RETAIL POINT-OF-SALE (POS)

Retail Point of Sale and Inventory Management Systems: A Literature Review and Proposed Integrated Approach for DailyMart

Jayvardhan Deepak Kamat, Siddhi Dattatray Joshi, Sahil Shantinath Magadum, Shrutika Sagar Chavan, Aniket Thorat

Department of Computer Science and Engineering, SGI Institute

Published November 2026

Abstract

A retail point-of-sale (POS) system connects each sale with stock records, replenishment, tax documents and store reports. This review looks at recent research on inventory accuracy, POS data, replenishment, perishable goods and digital stock visibility. It then considers how those findings apply to DailyMart, a proposed platform for multiple retail outlets in India. We examined 21 research papers published between 2020 and 2025, comparing their methods, contributions and limitations. The studies show that transaction data can help with audits and forecasting, but a digital record cannot reveal every physical loss. Expiry details and differences in the way individual stores work create further challenges. Most of the papers focus on one decision, such as replenishment, shelf allocation, food-waste reduction or sharing stock across channels. Fewer explain how a practical retail application should connect checkout, purchasing, stock reversals, tax invoices and receipt delivery. DailyMart aims to address this gap by linking sales and stock updates with goods receipts, supplier returns, outlet-level access, expiry alerts, traceable cancellations and a separate messaging service. This is a proposed design, not proof of better performance. We suggest tests for transaction accuracy, simultaneous checkout, expiry tracking and usability.

Keywords: Point of sale; retail inventory; inventory accuracy; replenishment; perishable goods; GST

1 Introduction

Retailers deal with a few basic questions every day: what has sold, what is still in stock, what needs to be ordered, and which items are close to expiry. When billing, inventory and purchasing are maintained separately, staff often have to match the records manually. That can lead to delayed corrections, stockouts or orders based on outdated figures. A POS system gives the store one shared record of transactions. For it to be useful, the related stock and purchasing records must update along with each sale.

DailyMart is a web-based retail operations project planned for supermarkets, grocery stores, cafés and businesses with multiple branches. The project brief describes a React interface, an ASP.NET Core API, PostgreSQL accessed through Npgsql and Dapper, and a separate Node.js gateway for WhatsApp receipts. Its listed features include barcode checkout, saved carts, GST-oriented invoices, purchase orders, goods receipts, supplier returns, customer records, reports, store settings and role-based access. Refunds, cash-drawer hardware, card terminals and the official Meta Cloud API are planned for a later stage.

This review looks at how recent retail research deals with POS data, inventory accuracy, replenishment and perishable stock, and what a student-developed integrated POS can take from that work. It brings together 21 studies, compares their methods and limitations, identifies a research gap and outlines a possible system response. It does not claim that DailyMart is faster, more accurate or more compliant than existing systems. Those points can only be established through testing.

1.1 Review method

We carried out a focused narrative review rather than a formal systematic review. Scholarly databases, publisher websites and university repositories were searched for studies published from 2020 to 2025. We considered papers on retail POS data, record accuracy, replenishment, shelf availability, perishable goods, food waste, stock visibility and store operations. For each paper, we noted its method, main contribution and any limitation that matters to an integrated POS project. The selected papers help explain the main themes, but they are not a complete count of all research in this area. We used available abstracts, article text and accepted manuscripts. The references direct readers to publisher records or DOI pages.

2 Literature review

2.1 POS data and inventory accuracy

Qi, Mak and Shen review data-driven research on product assortment, order fulfilment and inventory management. Their work explains how transaction data can support retail decisions, while also showing that the data must be connected to the policies stores actually follow [1]. Bassamboo, Moreno and Stamatopoulos look at audit and replenishment decisions when spoilage, damage or theft leave the system showing stock that is no longer available. Their algorithm can come close to the optimum when a service-level threshold is used [2]. For a POS project, the key point is simple: sales history cannot explain every missing item on a shelf.

