Used: Django testing framework (unittest).
- Example: Validate product price calculations.

2.3.3.2 Integration Testing

- Objective: Test interactions between modules.
- Example: A sale updates product stock correctly.

2.3.3.3 System Testing

- Objective: Verify full system behavior.
- Environment: Local or staging environment.

2.3.3.4 User Acceptance Testing (UAT)

- Objective: Ensure usability and acceptance.
- Approach: Team members simulated admin tasks feedback was gathered and applied.

2.3.3.5 Sample Test Cases

Table 2: Sample Test Case – Login Module

Test Case ID	TC-LINKIN-001
Test Type	Functional
Module	Authentication
Test Title	Admin login with correct credentials
Precondition	Admin account exists
Input Data	Username: admin Password: correct pw
Expected Result	Redirect to dashboard
Actual Result	Dashboard successfully displayed
Status	<input checked="" type="checkbox"/> Pass
Comment	Basic login functionality verified

Table 3: Sample Test Case – Product Management

Test Case ID	TC-PROD-002
Test Type	Functional
Module	Product Management
Test Title	Add a new product
Precondition	Admin is logged in
Input Data	Product Name: "Biscuit" Price: Rs.50 Stock: 100 units
Expected Result	Product added and visible in product list
Status	<input checked="" type="checkbox"/> Pass
Comment	Validated input values price > 0, stock recorded properly

Table 4: Sample Test Case – Sales and Billing Module

Status	<input checked="" type="checkbox"/> Pass
Comment	Validated input values price > 0, stock recorded properly
Status	<input checked="" type="checkbox"/> Pass
Comment	Validated input values price > 0, stock recorded properly
Status	<input checked="" type="checkbox"/> Pass
Comment	Validated input values price > 0, stock recorded properly
Status	<input checked="" type="checkbox"/> Pass
Comment	Validated input values price > 0, stock recorded properly
Status	<input checked="" type="checkbox"/> Pass
Comment	Validated input values price > 0, stock recorded properly
Status	<input checked="" type="checkbox"/> Pass
Comment	Validated input values price > 0, stock recorded properly

Table 5: Sample Test Case – Reporting Module

Test Case ID	TC-REPORT-001
Test Type	Functional
Module	Reporting
Test Title	Generate daily sales report
Precondition	At least one sale recorded today
Input Data	Date Range: Today
Expected Result	Report displays summary of today's sales
Actual Result	Report shows 10 transactions, Total Sales: Rs.1500
Status	<input checked="" type="checkbox"/> Pass
Comment	Daily total and transaction count correctly shown

CHAPTER THREE: CONCLUSION AND RECOMMENDATION

This chapter serves as the culmination of the project report for "GanaSalesLite: A Sales and Customer Management System for Ganapati Enterprise." It begins by summarizing the entire project, then presents the overall conclusions derived from the development lifecycle and the final system's capabilities. Finally, it offers actionable recommendations for potential future enhancements and expansions of the "GanaSalesLite" system.

3.1 Summary

The "GanaSalesLite" project was undertaken with the primary objective of designing and developing a tailored, user-friendly, web-based Sales and Customer Management System for Ganapati Enterprise. The core motivation was to address the inefficiencies and limitations of existing manual or overly complex systems for a single-admin/owner operation. The system aimed to automate key business processes including product management, customer record keeping, sales transaction entry, billing, basic inventory overview, and sales reporting.

The project adhered to the **Rapid Application Development (RAD)** model. This involved a systematic progression through:

- **Analysis:** Where requirements (functional and non-functional) were meticulously gathered and defined, a feasibility study (technical, operational, economic) was conducted to ensure project viability, and initial system models (Use Case Diagrams, Activity Diagrams) were created to conceptualize the system's scope and behavior.
- **Design:** This phase translated the analyzed requirements into a comprehensive blueprint. It included the design of an intuitive User Interface (UI) suitable for the target user, detailed Database Design (manifested in an Entity-Relationship Diagram and Data Dictionaries/Schema), and the creation of Object-Oriented Design Models (specifically Class Diagrams representing the Django application models).
- **Implementation:** The actual construction of "GanaSalesLite" was carried out using Python with the Django framework for the backend logic, PostgreSQL as the relational database (with SQLite utilized during development), and a frontend built with HTML, CSS (enhanced by Bootstrap), and JavaScript. The system was developed in a modular fashion, with distinct units for Admin Authentication, Product Management, Customer Management, Sales and Billing, Inventory Overview, and Reporting.
- **Testing:** A multi-faceted testing strategy was implemented, encompassing unit testing for individual components, integration testing to verify module

interactions, system testing for end-to-end functionality to ensure the system met the end-user's needs and expectations.

The outcome of this endeavor is "GanaSalesLite," a functional and targeted application designed to streamline the daily operations of Ganapati Enterprise.

3.2 Conclusion

The "GanaSalesLite" project has successfully achieved its intended goals, delivering a functional Sales and Customer Management System specifically tailored to the operational context and single-user requirement of Ganapati Enterprise. The system effectively addresses the core problems associated with manual record-keeping by providing an automated, centralized, and user-friendly platform.

