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

E-Commerce Data Management and Analytics Using a Lightweight Flask–Sqlite Web Application

E. Sumanth Raju

Department of Computer Science and Engineering, Hyderabad Institute of Technology and Management, Hyderabad, Telangana, India

Abstract

Small e-commerce operations need to maintain product, customer and order information while monitoring sales and inventory. This paper describes the design and implementation of a lightweight E-Commerce Data Management System (EDMS) developed using Python, Flask, SQLite and Jinja2. The application provides administrator authentication, modules for managing products, customers and orders, and a dashboard presenting total revenue, order count, customer count, low-stock count, a monthly sales trend, an order-status breakdown and low-stock alerts. In a local demonstration using sample data, the dashboard displayed Rs. 39,860.00 in revenue, 15 orders, 5 customers and 4 low-stock items. Evaluation is limited to the reported sample dataset and local demonstration; live-business performance, multi-user scalability and production security were not established.

Keywords: E-commerce; data management; analytics dashboard; Flask; Python; SQLite; inventory management; order management.

I. INTRODUCTION

E-commerce activities generate records related to products, customers, orders and stock. These records support daily operational decisions, including order tracking, revenue monitoring and replenishment. In smaller operations, information may be kept in separate spreadsheets or disconnected tools, making consolidated reporting difficult. This project addresses the problem through an administrator-oriented web application. It focuses on operational records rather than a customer-facing storefront or online payment gateway.

II. RELATED APPROACHES AND PROBLEM IDENTIFICATION

Spreadsheet-based record keeping is accessible but can become difficult to maintain as records grow. ERP platforms provide structured business modules but may be complex for small operations. Business intelligence platforms provide visual summaries but depend on clean, structured source data. The project adopts a compact application architecture using a lightweight web framework and an embedded relational database. The problem addressed is the fragmentation of product, customer and order information and the resulting difficulty of obtaining consistent revenue, order-status and stock summaries.

III. SYSTEM DESIGN AND METHODOLOGY

The backend is implemented in Python using Flask 3.x. SQLite is used as the relational data store, while Jinja2 templates render the browser interface. HTML, CSS and JavaScript support the front end. The browser sends requests to app.py, which handles routing, authentication and business logic. Database operations are delegated to database.py, which reads and writes ecommerce.db. The application returns rendered pages through Jinja2 templates.

A. Data Model and Development Process

The data model contains product, customer, order and administrator information. Product records include a name, category, stock quantity and reorder level. Customer records allow orders to be associated with customers. Orders include a date, value and status: Pending, Shipped, Delivered or Cancelled. Development followed four phases: requirement analysis and database design; backend and database-layer implementation; construction of the browser interface and dashboard; and testing with sample records and preparation of documentation.

IV. IMPLEMENTATION

The project is organized into app.py, database.py, ecommerce.db, templates/, static/, requirements.txt and README.md. The authentication module presents a username and password form and requires an authenticated session to access protected pages. The Products, Customers and Orders pages provide administrator-facing record management. The dashboard displays four key figures, a monthly sales trend chart, an order-status breakdown chart and a low-stock product table.

V. RESULTS AND DISCUSSION

The application was run locally using sample data. The dashboard displayed Rs. 39,860.00 in total revenue, 15 orders, 5 customers and 4 low-stock items. The low-stock table listed Bluetooth Headphones (stock 8; reorder level 10), Denim Jacket (6; 10), Fiction Novel Set (5; 10) and Flair Blue Pens (10; 100). The dashboard also presented a monthly sales trend and an order-status breakdown across Cancelled, Delivered, Pending and Shipped orders. These values describe the sample demonstration and are not results from a live business.

A. Limitations

Testing was performed with a small sample dataset containing 15 orders and 5 customers. The report does not establish performance under live workloads, simultaneous multi-user access or public deployment. The local Flask development server and SQLite database were used for demonstration. Demonstration credentials should be replaced with properly managed credentials before real-world use. No quantitative usability study, load test, security audit or forecasting evaluation is reported.

VI. CONCLUSION AND FUTURE WORK

This paper presented a lightweight E-Commerce Data Management System implemented with Python, Flask, SQLite and Jinja2. It combines administrator authentication, product/customer/order management and a dashboard of descriptive business indicators. The sample-data demonstration shows the integration of record management and summary analytics in a single application. Future work includes production-grade deployment, multiple user roles, securely managed accounts, migration to MySQL or PostgreSQL, downloadable CSV/PDF reports, date and category filters, low-stock notifications and demand forecasting. These are proposed enhancements, not implemented features.

REFERENCES

1. Pallets Projects, "Flask Documentation." https://flask.palletsprojects.com (2026).
2. Python Software Foundation, "Python 3 Documentation." https://docs.python.org/3/ (2026).
3. SQLite Consortium, "SQLite Documentation." https://www.sqlite.org/docs.html (2026).
4. Pallets Projects, "Jinja Template Designer Documentation." https://jinja.palletsprojects.com (2026).
5. Smaran F, Internship guidance and project material, Nexxoft Infotel, Bengaluru, India, 2026.
6. Editorial note: This manuscript is prepared from the supplied internship report. It is a draft for review, not evidence of acceptance or publication by a journal or conference.