Shabani et al. use data from a large fashion retailer to measure inventory record inaccuracy and distinguish it from sales-management performance [3]. Their findings show that a quantity stored in a digital system may still be incorrect. Falcão et al. take a sensor-based approach. Their FAIM prototype combines shelf-weight changes, visual recognition and product-layout information to track item movement in a convenience store [4]. It improves automatic identification in the test setting, but it needs extra equipment and covers only a limited set of interactions. Andaur et al. use POS data and variables shared by retailers to predict out-of-stock events in packaged foods [5]. DailyMart does not currently include this type of predictive report, although the study suggests a possible future use for dependable transaction histories.

These studies point to two different ways of checking stock. A POS ledger records events entered by the store, including sales and goods receipts. Physical stock counts or extra sensors are needed to find losses that were never recorded. DailyMart should keep a clear transaction history and make stock audits practical. Its ledger should be described as recorded stock, not as a guarantee of what is physically on the shelf.

2.2 Replenishment and store operations

Avlijas, Vukanovic Dumanovic and Radunovic compare manual and automatic ordering in 85 stores, covering 95 products. In the stores they studied, automatic ordering improved product availability,

particularly for higher-risk products and locations [6]. Dharmawardane, Sillanpää and Holmström examine frequent grocery forecasts. They describe a move away from repeatedly recalculating models and towards monitoring forecast errors [7]. Both studies point to a practical need: replenishment tools rely on timely, dependable data, and staff must be able to investigate unusual results.

A recent sales total is not enough to decide what a store should order next. Taleizadeh et al. compare vendor-managed inventory policies in situations where some demand is backordered while other sales are lost [8]. Kim and Moon bring product assortment, shelf space and replenishment together in a nonlinear optimization model [9]. Das, Carrel and Caplice study how case-pack sizes affect inventory levels and backroom storage [10]. Supplier terms, shelf capacity and store layout all influence the final decision. DailyMart's purchase-order and goods-receipt workflows record these activities, but automated forecasting and order optimization are outside the stated first phase.

Research on food waste brings the discussion closer to day-to-day store work. De Moraes et al. identify causes of retail food waste and review ways to reduce it [11]. Huang et al. examine retailer practices using the ideas of sustainable value and the 5R framework [12]. Riesenegger and Hübner look at proactive planning in stores [13], while Richards and Hamilton use distributor data to study beer losses and the effects of case size, product assortment and delivery decisions [14]. Better records are useful, but they do not reduce waste by themselves. Staff still need a practical response, such as rotating stock, changing an order, discounting an item or arranging a donation.

2.3 Perishable inventory and expiry visibility

Mallidis et al. develop a replenishment model that also considers when surplus perishable goods should be redistributed [15]. Herron et al. combine pallet-temperature experiments with Monte Carlo modelling to estimate how much shelf life remains after temperature abuse during last-mile delivery [16]. Their work highlights that items in the same delivery may have different temperature histories. Saghiri, Aktas and Mohammadipour propose measures for waste, lost sales and freshness in omnichannel perishable inventory. They identify shelf-life visibility and changes in demand as important factors [17].

Hansen, Misra and Sanders study how customers choose between products carrying different expiry dates. Shelf position and discounts can influence which item a shopper selects [18]. Chehrazi, Sanders and Stamatopoulos examine how incomplete stock and expiry information can restrict price changes for perishable groceries [19]. For DailyMart, showing an expiry alert does not mean the system follows FEFO (first expire, first out). To apply FEFO at checkout, the system needs batch-level records. Each goods receipt's expiry date and remaining quantity must be tracked as stock is sold or adjusted. The project brief mentions shelf-life settings and an expiry radar, but the team still needs to check whether the database and checkout process allocate stock by batch.