From the development process and the final product, the following key conclusions can be drawn:

1. **Fulfillment of Objectives:** The system successfully implements the specific objectives outlined, including secure admin access, comprehensive CRUD operations for products and customers, an efficient sales and billing process, automatic inventory updates tied to sales, Reorder Level (ROL) notifications, and the generation of essential sales reports.
2. **Enhanced Operational Efficiency:** "GanaSalesLite" provides Ganapati Enterprise with the means to significantly reduce time spent on manual data entry and retrieval, minimize the potential for human error in calculations and record-keeping, and streamline overall daily workflows.
3. **Improved Data Management and Accessibility:** The centralized PostgreSQL database ensures that all critical business data (products, customers, sales) is stored in an organized, secure, and easily accessible manner for the admin/owner.
4. **User-Centricity and Usability:** The focus on a simple, intuitive UI has resulted in a system that should be readily adoptable by the owner of Ganapati Enterprise, even with limited prior technical expertise.
5. **Effectiveness of Chosen Technology Stack:** The selection of Python/Django, PostgreSQL, and Bootstrap proved to be an effective combination, facilitating relatively rapid development, providing robust built-in features (especially from Django for security and ORM), and enabling the creation of a responsive web interface.
6. **Demonstrated Value for Micro-Enterprises:** This project underscores the significant value that even a relatively simple, custom-developed software solution can bring to micro or small-scale enterprises by providing tools that are directly aligned with their specific needs and operational scale, often being more appropriate than larger, more complex commercial systems.

7. **Valuable Learning Experience:** The project has provided an invaluable learning experience for the development team, encompassing the entire software development lifecycle, from requirements engineering through design, coding, testing, and documentation.

While "GanaSalesLite" operates within defined limitations (such as its single-user design and basic reporting capabilities), these were intentional choices to maintain project focus and align with the immediate requirements of Ganapati Enterprise. The system provides a solid and reliable foundation that successfully digitizes and improves core business processes.

3.3 Recommendation

"GanaSalesLite" currently serves as an effective tool for Ganapati Enterprise. However, to further enhance its capabilities, support potential business growth, and broaden its utility, the following recommendations for future enhancements are proposed:

- 1. Multi-User System with Role-Based Access Control (RBAC):**

As the business might expand, an essential upgrade would be to support multiple user accounts (e.g., for a sales assistant, inventory manager). RBAC would allow the primary admin to assign specific permissions to different users, controlling their access to various system modules and functionalities.

- 2. Advanced Reporting and Business Intelligence:**

Integrate more sophisticated reporting tools to provide deeper insights. This could include graphical data visualization (charts for sales trends, customer demographics, product performance), customizable report generation by the user, and options to export reports in formats like CSV or PDF for external use or record-keeping.

- 3. Comprehensive Inventory Management:**

Expand the inventory module to include features like purchase order management for restocking from vendors, tracking supplier information more extensively, managing multiple supplier options per product, and tools for periodic stock-taking and variance reconciliation. For businesses dealing with perishable goods, adding expiry date tracking and first-in-first-out (FIFO) suggestions would be beneficial.

- 4. Enhanced Mobile Accessibility/Progressive Web App (PWA):**

Improve the mobile responsiveness significantly or develop "GanaSalesLite" as a Progressive Web App (PWA) to provide a near-native app experience on mobile devices, including offline access for certain functionalities if feasible. This would allow the owner to manage aspects of their business on the go.

- 5. Automated Data Backup and Cloud Integration:**

Implement an automated data backup schedule and an easy-to-use restore function. Consider options for cloud-based backup or even deploying the entire application on a secure cloud platform for better data safety and remote accessibility.

6. Integration with Communication Tools:

Allow for basic customer communication (e.g., sending out promotional SMS/emails, if consent is obtained) directly from the customer management module.

7. Barcode Functionality:

Integrate support for barcode scanners to speed up product selection during sales and for more efficient inventory management tasks like stock receiving and audits.

8. Audit Trails:

Implement an audit trail feature to log key system activities (e.g., who made changes to product prices, who deleted a sale record if that functionality is added for admins), which can be important for accountability and tracking.

9. Customizable Dashboard:

Allow the admin to customize the information and widgets displayed on their main dashboard to better suit their immediate priorities.

REFERENCES

Deloitte (2012), The Changing face of retail Retrieved 18, Jun 2013 from <http://www.deloitte.com/assets/DcomUnitedKingdom/Local%20Assets/Documents/Industries/consumer%20Business/uk-cb-store-of-the-future-report.pdf>

Tim Crosby (2007), How Inventory Management Systems Work Retrieved 22, Jun 2013 from <http://money.howstuffworks.com/how-inventory-management-systems-work1.htm>

Wikipedia (2013), Inventory Management Software Retrieved 22 Jun 2013 from http://en.wikipedia.org/wiki/Inventory_management_software

Anton Dolinsky (2007), Barcodes, sales and inventory control Retrieved 22 June 2013 <http://www.almyta.com/InventoryManagementHistory4.asp>

EvrenSahin & Yves Dallery (2010), The impact of the use of RFID on Supply chain operations Retrieved 30 July 2013 from <http://ebajic.free.fr/RFID%20Forum/Papers%20submitted%20but%20not%20presented/A%20LITERATURE%20REVIEW%20ON%20THE%20POTENTIAL%20OF%20THE%20RFID%20TO%20TACKLE%20INVENTORY%20MANAGEMENT%20ISSUES.pdf>.