2.4 Multi-outlet and channel integration

Goedhart, Haijema and Akkerman use a Markov decision process to study replenishment and stock rationing for an omnichannel retailer [20]. Silbermayr and Waitz examine perishable products shared between online and physical stores, including substitution and transshipment [21]. Neither study is about the cashier's screen. They do show, however, why a retailer needs to know which outlet or channel can use each unit of stock. Different demand channels may compete for the same items, and each can have its own service expectations.

Researchers have developed detailed methods for forecasting, stock audits, shelf allocation, waste reduction, expiry-based choices and channel rationing. Most of these studies deal with one specific decision. DailyMart is closer to the application level: its proposed design connects checkout, inventory, purchasing, invoice history and outlet records. The research can help shape its controls and testing, but it does not establish that DailyMart has already improved store performance.

2.5 Cross-study synthesis

A few common findings emerge from the studies. Transaction data are useful, but they cannot account for every physical item in a store. Sales and goods receipts explain recorded stock movements; audits or sensors are needed to identify shrinkage and other changes that were never entered into the system [2]–[5]. Replenishment also depends on more than recent sales. Forecast timing, pack sizes, supplier arrangements, promotions and available space all affect product availability [6]–[10]. A single reorder threshold may not work for every product or outlet. Perishable goods need age information as well as quantity. Expiry records become useful when they lead to action, such as stock rotation, discounting or redistribution, which can help reduce waste and protect freshness [11]–[19].

The studies do not all provide the same kind of evidence. Analytical models explain trade-offs under specific assumptions, but they do not show whether store staff can follow a policy during a busy shift. Retailer datasets and case studies offer practical detail; still, a result from one chain or product category may not apply elsewhere. Sensor prototypes can detect movements that a ledger misses, although they bring equipment and maintenance costs. Reviews summarize practices across a field, while field experiments can isolate how customers or staff respond to a particular process. For a practical POS project, a dependable transaction ledger and clear audit steps are a sensible starting point. No single published model solves every store's problems.

For DailyMart, the findings support a staged approach. The first task is to make recorded events reliable: checkout, goods receipts, returns and cancellations should update their related records together. Store-level reports and useful expiry alerts can follow. Forecasting, automatic ordering, sensors and dynamic pricing can be considered later, once the transaction history can be trusted and the retailer knows how to act on the recommendations. This keeps the first phase within the project's stated scope and gives future analytics a better foundation.

Table 1 compares the recent retail inventory and perishable-goods studies reviewed.

Table 1. Comparison of recent retail inventory and perishable-goods studies.

Study / method	Contribution	Limitation
[1] Qi et al., 2020: Review of data-driven retail operations	Maps research on assortment, fulfilment and inventory; gives a broad overview.	Analytical review; does not set out an integrated POS design.
[2] Bassamboo et al., 2020: Optimization model using POS and audit data	Studies auditing and replenishment together when recorded stock may be inaccurate.	Assumes inventory is known after replenishment; addresses a specific decision.
[3] Shabani et al., 2021: Network DEA using retailer data	Measures store-level record errors and inventory-management performance.	Uses data from one fashion retailer and does not design checkout processes.

Study / method	Contribution	Limitation
[4] Falcão et al., 2020: Vision and shelf-weight sensor prototype	Combines sensor inputs to identify when products are picked up or returned.	Equipment costs and limited prototype conditions restrict use in a basic POS.
[5] Andaur et al., 2021: Machine-learning out-of-stock prediction	Uses POS and retail data to identify or predict out-of-stock events.	Predictive model from a manufacturer's perspective, not a complete store system.
[6] Avlijas et al., 2021: 85-store empirical comparison	Examines the effect of automatic replenishment on product availability.	Based on one chain and sample; automation still relies on reliable store data.
[7] Dharmawardane et al., 2021: Forecasting intervention in grocery retail	Studies frequent SKU-store forecasts and the monitoring of forecast errors.	Focuses on forecasting, not transaction accuracy or purchasing workflows.
[8] Taleizadeh et al., 2020: VMI analytical replenishment model	Compares reorder policies when some demand is backordered and some sales are lost.	Uses a simplified vendor-buyer setting rather than a working retail application.
[9] Kim and Moon, 2021: MINLP and heuristic optimization	Brings product assortment, shelf allocation and replenishment into one model.	Does not cover operational data capture or stock auditing.
[10] Das et al., 2021: Analytical inventory/space model	Shows how pack size affects inventory and backroom storage requirements.	Single-product assumptions make direct use across many categories difficult.
[11] de Moraes et al., 2020: Literature review of retail food waste	Identifies common causes of retail food waste and ways to reduce it.	Broad review; does not define an integrated POS data structure.
[12] Huang et al., 2021: Review of retailer waste practices	Groups retailer waste-reduction practices using the 5R and sustainable-value ideas.	Actual implementation depends on the retailer and its local conditions.
[13] Riesenegger and Hübner, 2022: Exploratory multi-case study	Highlights store-planning approaches that may help prevent food waste.	Cases focus on German grocery and bakery settings; other settings need study.
[14] Richards and Hamilton, 2022: Empirical distributor-data analysis	Links assortment, case size and operating decisions with beer losses.	Findings from one product and distributor context may not apply to all retail.
[15] Mallidis et al., 2020: Single-period optimization model	Plans replenishment and redistribution when perishable stock is left over.	Uses a limited planning period and depends on model and data assumptions.
[16] Herron et al., 2022: Temperature experiment and Monte Carlo	Estimates the effect of pallet position and temperature abuse on remaining shelf life.	Focuses on chicken and simulated transport conditions; requires suitable data and sensors.
[17] Saghiri et al., 2023: Industry interviews, review and simulation	Sets out measures for waste, lost sales and freshness in omnichannel perishable goods.	Mainly conceptual and simulation-based; does not evaluate a POS workflow.
[18] Hansen et al., 2024: Field data and consumer-choice model	Examines how expiry dates affect shopper choice, stock rotation and discounts.	Limited to one retailer/category context and needs item-level expiry data.
[19] Chehrazi et al., 2025: Inventory-information theory and pilots	Connects inventory visibility with decisions about changing prices for perishables.	Focuses on pilots and pricing, not the full purchasing and billing process.

Study / method	Contribution	Limitation
[20] Goedhart et al., 2022: Markov decision process model	Studies how inventory should be allocated and replenished across sales channels.	Presents a policy model rather than a working POS implementation.
[21] Silbermayr and Waitz, 2024: Perishable omnichannel inventory model	Examines substitution and stock transfers between online and physical stores.	Uses a two-product model and does not cover store transactions or tax records.

3 Research gap

The studies reviewed explain why POS data, accurate stock records, replenishment analysis and expiry visibility matter. Most, however, look at only one part of the retail process. POS research often focuses on audits or forecasting. Replenishment papers concentrate on order decisions, while perishable-goods studies examine expiry, waste or customer choices. Sensor-based studies depend on specialized equipment. Taken separately, these papers do not explain how a small outlet can maintain one traceable record from purchase order and goods receipt to checkout, invoice storage, cancellation and supplier return.

The quantity shown in the system is not always the quantity a customer can buy. Sales and goods receipts keep the ledger updated, but damage, spoilage, theft, counting errors and transfers between outlets can still create differences. Many expiry models need the age of each item or batch; a general shelf-life setting for a product cannot guarantee FEFO. Multi-outlet operation brings another issue: staff should only see or change records for the stores they are permitted to manage. Receipt messaging also depends on an outside service. If WhatsApp is unavailable, a sale should still be saved.

The gap identified in this review is mainly about integration and evaluation. A practical web application should connect the main retail documents with stock movements, include transaction and audit controls, and clearly state which functions are not yet covered. This gap comes from comparing the reviewed studies with DailyMart's requirements. It does not mean that similar commercial products are unavailable.

One useful way to test the system is to ask whether a manager can trace a stock figure back to the transactions that changed it. In DailyMart, a purchase order should connect to the quantities actually received, an invoice should identify the items sold, and every return or cancellation should include a reason. This makes the stock ledger easier to understand. It still cannot reveal physical losses that were never recorded, so regular counts and reviews of discrepancies remain necessary. Expiry alerts can remind staff to act, but the team should only claim strict FEFO after batch quantities are tracked and allocated during checkout.

4 Proposed DailyMart approach

DailyMart is planned as a role-based web application for multiple outlets. It has four main parts: an interface for cashiers and managers, an ASP.NET Core API, a PostgreSQL database and a separate WhatsApp receipt service. Cashiers can scan barcodes and prepare carts. The server then creates invoice headers and line items, assigns a document number for the financial year and updates stock within one database transaction. If any of these database operations fails, the transaction should roll back completely. After it commits, the invoice becomes the saved record for printing and later reference.

Purchasing records should be linked directly to inventory. A purchase order stores what the outlet requested, while a goods receipt records what arrived and was accepted. Supplier returns should include a standard reason and a debit-note document. Product and store IDs help keep catalogue and stock records with the correct outlet. When an invoice is cancelled, the original invoice and its reason should remain in the system. Stock should then be reversed through a controlled, idempotent operation, so repeating the request cannot restore the same units twice. Tax fields and slabs may be configurable, but each retailer must check that the settings fit its situation and current GST requirements.

DailyMart can use shelf-life settings and recorded dates to flag products that are nearing expiry. Strict FEFO requires more detailed tracking: every inward batch must retain its expiry date and remaining quantity as items are sold or adjusted. The project brief includes expiry monitoring, but the team should check the database structure and checkout logic before claiming that batch-level control is in place. WhatsApp receipts should be sent only after the sale has been committed. A messaging failure must not cancel the invoice. Managers can then use reports to review sales, payment splits, supplier activity and stock alerts.

A sale should be treated as complete only after the database transaction commits. Before that happens, the server checks the outlet, product status, available quantity and payment choice, then saves the invoice and stock movement together. If a database write fails, the whole operation should roll back and return an error that the cashier can handle. Printing and WhatsApp delivery happen after the commit. This separation is important: a printer or messaging failure should not create another sale when staff retry the follow-up step.

System access should reflect each person's responsibilities in the store. Cashiers need to search for products, prepare carts, take payments and view invoices. Inventory staff manage purchase orders, inward receipts and supplier returns. Managers check outlet reports and expiry alerts. Administrators handle tax settings, users, outlets and sensitive cancellations. The API must verify both the user's role and the outlet they are allowed to access on every request. Hiding a button on the screen is not enough to secure an action. Each sensitive change should leave an audit record showing who made it, when it happened and why.

For Indian retailers, DailyMart is intended to prepare invoice fields and tax splits based on the outlet and transaction settings. Before using it in a live store, the retailer must verify its registration details, place-of-supply conditions, tax rates and document format. The papers reviewed in this study deal with retail operations and inventory; they do not confirm GST compliance. The tax module should therefore be treated as a software feature that still needs review by the project guide or a qualified tax professional.

Figure 1 shows the DailyMart system flow and its service boundaries.

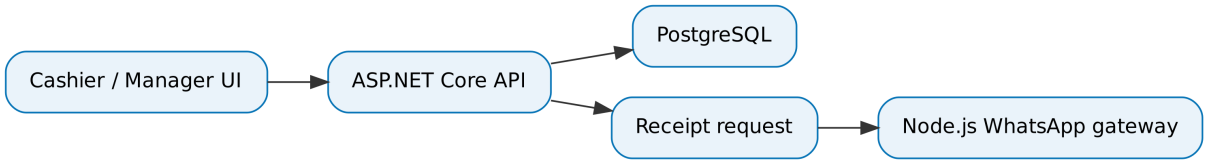


Figure 1. DailyMart system flow and service boundaries.

4.1 Core integrity controls

The integration also needs safeguards that work behind the user interface. Invoice creation and the related stock movement should be handled in one database transaction, while a unique constraint should prevent duplicate document numbers. SQL values should be passed through parameterized queries. The API must check user permissions and outlet access, and receipt and cancellation endpoints should prevent the same request from being processed twice. Sensitive changes need an audit entry with the user, time and reason. Before deployment, the team should test backup and restore procedures, monitor service health and protect application secrets.

4.2 Evaluation plan

Testing should check whether expected stock quantities match the quantities stored after sales, full or partial goods receipts, supplier returns and cancellations. For concurrency testing, several cashier sessions can try to purchase the last available units at the same time; the accepted invoices must not sell more stock than the outlet has. Failure tests should include API timeouts, database interruptions and loss of the WhatsApp connection. Any performance result must state the device, network, dataset and number of test runs. A store pilot could compare checkout time, stock adjustments, invoice corrections and expiry-related actions before and after deployment. The project brief does not provide these measurements, so this paper makes no numerical claim about improvement.

Each measure should be defined before the pilot begins. Checkout time can be measured from the cashier's final confirmation to the server's success response. Barcode response time can run from the completed scan until the product appears in the cart. For inventory discrepancy, divide the absolute difference between the system quantity and the physical count by the physical count; report cases with a physical count of zero separately. Cancellation accuracy can be checked by comparing the quantity in the original sale with the quantity restored to stock. For expiry alerts, record how many flagged batches staff review or act on before their stated expiry date.

A useful pilot needs a baseline and a comparable period after staff have been trained. The report should include store size, number of active products and cashiers, test dates, network conditions, missing observations and any operational changes. Show the spread of results and unusual cases, not only averages. Staff feedback can explain why a step was skipped or corrected, but it should be kept separate from transaction-log measurements. With these details, the design claims can be tested. Without them, claims about faster checkout, savings or reduced waste would be premature.

5 Conclusion

Retail inventory management involves much more than recording sales. Incorrect stock records can affect replenishment and product availability. Perishable goods need expiry information, and stores must balance availability, waste, cost and demand from different channels. The studies reviewed provide evidence and models for audits, forecasting, automatic ordering, shelf allocation, waste reduction and stock visibility. Most focus on one decision rather than the complete operating process of a small multi-outlet POS. DailyMart aims to bring these routine records and controls together in one application. Whether it works as intended still needs to be tested.

6 Future scope

The next stage is to verify batch-level FEFO, complete transaction and concurrency testing, measure checkout and report response times, and run a pilot with cashiers and inventory staff. Customer returns, cash-drawer hardware, confirmed card-terminal payments and the official WhatsApp Cloud API can be added as separate modules later. Any larger study should explain its test setup, sample, measures and limitations.

References

[1] Qi M, Mak HY, Shen ZJM. Data-driven research in retail operations—A review. *Naval Research Logistics*. 2020;67(8):595–616. https://doi.org/10.1002/nav.21949

[2] Bassamboo A, Moreno A, Stamatopoulos I. Inventory auditing and replenishment using point-of-sales data. *Production and Operations Management*. 2020;29(5):1219–1231. https://doi.org/10.1111/poms.13153

[3] Shabani A, Maroti G, de Leeuw S, Dullaert W. Inventory record inaccuracy and store-level performance. *International Journal of Production Economics*. 2021;235:108111. https://doi.org/10.1016/j.ijpe.2021.108111

[4] Falcão J, Ruiz C, Pan S, Noh HY, Zhang P. FAIM: Vision and weight sensing fusion framework for autonomous inventory monitoring in convenience stores. *Frontiers in Built Environment*. 2020;6:568372. https://doi.org/10.3389/fbuil.2020.568372

[5] Andaur JMR, Ruz GA, Goycoolea M. Predicting out-of-stock using machine learning: An application in a retail packaged foods manufacturing company. *Electronics*. 2021;10(22):2787. https://doi.org/10.3390/electronics10222787

[6] Avlijas G, Vukanovic Dumanovic V, Radunovic M. Measuring the effects of automatic replenishment on product availability in retail stores. *Sustainability*. 2021;13(3):1391. https://doi.org/10.3390/su13031391

[7] Dharmawardane C, Sillanpää V, Holmström J. High-frequency forecasting for grocery point-of-sales: Intervention in practice and theoretical implications for operational design. *Operations Management Research*. 2021;14:38–60. https://doi.org/10.1007/s12063-020-00176-7

[8] Taleizadeh AA, Shokr I, Konstantaras I, VafaeiNejad M. Stock replenishment policies for a vendor-managed inventory in a retailing system. *Journal of Retailing and Consumer Services*. 2020;55:102137. https://doi.org/10.1016/j.jretconser.2020.102137

[9] Kim G, Moon I. Integrated planning for product selection, shelf-space allocation, and replenishment decision with elasticity and positioning effects. *Journal of Retailing and Consumer Services*. 2021;58:102274. https://doi.org/10.1016/j.jretconser.2020.102274

[10] Das L, Carrel AL, Caplice C. Pack size effects on retail store inventory and storage space needs. *INFOR: Information Systems and Operational Research*. 2021;59(3):465–494. https://doi.org/10.1080/03155986.2021.1918487

[11] de Moraes CC, de Costa FHO, Pereira CR, da Silva AL, Delai I. Retail food waste: Mapping causes and reduction practices. *Journal of Cleaner Production*. 2020;256:120124. https://doi.org/10.1016/j.jclepro.2020.120124

[12] Huang IY, Manning L, James KL, Grigoriadis V, Millington A, Wood V, et al. Food waste management: A review of retailers' business practices and their implications for sustainable value. *Journal of Cleaner Production*. 2021;285:125484. https://doi.org/10.1016/j.jclepro.2020.125484

[13] Riesenegger L, Hübner A. Reducing food waste at retail stores—An explorative study. *Sustainability*. 2022;14(5):2494. https://doi.org/10.3390/su14052494

[14] Richards TJ, Hamilton SF. Inventory management and loss in beer retailing. *Agribusiness*. 2022;38(3):461–485. https://doi.org/10.1002/agr.21740

[15] Mallidis I, Vlachos D, Yakavenka V, Zafeiriou E. Development of a single period inventory planning model for perishable product redistribution. *Annals of Operations Research*. 2020;294(1):697–713. https://doi.org/10.1007/s10479-018-2948-2

[16] Herron CB, Garner LJ, Siddique A, Huang TS, Campbell JC, Rao S, et al. Building 'First Expire, First Out' models to predict food losses at retail due to cold chain disruption in the last mile. *Frontiers in Sustainable Food Systems*. 2022;6:1018807. https://doi.org/10.3389/fsufs.2022.1018807

[17] Saghiri S, Aktas E, Mohammadipour M. Grocery omnichannel perishable inventories: Performance measures and influencing factors. *International Journal of Operations & Production Management*. 2023;43(12):1891–1919. https://doi.org/10.1108/IJOPM-06-2022-0397

[18] Hansen K, Misra K, Sanders RE. Uninformed choices in perishables. *Marketing Science*. 2024;43(4):751–777. https://doi.org/10.1287/mksc.2021.0264

[19] Chehrazi N, Sanders RE, Stamatopoulos I. Inventory information frictions explain price rigidity in perishable groceries. *Marketing Science*. 2025;44(2):411–436. https://doi.org/10.1287/mksc.2023.0473

[20] Goedhart J, Haijema R, Akkerman R. Inventory rationing and replenishment for an omni-channel retailer. *Computers & Operations Research*. 2022;140:105647. https://doi.org/10.1016/j.cor.2021.105647

[21] Silbermayr L, Waitz M. Omni-channel inventory management of perishable products under transshipment and substitution. *International Journal of Production Economics*. 2024;267:109089. https://doi.org/10.1016/j.ijpe.2023.109089